AFRL-AFOSR-VA-TR-2016-0105

BRI) Fault Tolerant Paradigms

BENJAMIN ONG
MICHIGAN STATE UNIV EAST LANSING

02/26/2016
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ RTA2
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
02-09-2016	Final	11/30/2012 -- 11/29/2015

4. TITLE AND SUBTITLE
AFOSR FA9550-12-1-0455 Final Report

5a. CONTRACT NUMBER

5b. GRANT NUMBER
AFOSR FA9550-12-1-0455

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
B. W. Ong, A. J. Christlieb and Y. Wang

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Michigan State University
Office of Sponsored Programs
426 Auditorium Road, East Lansing, 48824

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
801 N Randolph St, Room 732
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This project had three principle aims:
1. Improving the scalability and efficiency of ``Ultra-scale'' methods for grid-based solutions to time-dependent PDEs;
2. Sparse storage and reconstruction of information;
3. Build-in several levels of resiliencies to handle various hard faults in the system.

Progress was made in all three areas, leading to fifteen published refereed articles, five articles in review, one completed masters thesis, and one doctoral thesis in progress.

15. SUBJECT TERMS
Parallel-in-Time, Domain Decomposition, Sparse FFT, Phase Retrieval

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Benjamin Ong
U	U	U	UU	382	19b. TELEPHONE NUMBER *(Include area code)* 906-487-3367

Reset

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18
Adobe Professional 7.0

AFOSR FA9550-12-1-0455 Final Report

B. W. Ong, A. J. Christlieb, Y. Wang

February 9, 2016

Contents

Project Objectives

This project had three principle aims:

1. Improving the scalability and efficiency of "Ultra-scale" methods for grid-based solutions to time-dependent PDEs;
2. Sparse storage and reconstruction of information;
3. Build-in several levels of resiliencies to handle various hard faults in the system.

Progress was made in all three areas, leading to fifteen published refereed articles, five articles in review, one completed masters thesis, and one doctoral thesis in progress. Broadly, PI Ong and his research team worked primarily in objectives 1.) and 3.), developing parallel-in-time and domain decomposition transmission conditions to improve scalability and resiliency of computations. PI Christlieb, PI Wang and their respective research teams worked primarily in objective 2.), developing sparse FFT algorithms and tackling the phase retrieval problem.

Personnel

This projected supported three faculty, three postdoctoral fellows, and two graduate students. Mr. High completed his M.Sc. in the Department of Mathematics at MSU under the supervision of Dr. Ong, and is now pursuing a Ph.D in computational science at UIUC. Dr. Ala Alzaalig is still working on his doctoral degree at Michigan Technological University.
Faculty Supported:

- PI: Dr. Andrew J. Christlieb (2012–2015)
- PI: Dr. Benjamin W. Ong (2012–2015)
- PI: Dr. Yang Wang (2012 – 2014)

Postdoctoral Scholars Supported:

- Dr. Yang Liu (2012 – 2014)
- Dr. Ke Wang (2013 – 2014)
- Dr. Bankim Mandal (2014 – 2015)

Graduate Students supported:

- Mr. Scott High (2013, graduated with M.Sc)
- Mr. Ethan Novak (graduate research project, Summer 2015)
- Mr. Ala Alzaalig (PhD in progress)

Scientific Workshops/Conferences

In addition to the AFOSR Computational Mathematics annual review, results from research related to this project was disseminated at the following workshops/conferences. This list does not include departmental seminars/colloquia at various universities.

1. "The Phase Retrieval Problem", *IPAM Workshop on Adaptive Data Analysis*, Los Angeles, CA, 2012

2. "Minimal frames for Phase Retrieval", *Workshop on Phaseless Reconstruction*, FFT 2013, College Park, MD, 2013

3. "Robust Sub-Linear Time Fourier Algorithms" *SIAM Conference on Computational Science and Engineering*, Boston, MA, 2013

4. "An Optimized RIDC-DD Space-time Method for Time Dependent Partial Differential Equations", *SIAM Conference on Computational Science and Engineering*, Boston, MA, 2013

5. "The Phase Retrieval Problem", *International Conference on Approximation Theory and Applications*, Hong Kong, 2013

6. " Mathematical Investigation of Authorship Attribution: A Case Study", *New Trends in Applied Harmonic Analysis*, CIMPA 2013, Mar del Plata, Argentina, 2013

7. "Pipeline Schwarz Waveform Relaxation", *Domain Decomposition 22*, Lugano, Switzerland, 2013

8. "The Phase Retrieval Problem", *Workshop on Applied Harmonic Analysis and Approximation Theory*, Guangzhou, China, 2014

9. "A Robust and Efficient Phase Retrieval Algorithm", *5th International Conference on Scientific Computing and Partial Differential Equations*, Hong Kong, 2014

10. "Fast Phase Retrieval for High Dimensions", *AMS Spring Sectional Meeting*, East Lansing, 2015

11. "Sub-Linear Sparse Fourier Algorithm for High Dimensional Data", SIAM Annual Meeting 2014, Chicago, IL, 2015

12. "RIDC methods with stepsize control", *4th Workshop on parallel-in-time integration*, Dresden, Germany, 2015

Summary of Results

1. Fast phase retrieval for high-dimensions
   M. Iwen, A. Viswanathan, Y. Wang
   eprint arXiv:1501.02377

   Description: A fast phase retrieval method which is near-linear time, making it computationally feasible for large dimensional signals. Both theoretical and experimental results demonstrate the method's speed, accuracy, and robustness. We then use this new phase retrieval method to help establish the first known sublinear-time compressive phase retrieval algorithm capable of recovering a given s-sparse vector $\mathbf{x} \in \mathbb{C}^d$ (up to an unknown phase factor) in just $\mathcal{O}(s \log^5 s \cdot \log d)$-time using only $\mathcal{O}(s \log^4 s \cdot \log d)$ magnitude measurements.

2. Detection of edges from two-dimensional Fourier data using Gaussian mollifiers
   A. Gelb, G. Song, A. Viswanathan and Y. Wang
   eprint

   Description: The detection of edges from two-dimensional truncated Fourier data is studied. Compared to edge detection from pixel data, this is a more challenging problem since we seek accurate local information from a small number of often noisy global measurements. Here we develop a highly effective algorithm using a specific class of spectral mollifiers which converges uniformly to sharp peaks along the singular support of the function.

3. Random matrices and erasure-robust frames
   Y. Wang
   eprint arXiv:1403.5969

   Description: Data erasure and robustness are important considerations for building redundant systems (frames). Can you build a system (frame) which is robust against more than 50% data erasures? This was the conjectured upper bound within the community. This paper shows that there isn't in fact such an upper bound. The random Gaussian frames can be robust against data erasures of arbitrary high percentage of erasure.

4. On the decay of the smallest singular value of submatrices of rectangular matrices
   Y. Liu and Y. Wang

   Description: The main contribution of this paper is to show the connection between the singular value problem and a combinatorial geometry problem. Using a technique from integral geometry and from the perspective of combinatorial geometry, we show that the smallest singular value of submatrices is realted to the minimal distance of points to the lines connecting two other points in a bounded point set. The decay rate of the minimal distance for the set of points can then be estimated.

5. A distributed and incremental SVD algorithm for agglomerative data analysis on large networks
   M. A. Iwen and B. W. Ong
   eprint arXiv:1601.07010

   Description: In this paper, an algorithm is formulated to compute the singular value decomposition of highly-rectangular, distributed matrices efficiently using a hierachical approach. The algorithm is proven to recover exactly the exact decomposition if the rank of the input matrix is known a priori. Additionally, the algorithm can be used to recover the d-largest

singular vectors with bounded error. The algorithm is shown to be stable with respect to roundoff errors, or corruption of the original matrix entries.

6. Robust sparse phase retrieval made easy
   M. Iwen, A. Viswanathan and Y. Wang
   Applied and Computational Harmonic Analysis
   to appear
   doi: 10.1016/j.acha.2015.06.007

   Description: In this paper we develop a two stage phase retrieval algorithm for phase retrieval of sparse vectors. It is incredibly fast and robust. Furthermore it requires the optimally small number of measurements. Our algorithm also settles a conjecture on the number of measurements needed to perform phase retrieval for complex signals.

7. A multiscale sub-linear time Fourier algorithm for noisy data
   A. J. Christlieb and D.J. Lawlor and Y. Wang
   Applied and Computational Harmonic Analysis
   to appear
   doi: 10.1016/j.acha.2015.04.002

   Description: The sparse Fourier algorithm for noiseless signals is extended to the noisy setting. We present two such extensions, the second of which exhibits a novel form of error-correction not unlike that of the β-encoders in analog-to-digital conversion. The algorithm runs in time $O(k \log(k) \log(N/k))$ on average, provided the noise is not overwhelming. The error-correction property allows the algorithm to outperform FFTW over a wide range of sparsity and noise values, and is to the best of our knowledge novel in the sparse Fourier transform context.

8. Pipeline Schwarz Waveform Relaxation
   B. W. Ong, S. High and F. Kwok
   Lecture Notes in Computational Science and Engineering, Domain Decomposition Methods in Science and Engineering XXII
   to appear

   Description: Schwarz Waveform Relaxation methods are reposed to allow for pipeline parallelization. This increases the scalability of the waveform relaxation algorithms with high effiiency.

9. Algorithm xxx - a family of parallel time integrators
   B. W. Ong, R. D. Haynes and K. Ladd
   ACM Transactions on Mathematical Software
   to appear

   Description: The Revisionist Intergal Deferred Correction software, a parallel-in-time integrator, is able to bootstrap lower order time integrators to provide high-order approximations in approximately the same wall clock time. The user supplied time step routine may be explicit or implicit and may make use of any auxilliary libraries which take care of the solution of any nonlinear algebraic systems which may arise.

10. Stable signal recovery from phaseless measurements
    B. Gao, Y. Wang and Z. Wu
    Journal of Fourier Analysis and Applications

(2015) pp. 1–21
doi: 10.1007/s00041-015-9434-x

Description: This paper studies the stability of the ℓ_1 minimization for the compressive phase retrieval and to extend the instance-optimality in compressed sensing to the real phase retrieval setting. We first show that the $m = \mathcal{O}(k \log(N/k))$ measurements is enough to guarantee the ℓ_1 minimization to recover k-sparse signals stably provided the measurement matrix A satisfies the strong RIP property. We use the results to build a parallel between compressive phase retrieval with the classical compressive sensing.

11. Gabor orthonormal bases generated by the unit cubes
J.-P. Gabardo, C.-K. Lai and Y. Wang
Journal of Functional Analysis
Vol. 269 (2015), pp 1515–1538

Description: This paper studies Gabor orthonormal bases generated by the characteristic functions of a unit cube. A complete characterization is given.

12. Probabilistic Estimates of the Largest Strictly Convex Singular Values of Pregaussian Random Matrices
Y. Liu
Journal of Mathematics and Statistics (2015)

Description: The p-singular values of random matrices with Gaussian entries defined in terms of the l_p-p-norm for $p > 1$ is studied.

13. The probabilistic estimates on the largest and smallest q-singular values of random matrices
M.-J. Lai and Y. Liu
Mathematics of Computation
84:294 (2015), pp. 1775 – 1794
doi: 10.1090/S0025-5718-2014-02895-0

Description: In this paper, the q-singular values of random matrices with pregaussian entries in the case $0 < q \leq 1$ are studied. The main result are decay estimates on the lower and upper tail probabilities of the q-singular values. The k-th q-singular value of an $m \times n$ matrix A is defined by

$$s_k^{(q)} = \inf_V \sup_{x \in V \setminus \{0\}} \frac{\|Ax\|_q}{\|x\|_q},$$

where $\|\cdot\|_q$ denotes the l^q-quasinorm ($q \geq 0$) and the inf is taken over all linear subspace $V \in \mathbb{R}^n$ of dimension at least $n - k + 1$.

14. A new approach for analyzing physiological time series
D. Mao, Y. Wang, and Q. Wu
Advances in Adaptive Data Analysis
(2015), pp 1550001
doi: 10.1142/S1793536915500016

Description: We developed a new approach for the analysis of physiological time series for the purpose of detection and classification. An iterative convolution filter is used to decompose the time series into various components. Statistics of these components are extracted as

features to characterize the mechanisms underlying the time series. Motivated by the studies that show many normal physiological systems involve irregularity while the decrease of irregularity usually implies the abnormality, the statistics for "outliers" in the components are used as features measuring irregularity. Support vector machines are used to select the most relevant features that are able to differentiate the time series from normal and abnormal systems. This new approach is successfully used in the study of congestive heart failure by heart beat interval time series.

15. Multiple authors detection: a quantitative analysis of Dream of the Red Chamber
X. Hu, Y. Wang and Q. Wu
Advances in Adaptive Data Analysis
Vol 6, Issue 4 (2014), pp 1450012
doi: 10.1142/S1793536914500125

Description: We develop an robust method based on machine learning as well as an effective set of features for the detection of multiple authorship within a book. We apply our method to the historic authorship controversy to show that the commonly read version of *Dream of the Red Chamber*, one of the greatest novel in the Chinese literature, must be written by two authors as suspected.

16. Invertibility and robustness of phaseless reconstruction
R. Balan and Y. Wang
Applied and Computational Harmonic Analysis
Vol 38 (2015), pp. 469–488
doi: 10.1016/j.acha.2014.07.003

Description: This paper is concerned with the question of reconstructing a vector in a finite-dimensional real Hilbert space when only the magnitudes of the coefficients of the vector under a redundant linear map are known. We analyze various Lipschitz bounds of the nonlinear analysis map and we establish theoretical performance bounds of any reconstruction algorithm. We show that robust and stable reconstruction requires additional redundancy than the critical threshold.

17. Revisionist integral deferred correction with adaptive stepsize control
A. Christlieb, C. Macdonald, B. Ong and R. Spiteri
Communications in Applied Mathematics and Computational Science
Vol 10, Number 1 (2015), pp. 1–25
doi: 10.2140/camcos.2015.10.1

Description: This paper builds stepsize control into the revisionist integral deferred correction framework. Three variants are explored. In the most successful variant, the prediction level is used for step-size control.

18. Phase retrieval for sparse signals
Y. Wang and Z. Xu
Applied and Computational Harmonic Analysis
Vol 37 (2014), pp. 531-544
doi: 10.1016/j.acha.2014.04.001

Description: In this paper we provide a theoretical foundation for sparse signal phase retrieval. We build a parallel frame for sparse signal phase retrieval that is analogous to

the theoretical framework for compressive sensing. In particular, we extend the RIP property and the Null Space property from compressive sensing to sparse phase retrieval.

19. Phase retrieval from very few measurements
M. Fickus, D. Mixon, A. Nelson and Y. Wang
Linear Algebra and Appl.
Vol 449 (2014), pp. 475–499
doi: 10.1016/j.laa.2014.02.011

Description: In this paper we provide a specific construction for phase retrieval in the complex setting where only $4n - 4$ measurements are needed. This is conjectured to be the smallest number of measurements for which phase retrieval is possible in the complex setting.

20. A hybrid MPI–OpenMP algorithm for the parallel space-time solution of time dependent PDES
B. W. Ong, R. D. Haynes
Lecture Notes in Computational Science and Engineering, Domain Decomposition Methods in Science and Engineering XXI
Vol 98 (2014), pp. 179–187
doi: 10.1007/978-3-319-05789-7_14

Description: The significance in correctly ordering parallel directives for the parallel space-time solution of time-dependent PDEs using revisionist integral deferred correction (a parallel time-integrator implemented using OpenMP) and domain decomposition (implemented using MPI) is explored. Surprisingly, a tightly-coupled fork–join implementation is more efficient.

FAST PHASE RETRIEVAL FOR HIGH-DIMENSIONS

MARK IWEN, ADITYA VISWANATHAN, AND YANG WANG

Abstract. We develop a fast phase retrieval method which is near-linear time, making it computationally feasible for large dimensional signals. Both theoretical and experimental results demonstrate the method's speed, accuracy, and robustness. We then use this new phase retrieval method to help establish the first known sublinear-time compressive phase retrieval algorithm capable of recovering a given s-sparse vector $\mathbf{x} \in \mathbb{C}^d$ (up to an unknown phase factor) in just $\mathcal{O}(s \log^5 s \cdot \log d)$-time using only $\mathcal{O}(s \log^4 s \cdot \log d)$ magnitude measurements.

1. Introduction

We consider the *phase retrieval problem* of recovering a given vector $\mathbf{x} \in \mathbb{C}^d$, up to an unknown global phase factor, from a set of squared magnitude measurements $|M\mathbf{x}|^2 \in \mathbb{R}^D$, with $D \geq d$. Here $M \in \mathbb{C}^{D \times d}$, and $|\cdot|^2 : \mathbb{C}^D \to \mathbb{R}^D$ computes the componentwise squared magnitude of each vector entry. Our objective is to design a computationally efficient recovery method, $\mathcal{A} : \mathbb{R}^D \to \mathbb{C}^d$, which can approximately recover $\mathbf{x}$ using the magnitude measurements $|M\mathbf{x}|^2$ that result from any member of a relatively large class of matrices $M \in \mathbb{C}^{D \times d}$. More specifically, we require that

$$\mathcal{A}\left(|M\mathbf{x}|^2\right) = \mathrm{e}^{-\mathrm{i}\theta}\mathbf{x} \tag{1}$$

for some unknown $\theta \in [0, 2\pi]$.

Phase retrieval problems arise in many crystallography and optics applications (see, e.g., [40, 30, 21, 29]). As a result, phase retrieval has been studied a great deal over the past decade within the applied mathematics community. The majority of this work has focussed on establishing upper and lower bounds for the number of magnitude measurements required for reconstructing $\mathbf{x}$ up to a global phase factor. It has been shown, e.g., that $\mathcal{O}(d)$ magnitude measurements suffice for phase retrieval of both real and complex vectors $\mathbf{x} \in \mathbb{C}^d$ [3, 6, 17]. Furthermore, it is also known that $\mathcal{O}(d)$ magnitude measurements are required [22].

There has also been a good deal of work done developing phase retrieval algorithms which are (i) computationally efficient, (ii) robust to measurement noise, and (iii) theoretically guaranteed to reconstruct a given vector up to a global phase error using a near-minimal number of magnitude measurements. For example, it has been shown that robust phase retrieval is possible with $D = \mathcal{O}(d)$ magnitude measurements by solving a semidefinite programming relaxation of it as a rank-1 matrix recovery problem [12, 11]. This allows polynomial-time convex optimization methods to be used for phase retrieval. Furthermore, the runtimes of these convexity-based methods can be reduced with the use of $\mathcal{O}(d \log d)$ magnitude measurements [14]. Other phase retrieval approaches include the use of spectral recovery methods together with magnitude measurement ensembles inspired by expander graphs [2]. These methods allow the recovery of $\mathbf{x}$ up to a global phase factor

M.A. Iwen: Department of Mathematics and Department of ECE, Michigan State University (markiwen@math.msu.edu). M.A. Iwen was supported in part by NSF DMS-1416752 and NSA H98230-13-1-0275.

A. Viswanathan: Department of Mathematics, Michigan State University (aditya@math.msu.edu).

Y. Wang: Department of Mathematics, The Hong Kong University of Science and Technology (yangwang@ust.hk). Y. Wang was partially supported by NSF DMS-1043032 and AFOSR FA9550-12-1-0455.

using $\mathcal{O}(d)$ magnitude measurements, and run in $\Omega(d^2)$-time in general.[1] All of these approaches utilize magnitude measurements $|M\mathbf{x}|^2$ resulting from either (i) Gaussian random matrices M, or (ii) unbalanced expander graph constructions, in order to prove their recovery guarantees.

In this paper we demonstrate that a relatively general class of invertible block circulant measurement matrices $M \in \mathbb{C}^{D\times d}$ results in $D = \mathcal{O}(d\log^c d)$ magnitude measurements, $|M\mathbf{x}|^2$, which allow for phase retrieval in just $\mathcal{O}(d\log^c d)$-time.[2] In particular, we construct a well-conditioned set of Fourier-based measurements, $M \in \mathbb{C}^{\mathcal{O}(d\log^3 d)\times d}$, which are theoretically guaranteed to allow for the phase retrieval of a given vector with high probability in $\mathcal{O}(d\log^4 d)$-time. These measurements are of particular interest given that they are closely related to short-time Fourier transform based measurements, which are off special significance in several application areas (see, e.g., [15] and the references therein). Numerical experiments both verify the speed and accuracy of the proposed phase retrieval approach, as well as indicate that the approach is highly robust to measurement noise. Finally, after establishing and analyzing our general phase retrieval method, we then utilize it in order to establish the first known sublinear-time compressive phase retrieval method capable of recovering s-sparse vectors $\mathbf{x}$ (up to an unknown phase factor) in only $\mathcal{O}(s\log^c d)$-time.

The remainder of this paper is organized as follows: In section 2 we establish notation and discuss important preliminary results. Next, in section 3, we present our general phase retrieval algorithm and discuss it's runtime complexity. We then analyze the our phase retrieval algorithm and prove recovery guarantees for specific types of Fourier-based measurement matrices in section 4. In section 5, we empirically evaluate the proposed phase retrieval method for speed and robustness. Finally, in section 6, we use our general phase retrieval algorithm in order to construct a sublinear-time compressive phase retrieval method which is guaranteed to recover sparse vectors (up to an unknown phase factor) in near-optimal time.

2. Preliminaries: Notation and Setup

For any matrix $X \in \mathbb{C}^{D\times d}$ we will denote the j^{th} column of X by $\mathbf{X}_j \in \mathbb{C}^D$. The conjugate transpose of a matrix $X \in \mathbb{R}^{D\times d}$ will be denoted by $X^* \in \mathbb{C}^{d\times D}$, and the singular values of any matrix $X \in \mathbb{C}^{D\times d}$ will always be ordered as $\sigma_1(X) \geq \sigma_2(X) \geq \cdots \geq \sigma_{\min(D,d)}(X) \geq 0$. Also, the condition number of the matrix X will denoted by $\kappa(X) := \sigma_1(X)/\sigma_{\min(D,d)}(X)$. We will use the notation $[n] := \{1,\ldots,n\} \subset \mathbb{N}$ for any $n \in \mathbb{N}$. Finally, given any $\mathbf{x} \in \mathbb{C}^d$, the vector ${\mathbf{x}_s}^{\text{opt}} \in \mathbb{C}^d$ will always denote an optimal s-sparse approximation to $\mathbf{x}$. That is, it preserves the s largest entries in magnitudes of $\mathbf{x}$ while setting the rest of the entires to 0. Note that ${\mathbf{x}_s}^{\text{opt}} \in \mathbb{C}^d$ may not be unique as there can be ties for the s^{th} largest entry in magnitude.

Hereafter we will assume that our measurement matrix $M \in \mathbb{C}^{D\times d}$ has $D := (2\delta-1)d$ rows, for a user specified value of $\delta \in \mathbb{N}$. Furthermore, we utilize the obvious decomposition of M into $(2\delta-1)$ blocks, $M_1,\ldots,M_{2\delta-1} \in \mathbb{C}^{d\times d}$, given by

$$M = \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_{2\delta-1} \end{pmatrix}. \tag{2}$$

Each $M_l \in \mathbb{C}^{d\times d}$ is itself assumed to be both circulant, with

$$(M_l)_{i,j} := (\mathbf{m}_l)_{(j-i) \bmod d + 1} \tag{3}$$

for some $\mathbf{m}_l \in \mathbb{C}^d$, and banded, so that $(\mathbf{m}_l)_i = 0$ for all $i > \delta$, and $1 \leq l \leq 2\delta-1$.[3]

[1] Their runtime complexity is dominated by the time required to solve an overdetermined linear system.

[2] Herein c is a fixed absolute constant.

[3] All indexes of vectors in $\mathbb{C}^d$ will automatically be considered modulo d, + 1, in this fashion hereafter.

As a consequence, the squared magnitude measurements from the l^{th}-block, $|M_l\mathbf{x}|^2 \in \mathbb{R}^d$, can be rewritten as

$$\left(|M_l\mathbf{x}|^2\right)_i = (M_l\mathbf{x})_i\,\overline{(M_l\mathbf{x})_i} = \sum_{j,k=1}^{\delta} (\mathbf{m}_l)_j\overline{(\mathbf{m}_l)}_k\; x_{j+i-1}\overline{x}_{k+i-1}. \tag{4}$$

Let $\mathbf{y} \in \mathbb{C}^D$ be defined by

$$y_i := x_{\lceil \frac{i+\delta-1}{2\delta-1} \rceil}\overline{x}_{\lceil \frac{i+\delta-1}{2\delta-1} \rceil + ((i+\delta-2) \bmod (2\delta-1)) - \delta + 1}. \tag{5}$$

Furthermore, let $\mathbf{0}_\alpha \in \mathbb{R}^{1\times\alpha}$ be the row vector of α zeros for any given $\alpha \in \mathbb{N}$, and let $\tilde{\mathbf{m}}_{(l,1)} \in \mathbb{C}^{1\times\delta}$ be such that

$$\left(\tilde{\mathbf{m}}_{(l,j)}\right)_k := (\mathbf{m}_l)_j\overline{(\mathbf{m}_l)}_k. \tag{6}$$

We can now re-express $|M_l\mathbf{x}|^2 \in \mathbb{R}^d$ from (4) as $\tilde{M}_l\mathbf{y}$, where $\tilde{M}_l \in \mathbb{C}^{d\times D}$ is a $(2\delta-1)$-circulant matrix defined by

$$\begin{pmatrix} \tilde{\mathbf{m}}_{(l,1)} & \mathbf{0}_{\delta-2} & \tilde{\mathbf{m}}_{(l,2)} & \mathbf{0}_{\delta-2} & \tilde{\mathbf{m}}_{(l,3)} & \dots & \tilde{\mathbf{m}}_{(l,\delta)} & 0 & 0 & \dots & 0 \\ \mathbf{0}_{2\delta-1} & \tilde{\mathbf{m}}_{(l,1)} & \mathbf{0}_{\delta-2} & \tilde{\mathbf{m}}_{(l,2)} & \mathbf{0}_{\delta-2} & \tilde{\mathbf{m}}_{(l,3)} & \dots & \tilde{\mathbf{m}}_{(l,\delta)} & 0 & \dots & 0 \\ & & & & & \ddots & & & & & \\ \left(\tilde{\mathbf{m}}_{(l,2)}\right)_2 & \dots & \left(\tilde{\mathbf{m}}_{(l,2)}\right)_\delta & \mathbf{0}_{\delta-2} & \tilde{\mathbf{m}}_{(l,3)} & \mathbf{0}_{\delta-2} & \dots & 0 & \tilde{\mathbf{m}}_{(l,1)} & \mathbf{0}_{\delta-2} & \left(\tilde{\mathbf{m}}_{(l,2)}\right)_1 \end{pmatrix}.$$

Finally, after reordering the entries of $|M\mathbf{x}|^2$ via a permutation matrix $P \in \{0,1\}^{D\times D}$, we arrive at our final form

$$P\,|M\mathbf{x}|^2 = M'\mathbf{y} = \begin{pmatrix} M'_1 & M'_2 & \dots & M'_\delta & 0 & 0 & \dots & 0 \\ 0 & M'_1 & M'_2 & \dots & M'_\delta & 0 & \dots & 0 \\ & & & \ddots & & & & \\ M'_2 & \dots & M'_\delta & 0 & \dots & 0 & \dots & M'_1 \end{pmatrix}\mathbf{y}. \tag{7}$$

Here $M' \in \mathbb{C}^{D\times D}$ is a block circulant matrix [38] whose blocks, $M'_1, \dots, M'_\delta \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$, have entries

$$(M'_l)_{i,j} := \begin{cases} (\mathbf{m}_i)_l\overline{(\mathbf{m}_i)}_{j+l-1} & \text{if } 1 \le j \le \delta - l + 1 \\ 0 & \text{if } \delta - l + 2 \le j \le 2\delta - l - 1 \\ (\mathbf{m}_i)_{l+1}\overline{(\mathbf{m}_i)}_{l+j-2\delta+1} & \text{if } 2\delta - l \le j \le 2\delta - 1, \text{ and } l < \delta \\ 0 & \text{if } j > 1, \text{ and } l = \delta \end{cases}. \tag{8}$$

Let I_α denote the $\alpha \times \alpha$ identity matrix. We now note that M' can be block diagonalized by via the unitary block Fourier matrices $U_\alpha \in \mathbb{C}^{\alpha d\times \alpha d}$, with parameter $\alpha \in \mathbb{N}$, defined by

$$U_\alpha := \frac{1}{\sqrt{d}} \begin{pmatrix} I_\alpha & I_\alpha & \dots & I_\alpha \\ I_\alpha & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}}{d}} & \dots & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot(d-1)}{d}} \\ & & \ddots & \\ I_\alpha & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot(d-2)}{d}} & \dots & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot(d-2)\cdot(d-1)}{d}} \\ I_\alpha & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot(d-1)}{d}} & \dots & I_\alpha \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot(d-1)\cdot(d-1)}{d}} \end{pmatrix}. \tag{9}$$

More precisely, one can see that we have

$$U_{2\delta-1}^{*}\ M'\ U_{2\delta-1} = J := \begin{pmatrix} J_1 & 0 & 0 & \dots & 0 \\ 0 & J_2 & 0 & \dots & 0 \\ & & \ddots & & \\ 0 & 0 & 0 & J_{d-1} & 0 \\ 0 & 0 & 0 & 0 & J_d \end{pmatrix} \tag{10}$$

where $J \in \mathbb{C}^{D\times D}$ is block diagonal with blocks $J_1, \cdots, J_d \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$ given by

$$J_k := \sum_{l=1}^{\delta} M_l' \cdot \mathbb{e}^{\frac{2\pi \mathbb{i}\cdot k\cdot l}{d}}. \tag{11}$$

Not so surprisingly, the fact that any block circulant matrix can be block diagonalized by block Fourier matrices will lead to more efficient computational techniques below.

2.1. **Johnson-Lindenstrauss Embeddings and Restricted Isometries.** Below we will utilize results concerning *Johnson-Lindenstrauss embeddings* [26, 19, 1, 13, 4, 27] of a given finite set $\mathcal{S} \subset \mathbb{C}^d$ into $\mathbb{C}^m$ for $m < d$. These are defined as follows:

Definition 1. *Let $\epsilon \in (0,1)$, and $\mathcal{S} \subset \mathbb{C}^d$ be finite. An $m \times d$ matrix A is a linear Johnson-Lindenstrauss embedding of $\mathcal{S}$ into $\mathbb{C}^m$ if*

$$(1-\epsilon)\| \mathbf{u}-\mathbf{v} \|_2^2 \leq \| A\mathbf{u} - A\mathbf{v} \|_2^2 \leq (1+\epsilon)\| \mathbf{u}-\mathbf{v} \|_2^2$$

holds $\forall \mathbf{u}, \mathbf{v} \in \mathcal{S} \cup \{\mathbf{0}\}$. In this case we will say that A is a JL(m,d,ϵ)-embedding of $\mathcal{S}$ into $\mathbb{C}^m$.

Linear JL(m,d,ϵ)-embeddings are closely related to the *Restricted Isometry Property* [9, 4, 18].

Definition 2. *Let $s \in [d]$ and $\epsilon \in (0,1)$. The matrix $A \in \mathbb{C}^{m\times d}$ has the Restricted Isometry Property if*

$$(1-\epsilon)\| \mathbf{x} \|_2^2 \leq \| A\mathbf{x} \|_2^2 \leq (1+\epsilon)\| \mathbf{x} \|_2^2 \tag{12}$$

holds $\forall \mathbf{x} \in \mathbb{C}^d$ containing at most s nonzero coordinates. In this case we will say that A is RIP(s,ϵ).

In particular, the following theorem due to Krahmer and Ward [27, 18] demonstrates that a matrix with the restricted isometry property can be used to construct a Johnson-Lindenstrauss embedding matrix.

Theorem 1. *Let $\mathcal{S} \subset \mathbb{C}^d$ be a finite point set with $|\mathcal{S}| = M$. For $\epsilon, p \in (0,1)$, let $A \in \mathbb{C}^{m\times d}$ be RIP($2s$,ϵ/C_1) for some $s \geq C_2 \cdot \ln(4M/p)$.[4] Finally, let $B \in \{-1,0,1\}^{d\times d}$ be a random diagonal matrix with independent and identically distributed (i.i.d.) symmetric Bernoulli entries on its diagonal. Then, AB is a JL(m,d,ϵ)-embedding of $\mathcal{S}$ into $\mathbb{C}^m$ with probability at least $1-p$.*

Below we will utilize Theorem 1 together with a result concerning the restricted isometry property for sub-matrices of a Fourier matrix. Let $F \in \mathbb{C}^{d\times d}$ be the unitary $d \times d$ discrete Fourier transform matrix. The *random sampling matrix*, $R' \in \mathbb{C}^{m\times d}$, for F is then

$$R' := \sqrt{\frac{d}{m}} \cdot RF \tag{13}$$

where $R \in \{0,1\}^{m\times d}$ is a random matrix with exactly one nonzero entry per row (i.e., each entry's column position is drawn independently from $[d]$ uniformly at random with replacement). The following theorem is proven in [18].[5]

[4] Here $C_1, C_2 \in (1,\infty)$ are both fixed absolute constants.

[5] See Theorem 12.32 in Chapter 12.

Theorem 2. *Let $p \in (0,1)$. If the number of rows in the random sampling matrix $R' \in \mathbb{C}^{m\times d}$ satisfies both*

$$\frac{m}{\ln(9m)} \geq C_3 \cdot \frac{s\ln^2(8s)\ln(8d)}{\epsilon^2} \tag{14}$$

and

$$m \geq C_4 \cdot \frac{s\log(1/p)}{\epsilon^2}, \tag{15}$$

then R' will be RIP($2s$,ϵ/C_1) with probability at least $1-p$.[6]

We are now prepared to present and analyze our phase retrieval method.

3. A Fast Phase Retrieval Algorithm

The proposed phase retrieval algorithm works in two stages. In the first stage, the vector $\mathbf{y} \in \mathbb{C}^D$ from (5) of local entrywise products of $\mathbf{x} \in \mathbb{C}^d$ with its conjugate is recovered by inverting the block circulant matrix M' in (7). Next, a greedy algorithm is used to recover the magnitudes and phases of each entry of $\mathbf{x}$ from $\mathbf{y}$ (up to a global phase factor). To see how this works, note that $\mathbf{y}$ will contain all of the products $x_i\overline{x}_j$ for all $i,j \in [d]$ with $|i-j \bmod d| < \delta$. As a result, the magnitude of each entry x_j can be obtained directly from $x_j\overline{x}_j = |x_j|^2$. Similarly, as long as $x_j\overline{x}_j > 0$, one can also compute the phase difference $\arg(x_i) - \arg(x_j)$ from $\arg\left(\frac{x_i\overline{x}_j}{x_j\overline{x}_j}\right)$. Thus, the phase of x_i can be determined once $\arg(x_j)$ is established. Repeating this process allows one to determine a network of phase differences which all depend uniquely on the choice of a single entry's unknown phase. This entry's phase becomes the global phase factor $\mathrm{e}^{\mathrm{i}\theta}$ from (1). See Algorithm 1 for additional details.

Algorithm 1 Fast Phase Retrieval

Input: Measurements $|M\mathbf{x}|^2 \in \mathbb{R}^D$ (Recall, e.g., (2) – (4))
Output: $\tilde{\mathbf{x}} \in \mathbb{C}^d$ with $\tilde{\mathbf{x}} \approx \mathrm{e}^{-\mathrm{i}\theta}\mathbf{x}$ for some $\theta \in [0, 2\pi]$ as per (1)
1: Compute $\mathbf{y} = (M')^{-1}P\,|M\mathbf{x}|^2$ (see (7))
2: Use Algorithm 2 with input $\mathbf{y} \in \mathbb{C}^D$ to compute the phase angles, ϕ_j, of $\tilde{x}_j$ for all $j \in [d]$
3: Set $\tilde{x}_j = \sqrt{x_j\overline{x}_j} \cdot \mathrm{e}^{\mathrm{i}\phi_j}$ for all $j \in [d]$, where each $x_j\overline{x}_j$ is obtained from $\mathbf{y}$

It is important to note that Algorithm 1 assumes that the block circulant matrix M' arising from our choice of measurements, M, is invertible. As we shall see in §4 and §5, this is relatively easy to achieve. Similarly, Algorithm 2 implicitly assumes that $\mathbf{x}$ does not contain any strings of $\delta-1$ consecutive zeros (or, more generally, $\delta-1$ consecutive entires with "very small" magnitudes). This assumption will also be discussed in §4 and §5, and justified for arbitrary $\mathbf{x}$ by modifying the measurements M. For the time being, then, we are left free to consider to the computational complexity of Algorithm 1.

3.1. Runtime Analysis.

We will begin our analysis the runtime complexity of Algorithm 1 by considering the computation of $\mathbf{y} \in \mathbb{C}^D$ in line 1. Recalling §2, we note that the permutation matrix P is based on a simple row reordering that clusters the first rows of $M_1,\ldots,M_{2\delta-1}$ into a contiguous block, the second rows of $M_1,\ldots,M_{2\delta-1}$ into a second contiguous block, etc. (see (2) and (3)). Thus, $P|M\mathbf{x}|^2$ is simple to compute using only $\mathcal{O}(d\cdot\delta)$-operations. To finish calculating $\mathbf{y} = (M')^{-1}P\,|M\mathbf{x}|^2$ we then use the decomposition of M' from (10) and compute $\mathbf{y} = U_{2\delta-1}J^{-1}U_{2\delta-1}^*P\,|M\mathbf{x}|^2$.

[6]Here $C_3, C_4 \in (1,\infty)$ are both fixed absolute constants.

Algorithm 2 Naive Greedy Angular Synchronization

Input: $x_i\overline{x}_j, \quad i,j = 0,\dots,d-1, \quad |i-j \bmod d| < \delta.$

Output: Relative phase values: $\angle x_i, \quad i = 0,\dots,d-1.$

1: Identify largest magnitude entry and set its phase to zero.

$$\angle x_j = 0, \qquad j = \arg\max_i \ x_i\overline{x}_i, \quad i = 0,\dots,d-1.$$

Note: We recover the unknown phases up to a global phase factor.

2: Define a binary vector, phaseFlag $\in \{0,1\}^d$, to keep track of entries whose phase has already been set.

$$\text{phaseFlag}_i = \begin{cases} 0, & i = j, \\ 1, & \text{else.} \end{cases}$$

3: **while** $\displaystyle\sum_{i\in(j,j+\delta)} \text{phaseFlag}_i > 0$ **do**

4: **for** $i = 1-\delta, 2-\delta, \dots, 0, \dots, \delta-1$ **do** {Set phase for the $2\delta-1$ entries nearest x_j}

5: **if** $\text{phaseFlag}_{j+i \bmod d} = 1$ **then** {Do not over-write previously set phases}

6: Use the reference phase, $\angle x_j$, and the computed phase differences, $\arg\left(\frac{x_{j+i \bmod d}\overline{x}_j}{x_j\overline{x}_j}\right)$ and $\arg\left(\frac{x_j\overline{x}_{j+i \bmod d}}{x_j\overline{x}_j}\right)$, to set the phase of entry $x_{j+i \bmod d}$

$$\angle x_{j+i \bmod d} = \angle x_j + \frac{1}{2}\left(\arg\left(\frac{x_{j+i \bmod d}\overline{x}_j}{x_j\overline{x}_j}\right) - \arg\left(\frac{x_j\overline{x}_{j+i \bmod d}}{x_j\overline{x}_j}\right)\right).$$

$$\text{phaseFlag}_{j+i \bmod d} = 0.$$

7: **end if**

8: **end for**

9: Update the reference phase

$$j = \left(j + \arg\max_{0<i<\delta} \ x_{j+i \bmod d}\overline{x}_{j+i \bmod d}\right) \bmod d$$

10: **end while**

Recalling the definition of $U_{2\delta-1}$ (9), one can see that both $U_{2\delta-1}$ and $U^*_{2\delta-1}$ have fast matrix-vector multiplies (i.e., because they can be computed by performing $2\delta-1$ independent fast Fourier transforms on different sub-vectors of size d). Hence, matrix-vector multiplies with both of these matrices can be accomplished with $\mathcal{O}(\delta \cdot d\log d)$ operations. Finally, J is block-diagonal with d blocks of size $(2\delta-1)\times(2\delta-1)$ (see (11)). Thus, J and J^{-1} can both be computed using $\mathcal{O}(d\cdot\delta^3)$ total operations. Putting everything together, we can now see that line 1 of Algorithm 1 requires only $\mathcal{O}(d\cdot\delta^3 + \delta\cdot d\log d)$ operations in general. Furthermore, these computations can easily benefit from parallelism due to the fact that the calculations above are all based on explicitly defined block decompositions.

The second line of Algorithm 1 calls Algorithm 2 whose runtime complexity is dominated by its main while-loop (lines 3 through 10). This loop will visit each entry of the input vector $\mathbf{y}$ at most a constant number of times. Hence, it requires $\mathcal{O}(\delta\cdot d)$ operations. Finally, the third line of Algorithm 1 uses only $\mathcal{O}(d)$ operations. Thus, the total runtime complexity of Algorithm 1 is $\mathcal{O}(d\cdot\delta^3 + \delta\cdot d\log d)$ in general.

4. Error Analysis and Recovery Guarantees

In this section we analyze the performance of the proposed phase retrieval method (see Algorithm 1), and demonstrate measurement matrices which allow it to recover arbitrary vectors, up to

an unknown phase factor, with high probability. Our analysis proceeds in two steps. First, in §4.1 and §4.2, we construct a deterministic set of measurements, $M \in \mathbb{C}^{D\times d}$, which allow Algorithm 1 to recover all relatively flat vectors $\mathbf{x} \in \mathbb{C}^d$. Here, "flat" simply means that all entrees of $\mathbf{x}$ are bounded away from zero in magnitude. The developed measurements M are Fourier-like, roughly corresponding to a set of damped and windowed Fourier measurements of overlapping portions of $\mathbf{x}$. In addition to being well conditioned, these Fourier measurements also have fast inverse matrix-vector multiplies via (an additional usage of) the FFT. Hence, they confer additional computational advantages beyond those already enjoyed by our general block circulant measurement setup.

Next, in §4.3, we extend our deterministic recovery guarantee for flat vectors to a probabilistic recovery guarantee for arbitrary vectors. This is accomplished by right-multiplying M with a concatenation of several Johnson-Lindenstrauss embedding matrices, each of which tends to "flatten out" vectors they are multiplied against. In particular, we construct a set of such matrices which are both (i) collectively unitary, and (ii) rapidly invertible as a group via (yet another usage of) the FFT. The fact that this flattening matrix is unitary preserves the well conditioned nature of our initial measurements, M. Furthermore, the fact that the flattening matrix enjoys a fast inverse matrix-vector multiply via the FFT allows us to maintain computational efficiency. Finally, the fact that the flattening matrix produces a flattened version of $\mathbf{x}$ with high probability allows us to apply our deterministic recovery guarantee for flat vectors to vectors which are not initially flat. The end result of this line of reasoning is the following recovery guarantee for noiseless measurements.

Theorem 3. *Let $\mathbf{x} \in \mathbb{C}^d$ with d sufficiently large. Then, one can select a random measurement matrix $\tilde{M} \in \mathbb{C}^{D\times d}$ such that the following holds with probability at least $1 - \frac{1}{C\cdot\ln^2(d)\cdot\ln^3(\ln d)}$.*[7] *Algorithm 1 will recover an $\tilde{\mathbf{x}} \in \mathbb{C}^d$ with*

$$\min_{\theta\in[0,2\pi]} \left\| \mathbf{x} - \mathrm{e}^{\mathrm{i}\theta}\tilde{\mathbf{x}} \right\|_2 = 0 \tag{16}$$

when given the noiseless magnitude measurements $|\tilde{M}\mathbf{x}|^2 \in \mathbb{R}^D$. Here D can be chosen to be $\mathcal{O}(d\cdot\ln^2(d)\cdot\ln^3(\ln d))$. Furthermore, Algorithm 1 will run in $\mathcal{O}(d\cdot\ln^3(d)\cdot\ln^3(\ln d))$-time in that case.

In fact, we obtain a bit more than this most basic noiseless recovery result. For example, we derive explicit bounds on the condition number of the measurements M' proposed in §4.1 (as opposed to simply proving them to be invertible). Continuing in this vein one can, in fact, easily prove rather ugly (and not terribly enlightening) worst-case recovery guarantees for Algorithm 1 when it's provided with noisy magnitude measurements instead of noiseless ones. However, we will leave a careful theoretical analysis of the robustness of Algorithm 1 to measurement noise for future work. For now, we simply direct the concerned reader to §5 after noting that Algorithm 1 appears to be highly robust to measurement noise in practice. We are now ready to begin proving Theorem 3.

4.1. **Well Conditioned Measurements.** In this section we develop a set of deterministic measurements $M \in \mathbb{C}^{D\times d}$ that lead to well conditioned block circulant matrices $M' \in \mathbb{C}^{D\times D}$ in (7). To begin, we choose $a \in [4,\infty)$ and then set

$$(\mathbf{m}_l)_i = \begin{cases} \frac{\mathrm{e}^{-i/a}}{\sqrt[4]{2\delta-1}}\cdot \mathrm{e}^{\frac{2\pi\mathrm{i}\cdot(i-1)\cdot(l-1)}{2\delta-1}} & \text{if } i \le \delta \\ 0 & \text{if } i > \delta \end{cases} \tag{17}$$

[7] Here $C \in \mathbb{R}^+$ is a fixed absolute constant.

for $1 \leq l \leq 2\delta-1$, and $1 \leq i \leq d$. This leads to blocks $M'_l \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$ with entries given by

$$(M'_l)_{i,j} := \begin{cases} (\mathbf{m}_i)_l \overline{(\mathbf{m}_i)}_{j+l-1} \;=\; \frac{\mathfrak{e}^{-(2l+j-1)/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{-\frac{2\pi\mathfrak{i}\cdot(i-1)\cdot(j-1)}{2\delta-1}} & \text{if } 1 \leq j \leq \delta-l+1 \\ 0 & \text{if } \delta-l+2 \leq j \leq 2\delta-l-1 \\ (\mathbf{m}_i)_{l+1} \overline{(\mathbf{m}_i)}_{l+j-2\delta+1} \;=\; \frac{\mathfrak{e}^{-(2l+j-2(\delta-1))/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{-\frac{2\pi\mathfrak{i}\cdot(i-1)\cdot(j-2\delta)}{2\delta-1}} & \text{if } 2\delta-l \leq j \leq 2\delta-1,\ l<\delta \\ 0 & \text{if } j>1, \text{ and } l=\delta \end{cases}.$$

We will now begin to bound the condition number of this block circulant matrix, M', by block diagonalizing it via (10).

Considering the entries of each $J_k \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$ from (11) results in two cases. First, suppose that $1 \leq j \leq \delta$. In this case one can see that

$$(J_k)_{i,j} \;=\; \frac{\mathfrak{e}^{(1-j)/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{\frac{-2\pi\mathfrak{i}\cdot(i-1)\cdot(j-1)}{2\delta-1}} \cdot \sum_{l=1}^{\delta-j+1} \mathfrak{e}^{-2l/a} \cdot \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k\cdot l}{d}}, \tag{18}$$

$$= \frac{\mathfrak{e}^{-(j+1)/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{\frac{-2\pi\mathfrak{i}\cdot(i-1)\cdot(j-1)}{2\delta-1}} \cdot \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k}{d}} \cdot \frac{1-\mathfrak{e}^{-2(\delta-j+1)/a}\cdot\mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k\cdot(\delta-j+1)}{d}}}{1-\mathfrak{e}^{-2/a}\cdot\mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k}{d}}}. \tag{19}$$

Second, suppose that $\delta+1 \leq j \leq 2\delta-1$. In this case one can see that

$$(J_k)_{i,j} \;=\; \frac{\mathfrak{e}^{-(j-2(\delta-1))/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{\frac{-2\pi\mathfrak{i}\cdot(i-1)\cdot(j-1)}{2\delta-1}} \cdot \sum_{l=2\delta-j}^{\delta-1} \mathfrak{e}^{-2l/a} \cdot \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k\cdot l}{d}}, \tag{20}$$

$$= \frac{\mathfrak{e}^{-(2(\delta+1)-j)/a}}{\sqrt{2\delta-1}} \cdot \mathfrak{e}^{\frac{-2\pi\mathfrak{i}\cdot(i-1)\cdot(j-1)}{2\delta-1}} \cdot \mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k(2\delta-j)}{d}} \cdot \frac{1-\mathfrak{e}^{-2(j-\delta)/a}\cdot\mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k\cdot(j-\delta)}{d}}}{1-\mathfrak{e}^{-2/a}\cdot\mathfrak{e}^{\frac{2\pi\mathfrak{i}\cdot k}{d}}}. \tag{21}$$

Let $F_\alpha \in \mathbb{C}^{\alpha\times\alpha}$ be the unitary $\alpha\times\alpha$ discrete Fourier transform matrix. Defining

$$s_{k,j} := \begin{cases} \mathfrak{e}^{-(j+1)/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k/d}\cdot\frac{1-\mathfrak{e}^{-2(\delta-j+1)/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k\cdot(\delta-j+1)/d}}{1-\mathfrak{e}^{-2/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k/d}} & \text{if } 1 \leq j \leq \delta \\ \mathfrak{e}^{-(2(\delta+1)-j)/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k(2\delta-j)/d}\cdot\frac{1-\mathfrak{e}^{-2(j-\delta)/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k\cdot(j-\delta)/d}}{1-\mathfrak{e}^{-2/a}\cdot\mathfrak{e}^{2\pi\mathfrak{i}\cdot k/d}} & \text{if } \delta+1 \leq j \leq 2\delta-1 \end{cases},$$

we now have that

$$J_k = \; F_{2\delta-1} \begin{pmatrix} s_{k,1} & 0 & \dots & 0 \\ 0 & s_{k,2} & 0 & \dots \\ 0 & 0 & \ddots & 0 \\ 0 & \dots & 0 & s_{k,2\delta-1} \end{pmatrix}. \tag{22}$$

Note that the condition number of J, and therefore of M', will be dictated by the singular values of these J_k matrices. Thus, we will continue by developing bounds for the singular values of each $J_k \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$.

The fact that $F_{2\delta-1}$ is unitary implies that

$$\min_{j\in[2\delta-1]} |s_{k,j}| \leq \sigma_{2\delta-1}(J_k) \leq \sigma_1(J_k) \leq \max_{j\in[2\delta-1]} |s_{k,j}| \tag{23}$$

for all $k \in [d]$. Thus, we will now devote ourselves to bounding the maximum and minimum values of $|s_{k,j}|$ from above and below, respectively, over all $k \in [d]$ and $j \in [2\delta-1]$. These bounds will then collectively yield an upper bound on the condition number of our block circulant measurement matrix M'. The following simple technical lemmas will be useful.

Lemma 1. *Let $x \in [2,\infty)$. Then, $1-\mathfrak{e}^{-1/x} \;>\; \frac{2-\mathfrak{e}^{1/x}}{x} \;\geq\; \frac{2-\sqrt{\mathfrak{e}}}{x} > \frac{7}{20\cdot x}$.*

Proof: Note that $1-\mathrm{e}^{-1/x} = \sum_{n=1}^{\infty}\frac{(-1)^{n+1}}{x^n n!} > \frac{1}{x}\cdot\left(2-\sum_{n=0}^{\infty}\frac{1}{x^n(n+1)!}\right) > \frac{2-\mathrm{e}^{1/x}}{x}$. Furthermore, the numerator is a monotonically increasing function of x. □

Lemma 2. *Let $a,b,c\in\mathbb{R}^+$, and $f:\mathbb{R}\to\mathbb{R}$ below. Then,*

(1) $f(x)=b\cdot\mathrm{e}^{-x/a}\left(1+c\cdot\mathrm{e}^{2x/a}\right)$ *has a unique global minimum at $x=-\frac{a}{2}\ln(c)$, and*
(2) $f(x)=b\cdot\mathrm{e}^{-x/a}\left(1-c\cdot\mathrm{e}^{2x/a}\right)$ *is monotonically decreasing.*

Proof: In either case we have that $f'(x)=-\frac{b}{a}\cdot\mathrm{e}^{-x/a}\pm\frac{bc}{a}\cdot\mathrm{e}^{x/a}$, and $f''(x)=\frac{b}{a^2}\cdot\mathrm{e}^{-x/a}\pm\frac{bc}{a^2}\cdot\mathrm{e}^{x/a}$. For (1) we have a single critical point at $x=-\frac{a}{2}\ln(c)$, which is a global minimum since $f''(x)>0\ \forall x\in\mathbb{R}$. For (2) we have $f'(x)<0$ for all $x\in\mathbb{R}$. □

Note that

$$|s_{k,j}| = \begin{cases} \mathrm{e}^{-(j+1)/a}\cdot\sqrt{\frac{1+\mathrm{e}^{-4(\delta-j+1)/a}-2\mathrm{e}^{-2(\delta-j+1)/a}\cos(2\pi\cdot[\delta-j+1]\cdot k/d)}{1+\mathrm{e}^{-4/a}-2\mathrm{e}^{-2/a}\cos(2\pi k/d)}} & \text{if } 1\le j\le\delta \\ \mathrm{e}^{-(2(\delta+1)-j)/a}\cdot\sqrt{\frac{1+\mathrm{e}^{-4(j-\delta)/a}-2\mathrm{e}^{-2(j-\delta)/a}\cos(2\pi\cdot[j-\delta]\cdot k/d)}{1+\mathrm{e}^{-4/a}-2\mathrm{e}^{-2/a}\cos(2\pi k/d)}} & \text{if } \delta+1\le j\le 2\delta-1 \end{cases}. \tag{24}$$

Fix $k\in[d]$. When $1\le j\le\delta$ we have

$$\max_{j\in[\delta]}|s_{k,j}| \le \max_{j\in[\delta]}\left(\mathrm{e}^{-(j+1)/a}\cdot\frac{1+\mathrm{e}^{-2(\delta+1-j)/a}}{1-\mathrm{e}^{-2/a}}\right) \le \frac{\mathrm{e}^{-2/a}(1+\mathrm{e}^{-2\delta/a})}{1-\mathrm{e}^{-2/a}}, \tag{25}$$

where the second inequality follows from part one of Lemma 2. When $\delta+1\le j\le 2\delta-1$ we have

$$\max_{j\in[2\delta-1]\setminus[\delta]}|s_{k,j}| \le \max_{j\in[2\delta-1]\setminus[\delta]}\left(\mathrm{e}^{-(2(\delta+1)-j)/a}\cdot\frac{1+\mathrm{e}^{-2(j-\delta)/a}}{1-\mathrm{e}^{-2/a}}\right) \le \frac{\mathrm{e}^{-3/a}(1+\mathrm{e}^{-2(\delta-1)/a})}{1-\mathrm{e}^{-2/a}}, \tag{26}$$

where the second inequality again follows from part one of Lemma 2. Finally, combining (25) and (26) one can see that

$$\sigma_1(J_k) \le \frac{\mathrm{e}^{-2/a}(1+\mathrm{e}^{-2\delta/a})}{1-\mathrm{e}^{-2/a}} < a\cdot\frac{\mathrm{e}^{-2/a}(1+\mathrm{e}^{-2\delta/a})}{2(2-\mathrm{e}^{2/a})} < a\cdot\frac{20\mathrm{e}^{-2/a}}{7} < 3a\cdot\mathrm{e}^{-2/a}, \tag{27}$$

where the second inequality follows from Lemma 1 with $a\in[4,\infty)$.

Turning our attention to the lower bound, we note that part two of Lemma 2 implies that

$$\min_{j\in[\delta]}|s_{k,j}| \ge \min_{j\in[\delta]}\left(\mathrm{e}^{-(j+1)/a}\cdot\frac{1-\mathrm{e}^{-2(\delta+1-j)/a}}{1+\mathrm{e}^{-2/a}}\right) \ge \frac{\mathrm{e}^{-(\delta+1)/a}(1-\mathrm{e}^{-2/a})}{1+\mathrm{e}^{-2/a}}. \tag{28}$$

Similarly, part two of Lemma 2 also ensures that

$$\min_{j\in[2\delta-1]\setminus[\delta]}|s_{k,j}| \ge \min_{j\in[2\delta-1]\setminus[\delta]}\left(\mathrm{e}^{-(2(\delta+1)-j)/a}\cdot\frac{1-\mathrm{e}^{-2(j-\delta)/a}}{1+\mathrm{e}^{-2/a}}\right) \ge \frac{\mathrm{e}^{-(\delta+1)/a}(1-\mathrm{e}^{-2/a})}{1+\mathrm{e}^{-2/a}}. \tag{29}$$

Combining (28) and (29) we see that

$$\sigma_{2\delta-1}(J_k) \ge \frac{\mathrm{e}^{-(\delta+1)/a}(1-\mathrm{e}^{-2/a})}{1+\mathrm{e}^{-2/a}} > \frac{7}{20a}\cdot\mathrm{e}^{-(\delta+1)/a}, \tag{30}$$

where the second inequality follows from Lemma 1 with $a\in[4,\infty)$. We are now equipped to prove the main theorem of this section.

Theorem 4. *Define $M' \in \mathbb{C}^{D\times D}$ via (17) with $a := \max\left\{4,\ \frac{\delta-1}{2}\right\}$. Then,*

$$\kappa\left(M'\right) \ < \ \max\left\{144\mathbb{e}^2,\ \frac{9\mathbb{e}^2}{4}\cdot(\delta-1)^2\right\}.$$

Proof: We have from (27) and (30) that

$$\kappa\left(M'\right) = \frac{\sigma_1\left(M'\right)}{\sigma_D\left(M'\right)} \ = \ \frac{\sigma_1\left(J\right)}{\sigma_D\left(J\right)} \ \leq \ \frac{\max_{k\in[d]}\sigma_1\left(J_k\right)}{\min_{k\in[d]}\sigma_{2\delta-1}\left(J_k\right)} \ < \ 9a^2\cdot\mathbb{e}^{(\delta-1)/a}. \tag{31}$$

Minimizing the rightmost upper bound as a function of a yields the stated result. □

Theorem 4 guarantees the existence of measurements which allow for the robust recovery of the phase difference vector $\mathbf{y} \in \mathbb{C}^D$ defined in (5). In the next three subsections we analyze the recovery of $\mathbf{x} \in \mathbb{C}^d$ from $\mathbf{y}$ via the techniques discussed in §3.

4.2. **A Recovery Guarantee for Flat Vectors.** As mentioned in §3, Algorithm 1 implicitly assumes that $\mathbf{x} \in \mathbb{C}^d$ does not contain any strings of $\delta-1$ consecutive entires with very small magnitudes (mod d). We will refer to such vectors as being "flat". More specifically, we will utilize the following more concrete definition.

Definition 3. *Let $m \in [d]$. A vector $\mathbf{u} \in \mathbb{C}^d$ will be called m-flat if its entries can be partitioned into at least $\left\lfloor\frac{d}{m}\right\rfloor$ contiguous blocks such that:*

1. *Every block contains either m or $m+1$ entries,*
2. *Every block contains at least one entry whose magnitude is $\geq \frac{\|\mathbf{u}\|_2}{2\sqrt{d}}$, and*
3. *All entries of $\mathbf{u}$ have magnitude $\leq \sqrt{\frac{3m+3}{2d}}\cdot\|\mathbf{u}\|_2$.*

Note that Algorithm 1 will always successfully recover $\left\lfloor\frac{(\delta-1)}{2}\right\rfloor$-flat vectors whenever $(M')^{-1}$ exists. To see why, it suffices to consider the main while-loop of Algorithm 2 (i.e., lines 3 through 10). In particular, line 6 will always succeed in computing the correct (relative) phase of the entry in question as long as $|x_j| > 0$. Furthermore, such a j will always be discovered in line 9 if $\mathbf{x}$ is $\left\lfloor\frac{(\delta-1)}{2}\right\rfloor$-flat. This observation leads us to the following theorem.

Theorem 5. *Let $M \in \mathbb{C}^{D\times d}$ be defined as in §4.1, and suppose that $\mathbf{x} \in \mathbb{C}^d$ is m-flat for some $m \leq \left\lfloor\frac{(\delta-1)}{2}\right\rfloor$. Then, Algorithm 1 will recover an $\tilde{\mathbf{x}} \in \mathbb{C}^d$ with*

$$\min_{\theta\in[0,2\pi]}\left\|\mathbf{x}-\mathbb{e}^{\mathbb{i}\theta}\tilde{\mathbf{x}}\right\|_2 = 0 \tag{32}$$

when given the noiseless input measurements $|M\mathbf{x}|^2 \in \mathbb{R}^D$. Furthermore, Algorithm 1 requires just $\mathcal{O}(\delta\cdot d\log d)$ operations in this case.

Proof: The recovery guarantee (32) follows from Theorem 4 together with the preceding paragraph. The runtime complexity of Algorithm 1 simplifies to $\mathcal{O}(\delta\cdot d\log d)$ operations when using the measurements defined in §4.1 because the matrix J ends up having a simple factorization (see (22)). □

Of course, not all vectors are flat. We remedy this defect in the next subsection.

4.3. **Flattening Arbitrary Vectors with High Probability.** Let $W \in \mathbb{C}^{d\times d}$ be the random unitary matrix

$$W := PFB, \tag{33}$$

where $P \in \{0,1\}^{d\times d}$ is a permutation matrix selected uniformly at random from the set of all $d \times d$ permutation matrices, F is the unitary $d \times d$ discrete Fourier transform matrix, and $B \in \{-1,0,1\}^{d\times d}$ is a random diagonal matrix with i.i.d. symmetric Bernoulli entries on its diagonal. For any given $m \in [d]$, one can naturally partition W into $\lfloor \frac{d}{m} \rfloor$ blocks of contiguous rows, each of cardinality either m or $m+1$. This defines the $\lfloor \frac{d}{m} \rfloor$ sub-matrices of W, $W_1, \ldots, W_{d-m\lfloor \frac{d}{m} \rfloor} \in \mathbb{C}^{(m+1)\times d}$ and $W_{d-m\lfloor \frac{d}{m} \rfloor+1}, \ldots, W_{\lfloor \frac{d}{m} \rfloor} \in \mathbb{C}^{m\times d}$, by

$$W = \begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_{\lfloor \frac{d}{m} \rfloor} \end{pmatrix}. \tag{34}$$

Note that each renormalized sub-matrix of W, $\sqrt{\frac{d}{m}} \cdot W_j$ for $j \in \left[\lfloor \frac{d}{m} \rfloor\right]$, is "almost" a random sampling matrix (13) times a random diagonal Bernoulli matrix. As a result, Theorems 1 and 2 suggest that each $\sqrt{\frac{d}{m}} \cdot W_j$ should behave like a JL(m,d,ϵ)-embedding of our signal $\mathbf{x}$ into $\mathbb{C}^m$ (or $\mathbb{C}^{m+1}$). If true, it would then be reasonable to expect that each block of m consecutive entries of $W\mathbf{x}$ should have roughly the same ℓ_2-norm as one another. This, in turn, suggests that the random unitary matrix W should effectively flatten $\mathbf{x}$ with high probability, especially when m is small.

Of course, there are several small difficulties that must be addressed before the argument above can be made rigorous. First, the rows of F contributing to $\sqrt{\frac{d}{m}} \cdot W_j$ are effectively independently sampled uniformly *without replacement* from the set of all rows of F by our choice of P. This means that Theorem 2 does not strictly apply in our situation since we can not select any row of F more than once. Secondly, some care must be taken in order to select the smallest value of m possible in (34), since $W\mathbf{x}$ will "become flatter" as m decreases. As a result, m will effectively provide a theoretical lower bound on the size of δ that one can utilize and still be guaranteed to accurately recover $W\mathbf{x}$ via our §3 techniques (recall also §4.2 above). We are now ready to begin proving our main result concerning W.

The following simple lemma will be used in order to help adapt Theorem 2 to the situation where the rows of F are sampled uniformly without replacement.

Lemma 3. *Let $m \in \mathbb{N}$ with $m \leq \sqrt{d}$. Independently draw $x_1, \ldots, x_m$ from $[d]$ uniformly at random with replacement. Then, $\mathbb{P}\left[|\{x_1, \ldots, x_m\}| = m\right] \geq 1/2$.*

Proof: A short induction argument establishes that

$$\mathbb{P}\left[|\{x_1, \ldots, x_m\}| = m\right] = \prod_{j=1}^{m-1} \left(1 - \frac{j}{d}\right) \geq 1 - \sum_{j=1}^{m-1} \frac{j}{d} = 1 - \frac{m^2 - m}{2d}. \tag{35}$$

The result now follows easily via algebraic manipulation. □

The following corollary of Theorem 2 now demonstrates that a random sampling matrix R' formed by sampling a subset of rows of size m uniformly at random from F will still be RIP($2s$,ϵ/C_1) with high probability.

Corollary 1. *Let $p \in (0,1)$. Form a random sampling matrix $R' \in \mathbb{C}^{m\times d}$ by independently sampling m rows from F uniformly without replacement. If the number of rows, m, satisfies both*

$$\sqrt{d} \geq m \geq C_3 \cdot \frac{s \ln^2(8s) \ln(8d) \ln(9m)}{\epsilon^2} \tag{36}$$

and

$$\sqrt{d} \geq m \geq C_4 \cdot \frac{s \log(2/p)}{\epsilon^2}, \tag{37}$$

then R' will be RIP($2s$,ϵ/C_1) with probability at least $1-p$.

Proof: Let $\mathcal{S} := \{x_1, \ldots, x_m\}$, where each $x_j \in [d]$ is selected independently and uniformly at random from $[d]$ (with replacement). Similarly, let $\mathcal{S}' \subset [d]$ be a subset of $[d]$ chosen uniformly at random from all subsets of $[d]$ with cardinality m (i.e., let $\mathcal{S}'$ contain m elements sampled independently and uniformly from $[d]$ without replacement). Furthermore, let E denote the event that the random sampling matrix whose rows from F are $x_1, \ldots, x_m$ is not RIP($2s$,ϵ/C_1). Finally, let E' denote the event that the random sampling matrix whose rows from F are the elements of $\mathcal{S}'$ is not RIP($2s$,ϵ/C_1). Applying Lemma 3 we can now see that

$$\mathbb{P}[E] \;\geq\; \mathbb{P}\left[E \;\middle|\; |\mathcal{S}| = m\right] \cdot \mathbb{P}\left[|\mathcal{S}| = m\right] \;=\; \mathbb{P}\left[E'\right] \cdot \mathbb{P}\left[|\mathcal{S}| = m\right] \;\geq\; \frac{1}{2} \cdot \mathbb{P}\left[E'\right]. \tag{38}$$

The stated result now follows from Theorem 2. □

We are now ready to prove that W will flatten the signal $\mathbf{x} \in \mathbb{C}^d$ with high probability provided that m can be chosen appropriately. We have the following theorem:

Theorem 6. *Let $W \in \mathbb{C}^{d\times d}$ be formed as per (33) for $d \geq 8$. Then, $W\mathbf{x} \in \mathbb{C}^d$ will be m-flat with probability at least $1 - \frac{1}{m}$ provided that $\sqrt{d} \geq m+1 \geq C_5 \cdot \ln^2(d) \cdot \ln^3(\ln d)$.*[8]

Proof: Our first goal will be to show that each $W_1, \ldots, W_{\lfloor \frac{d}{m} \rfloor}$ from (34) is a is a rescaled JL(m,d,1/2)-embedding of $\{\mathbf{x}\}$ into $\mathbb{C}^m$ (or $\mathbb{C}^{m+1}$). This will guarantee that each consecutive block of m (or $m+1$) entries of $W\mathbf{x}$ has roughly the same ℓ_2-norm.

To achieve this goal we will apply Theorem 1 to each $\sqrt{\frac{d}{m}} \cdot W_1, \ldots, \sqrt{\frac{d}{m}} \cdot W_{\lfloor \frac{d}{m} \rfloor}$ in order to show that each one embeds $\{\mathbf{x}\}$ into $\mathbb{C}^m$ (or $\mathbb{C}^{m+1}$) with probability at least $1 - \frac{1}{2d}$. The union bound will then imply that $\{\mathbf{x}\}$ is embedded by all the $\sqrt{\frac{d}{m}} \cdot W_j$ with probability at least $1 - \frac{1}{2m}$. This argument will go through as long as each $\sqrt{\frac{d}{m}} \cdot W_1 B^{-1}, \ldots, \sqrt{\frac{d}{m}} \cdot W_{\lfloor \frac{d}{m} \rfloor} B^{-1}$ is RIP($2s$,$1/2C_1$) for some $s \geq C_2 \cdot \ln(8d)$. Hence, we will now focus on determining the range of m which guarantees that all $\lfloor \frac{d}{m} \rfloor$ of these matrices are RIP($\lceil 2C_2 \cdot \ln(8d) \rceil$,$1/2C_1$).

To demonstrate that each $\sqrt{\frac{d}{m}} \cdot W_j B^{-1}$ is RIP($\lceil 2C_2 \cdot \ln(8d) \rceil$,$1/2C_1$) with probability at least $1 - \frac{1}{2d}$ one may apply Corollary 1 with m (or $m+1$) chosen as above (assuming $d \geq 8$). Another application of the union bound then establishes that all of $\sqrt{\frac{d}{m}} \cdot W_1 B^{-1}, \ldots, \sqrt{\frac{d}{m}} \cdot W_{\lfloor \frac{d}{m} \rfloor} B^{-1}$ will be RIP($\lceil 2C_2 \cdot \ln(8d) \rceil$,$1/2C_1$) with probability at least $1 - \frac{1}{2m}$. One final application of the union bound then establishes our first goal: All of $\sqrt{\frac{d}{m}} \cdot W_1, \ldots, \sqrt{\frac{d}{m}} \cdot W_{\lfloor \frac{d}{m} \rfloor}$ will be JL(m,d,1/2)-embeddings of $\{\mathbf{x}\}$ with probability at least $1 - \frac{1}{m}$.

[8] Here $C_5 \in \mathbb{R}^+$ is a fixed absolute constant.

To finish the proof, we now note that $W\mathbf{x}$ will be m-flat whenever all $\lfloor \frac{d}{m} \rfloor$ of the $\sqrt{\frac{d}{m}} \cdot W_j$ matrices are JL(m,d,1/2)-embeddings of $\{\mathbf{x}\}$. To see why, suppose that

$$\frac{1}{2}\| \mathbf{x} \|_2^2 \leq \frac{d}{m}\|W_j\mathbf{x}\|_2^2 \leq \frac{3}{2}\| \mathbf{x} \|_2^2.$$

This implies that $\frac{3m}{2d}\| \mathbf{x} \|_2^2 \geq \|W_j\mathbf{x}\|_2^2 \geq \frac{m}{2d}\|\mathbf{x}\|_2^2$, which can only happen if both of the following hold: (i) at least one entry of $W_j\mathbf{x}$ has magnitude at least $\frac{\|\mathbf{x}\|_2}{2\sqrt{d}} = \frac{\|W\mathbf{x}\|_2}{2\sqrt{d}}$, and (ii) all entires of $W_j\mathbf{x}$ have magnitude less than $\sqrt{\frac{3m+3}{2d}}\|\mathbf{x}\|_2 = \sqrt{\frac{3m+3}{2d}}\|W\mathbf{x}\|_2$. This proves the theorem. □

Theorem 6 now allows us to alter our measurements so that we can recover arbitrary vectors. We are now ready to prove Theorem 3.

4.4. **Proof of Theorem 3.** We set our measurement matrix $\tilde{M} \in \mathbb{C}^{D\times d}$ to be $\tilde{M} := MW$ where $M \in \mathbb{C}^{D\times d}$ is defined as in §4.1, and $W \in \mathbb{C}^{d\times d}$ is as defined as in (33). Theorem 6 guarantees that $W\mathbf{x}$ will be $m = \mathcal{O}(\ln^2(d)\cdot\ln^3(\ln d))$-flat with probability at least $1 - \frac{1}{C_5\cdot\ln^2(d)\cdot\ln^3(\ln d)}$ provided that d is sufficiently large. Furthermore, if $W\mathbf{x}$ is m-flat and $\delta \geq 2m+1$, then Theorem 5 guarantees that Algorithm 1 will recover an $\mathbf{x}' \in \mathbb{C}^d$ satisfying

$$\min_{\theta\in[0,2\pi]} \left\| W\mathbf{x} - \mathbb{e}^{\mathbb{i}\theta}\mathbf{x}' \right\|_2 = 0 \tag{39}$$

when given the noiseless input measurements $|MW\mathbf{x}|^2 \in \mathbb{R}^D$. Hence, choosing $\delta = \mathcal{O}(\ln^2(d)\cdot\ln^3(\ln d))$ allows us to recover $\mathbf{x}' = W\left(\mathbb{e}^{\mathbb{i}\phi}\mathbf{x}\right)$, for some unknown phase $\phi \in [0, 2\pi]$, with probability at least $1 - \frac{1}{C_5\cdot\ln^2(d)\cdot\ln^3(\ln d)}$.[9] We then set $\tilde{\mathbf{x}} = W^*\mathbf{x}'$.

Considering the runtime complexity, we note that $\mathbf{x}'$ can be obtained in $\mathcal{O}(\delta\cdot d\log d) = \mathcal{O}(d\cdot\ln^3(d)\cdot\ln^3(\ln d))$ operations by Theorem 5. Computing $W^*\mathbf{x}'$ can then be done in $\mathcal{O}(d\log d)$ operations via an inverse fast Fourier transform. The stated runtime complexity follows.

It is interesting to note that alternate constructions of flattening matrices, W, with fast inverse-matrix vector multiplies can also be created by using sparse Johnson-Lindenstrauss embedding matrices in the place of our Fourier-based matrices (see, e.g., [7]). Thus, one has several choices of matrices W to use in concert with a given block-circulant measurement matrix M in principle.

5. Empirical Evaluation

We now present numerical results demonstrating the efficiency and robustness of the phase retrieval algorithm 1. We test our algorithm on unit-norm i.i.d zero-mean complex random Gaussian test signals. To test noise robustness, we add i.i.d random Gaussian noise to the squared magnitude measurements at desired signal to noise ratios (SNRs); i.e.,

$$\mathbf{y} = |M\mathbf{x}|^2 + \mathbf{n}, \tag{40}$$

where $\mathbf{y} \in \mathbb{R}^D$ denotes the noisy measurement vector and the noise $\mathbf{n} \in \mathbb{R}^D$ is chosen to be i.i.d $\mathcal{N}(\mathbf{0}, \sigma^2\mathcal{I}_D)$. The variance σ^2 is chosen such that

$$\text{SNR (dB)} = 10\log_{10}\left(\frac{\|M\mathbf{x}\|_2^2}{D\,\sigma^2}\right).$$

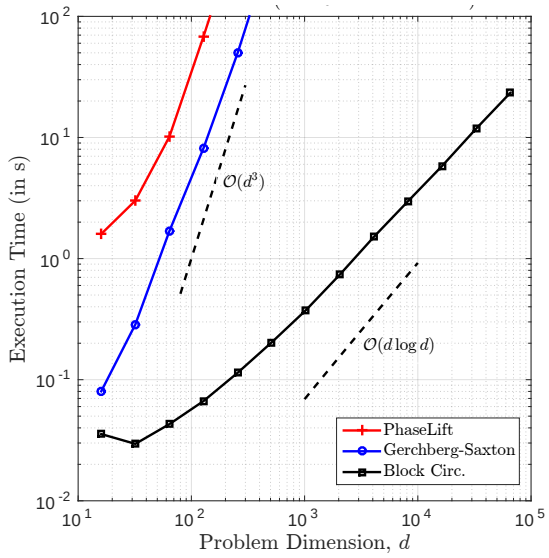
Errors in the recovered signal are also reported in dB with

$$\text{Error (dB)} = 10\log_{10}\left(\frac{\|\tilde{\mathbf{x}} - \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2}\right),$$

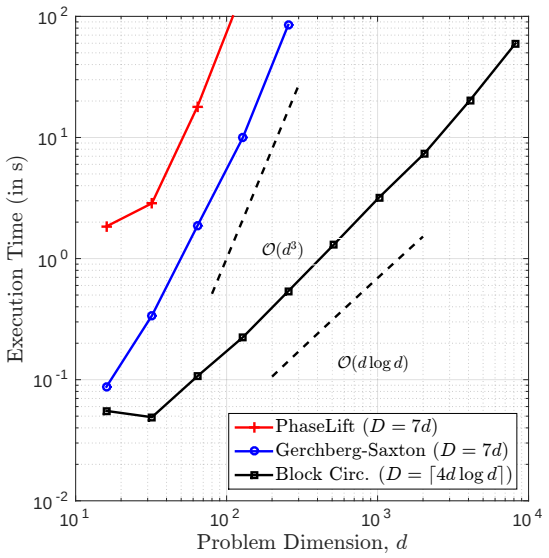
[9] The probability estimate in Theorem 3 follows immediately with $C = C_5$.

where $\tilde{\mathbf{x}}$ denotes the recovered signal. Matlab code used to generate the numerical results is freely available at [25].

We start by presenting numerical simulations demonstrating the efficiency of the block circulant construction introduced in this paper. In particular, we plot the execution time for solving the phase retrieval problem (averaged over 100 trials) in Figure 1. Simulations were performed on a laptop computer with an Intel® Core™i3-3120M processor, 4GB RAM and Matlab R2014b. For comparison, we also plot execution times for the Gerchberg–Saxton [20, 35] alternating projection and *PhaseLift* algorithms.[10] In each case, we recover a random complex Gaussian signal from noiseless magnitude measurements. We consider two cases: (i) Figure 1a, which plots the execution time for solving the phase retrieval problem using $5D$ measurements (suitable for high SNR applications), and (ii) Figure 1b, which plots the execution time when $4d \log d$ block circulant measurements are used (suitable for generic applications at a wide range of SNRs). Both plots confirm the log-linear execution time for implementing Algorithm 1. Moreover, it is clear that the block circulant construction introduced here is several orders of magnitude faster than equivalent methods, thereby allowing us to solve high-dimensional problems previously thought to be computationally infeasible.



(A) Execution time – Phase Retrieval from $5D$ measurements.



(B) Execution time – Phase Retrieval from $4d \log d$ measurements.

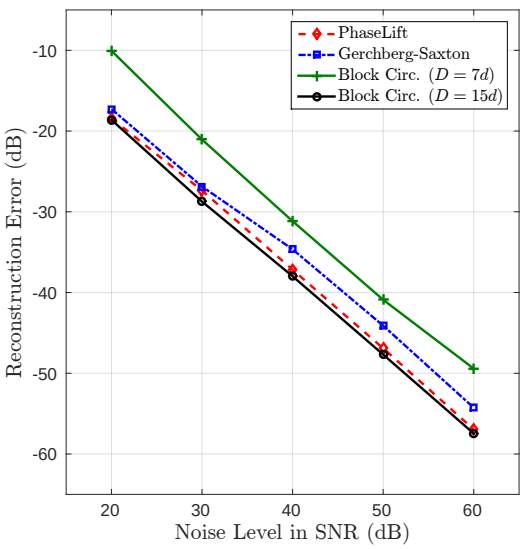
FIGURE 1. Computational Efficiency of the Block-Circulant Phase Retrieval Algorithm

We next demonstrate robustness to additive noise. Figure 2a plots the reconstruction error in recovering a $d = 64$ complex random Gaussian signal at different SNRs, with each data point computed as the average of 100 trials.[11] We include reconstruction results using the Gerchberg–Saxton alternating projection and *PhaseLift* algorithms for comparison. The deterministic windowed Fourier-like measurements introduced in §4.1 were used for the block circulant construction, while complex random Gaussian measurements were used for the other methods. We observe that all methods recover the underlying signal to the level of noise, although the block circulant construction requires approximately twice the number of measurements as the other methods.

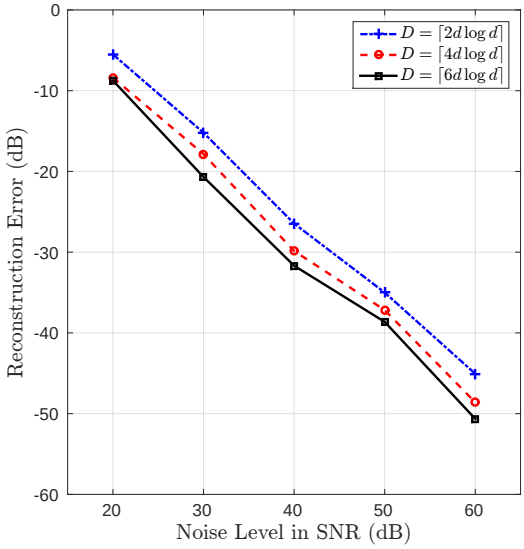
[10] Simulation results using *PhaseLift* and the Gerchberg–Saxton alternating projection algorithm use random complex Gaussian measurements.

[11] A few iterations of the alternating projection algorithm were used to post-process the reconstructions.

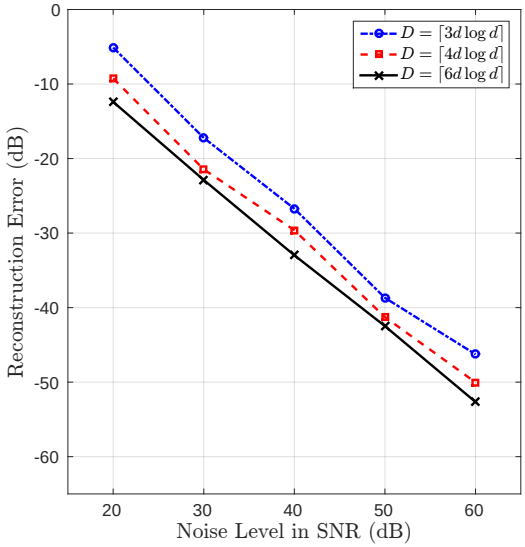
For completeness, we also plot the reconstruction error for a larger problem ($d = 2048$) in Figure 2b for three different number of measurements (D) and using the deterministic measurement construction. We note that the dimensions of this problem would make it be computationally intractable (on a conventional laptop or desktop machine implemented in Matlab) for methods such as Gerchberg–Saxton or *PhaseLift*.



(A) Robustness to Additive Noise ($d = 64$).



(B) Robustness to Additive Noise ($d = 2048$).



(C) Recovery using Random Masks ($d = 2048$).

FIGURE 2. Robustness to Additive Noise of the Block-Circulant Phase Retrieval Algorithm

To illustrate the flexibility of the measurement construction introduced in this paper, we also include results using random masks in Figure 2c. In particular, the entries of the block circulant measurement matrix are chosen to be i.i.d. standard complex Gaussian. Moreover, we may fix the

block length δ and collect oversampled measurements to improve the noise robustness of the recovery algorithm. In Figure 2c, the block length was fixed to be $\delta = \lceil 2d \log d \rceil$, oversampled measurements (by factors of $1.5, 2$ and 3) were used to recover the $d = 2048$ length i.i.d complex Gaussian test signal. The figure confirms that the random block-circulant construction also demonstrates robustness to additive noise across a wide range of SNRs, while the reconstruction accuracy improves with the oversampling factor.

Finally, Figure 3 plots the condition number of the system matrix used to solve for the phase differences (matrix M' in §2) for the deterministic block circulant measurement construction introduced in §4.1. The figure plots the condition number as a function of the block length δ for $d = 64$.[12] It confirms that the condition number scales as a small multiple of δ^2. The figure also includes a plot of the condition number when using random masks at an oversampling factor of 1.5.

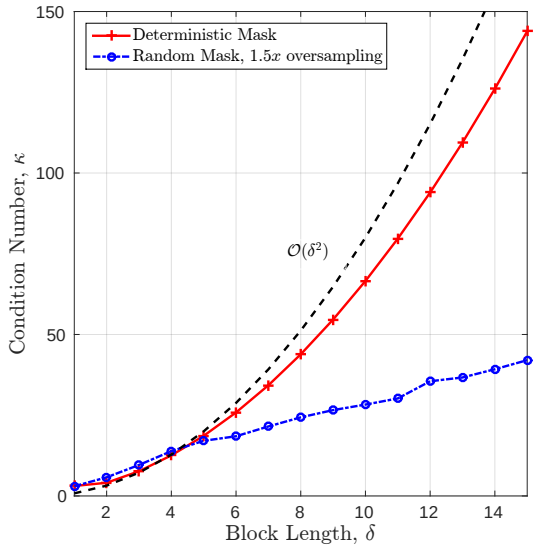


FIGURE 3. Well Conditioned Measurements – Condition Number as a Function of the Block Length δ

6. Sublinear-Time Phase Retrieval for Compressible Signals

In this section we briefly focus on the compressive phase retrieval setting, (see, e.g., [34, 36, 28, 41, 16, 37]), where one aims to approximate a sparse or compressible $\mathbf{x} \in \mathbb{C}^d$ using fewer magnitude measurements than required for the recovery of general $\mathbf{x}$. It is known that robust compressive phase retrieval for s-sparse vectors is possible using only $\mathcal{O}(s \log(d/s))$ magnitude measurements [16, 24]. In this section we prove that it is also possible to recover s-sparse vectors $\mathbf{x} \in \mathbb{C}^d$ up to an unknown phase factor in only $\mathcal{O}(s \log^6 d)$-time using $\mathcal{O}(s \log^5 d)$ magnitude measurements. Thus, we establish the first known nearly runtime-optimal (i.e., essentially linear-time in s) compressive phase retrieval recovery result. In particular, we prove the following theorem.

Theorem 7. *There exists a deterministic algorithm $\mathcal{A} : \mathbb{R}^D \to \mathbb{C}^d$ for which the following holds: Let $\epsilon \in (0, 1]$, $\mathbf{x} \in \mathbb{C}^d$ with d sufficiently large, and $s \in [d]$. Then, one can select a random*

[12] The condition number is independent of the problem dimension d and depends only on the block length δ.

measurement matrix $\tilde{M} \in \mathbb{C}^{D\times d}$ *such that*

$$\min_{\theta\in[0,2\pi]} \left\| \mathrm{e}^{\mathrm{i}\theta}\mathbf{x} - \mathcal{A}\left(|\tilde{M}\mathbf{x}|^2\right)\right\|_2 \;\leq\; \left\|\mathbf{x} - \mathbf{x}_s^{\mathrm{opt}}\right\|_2 + \frac{22\epsilon\left\|\mathbf{x}-\mathbf{x}^{\mathrm{opt}}_{(s/\epsilon)}\right\|_1}{\sqrt{s}} \tag{41}$$

is true with probability at least $1-\frac{1}{C\cdot\ln^2(d)\cdot\ln^3(\ln d)}$.[13] *Here* D *can be chosen to be* $\mathcal{O}\big(\frac{s}{\epsilon}\cdot\ln^3(\frac{s}{\epsilon})\cdot\ln^3\left(\ln\frac{s}{\epsilon}\right)\cdot\ln d\big)$. *Furthermore, the algorithm will run in* $\mathcal{O}\left(\frac{s}{\epsilon}\cdot\ln^4(\frac{s}{\epsilon})\cdot\ln^3\left(\ln\frac{s}{\epsilon}\right)\cdot\ln d\right)$*-time in that case.*[14]

We prove Theorem 7 by following the generic compressive phase retrieval recipe presented in [24]. Let $C \in \mathbb{C}^{m\times d}$ be any compressive sensing matrix with an associated sparse approximation algorithm $\Delta : \mathbb{C}^m \to \mathbb{C}^d$ (see, e.g., [8, 10, 39, 31, 5, 32, 33]), and let $P \in \mathbb{C}^{D\times m}$ be any phase retrieval matrix with an associated recovery algorithm $\Phi : \mathbb{R}^D \to \mathbb{C}^m$. Then, $\Delta\circ\Phi : \mathbb{R}^D \to \mathbb{C}^d$ will approximately recover compressible vectors $\mathbf{x} \in \mathbb{C}^d$ up to an unknown phase factor when provided with the magnitude measurements $|PC\mathbf{x}|$. That is, one may first use Φ to recover $\mathrm{e}^{\mathrm{i}\phi}(C\mathbf{x}) = C(\mathrm{e}^{\mathrm{i}\phi}\mathbf{x})$ for some unknown $\phi \in [0, 2\pi]$ from $|PC\mathbf{x}|$, and then use Δ to recover $\mathrm{e}^{\mathrm{i}\phi}\mathbf{x}$ from $C(\mathrm{e}^{\mathrm{i}\phi}\mathbf{x})$. If both Φ and Δ are efficient, the result will be an efficient sparse phase retrieval method.

Herein we will utilize Algorithm 1 as our phase retrieval method. Note that it's runtime is only $\mathcal{O}(d\log^4 d)$, making it optimal up to log factors (recall Theorem 3). For the compressive sensing method we will utilize the following algorithmic result from [23].

Theorem 8. *Let* $\epsilon \in (0, 1]$, $\sigma \in [2/3, 1)$, $\mathbf{x} \in \mathbb{C}^d$, *and* $s \in [d]$. *With probability at least* σ *the deterministic compressive sensing algorithm from* [23] *will output a vector* $\mathbf{z} \in \mathbb{C}^d$ *satisfying*

$$\|\mathbf{x}-\mathbf{z}\|_2 \;\leq\; \left\|\mathbf{x}-\mathbf{x}_s^{\mathrm{opt}}\right\|_2 + \frac{22\epsilon\left\|\mathbf{x}-\mathbf{x}^{\mathrm{opt}}_{(s/\epsilon)}\right\|_1}{\sqrt{s}} \tag{42}$$

when executed with random linear input measurements $\mathcal{M}\mathbf{x} \in \mathbb{C}^m$. *Here* $m = \mathcal{O}\left(\frac{s}{\epsilon}\cdot\ln\left(\frac{s/\epsilon}{1-\sigma}\right)\ln d\right)$ *suffices. The required runtime of the algorithm is* $\mathcal{O}\left(\frac{s}{\epsilon}\cdot\ln\left(\frac{s/\epsilon}{1-\sigma}\right)\ln\left(\frac{d}{1-\sigma}\right)\right)$ *in this case.*[15]

Theorem 7 now follows easily from Theorems 3 and 8.

References

[1] D. Achlioptas. Database-friendly random projections. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 274–281. ACM, 2001.

[2] B. Alexeev, A. S. Bandeira, M. Fickus, and D. G. Mixon. Phase retrieval with polarization. *SIAM Journal on Imaging Sciences*, 7(1):35–66, 2014.

[3] R. Balan, P. Casazza, and D. Edidin. On signal reconstruction without phase. *Applied and Computational Harmonic Analysis*, 20(3):345–356, 2006.

[4] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3):253–263, 2008.

[5] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[6] B. G. Bodmann and N. Hammen. Stable phase retrieval with low-redundancy frames. *Advances in Computational Mathematics*, pages 1–15, 2013.

[7] J. Bourgain and J. Nelson. Toward a unified theory of sparse dimensionality reduction in euclidean space. *arXiv preprint arXiv:1311.2542*, 2013.

[8] E. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, 52:489–509, 2006.

[9] E. Candes and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[13] Here $C \in \mathbb{R}^+$ is a fixed absolute constant.

[14] For the sake of simplicity, we assume $s = \Omega(\log d)$ when stating the measurement and runtime bounds above.

[15] For the sake of simplicity, we assume $s = \Omega(\log d)$ when stating the measurement and runtime bounds above.

[10] E. Candes and T. Tao. Near optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. on Information Theory*, 2006.
[11] E. J. Candes and X. Li. Solving quadratic equations via phaselift when there are about as many equations as unknowns. *Foundations of Computational Mathematics*, 14(5):1017–1026, 2014.
[12] E. J. Candes, T. Strohmer, and V. Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications on Pure and Applied Mathematics*, 66(8):1241–1274, 2013.
[13] S. Dasgupta and A. Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.
[14] L. Demanet and P. Hand. Stable optimizationless recovery from phaseless linear measurements. *Journal of Fourier Analysis and Applications*, 20(1):199–221, 2014.
[15] Y. Eldar, P. Sidorenko, D. Mixon, S. Barel, and O. Cohen. Sparse phase retrieval from short-time fourier measurements. *IEEE Signal Proc. Letters*, 22(5), 2015.
[16] Y. C. Eldar and S. Mendelson. Phase retrieval: Stability and recovery guarantees. *Applied and Computational Harmonic Analysis*, 36(3):473–494, 2014.
[17] M. Fickus, D. Mixon, A. Nelson, and Y. Wang. Phase retrieval from very few measurements. *Linear Algebra and its Applications*, 2014.
[18] S. Foucart and H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. Springer, 2013.
[19] P. Frankl and H. Maehara. The johnson-lindenstrauss lemma and the sphericity of some graphs. *Journal of Combinatorial Theory, Series B*, 44(3):355–362, 1988.
[20] R. Gerchberg and W. Saxton. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237–246, 1972.
[21] R. W. Harrison. Phase problem in crystallography. *J. Opt. Soc. Am. A*, 10(5):1046–1055, 1993.
[22] T. Heinosaari, L. Mazzarella, and M. M. Wolf. Quantum tomography under prior information. *Communications in Mathematical Physics*, 318(2):355–374, 2013.
[23] M. Iwen. Compressed sensing with sparse binary matrices: Instance optimal error guarantees in near-optimal time. *Journal of Complexity*, 30(1):1 – 15, 2014.
[24] M. Iwen, A. Viswanathan, and Y. Wang. Robust sparse phase retrieval made easy. *Preprint*, 2014.
[25] M. Iwen, Y. Wang, and A. Viswanathan. BlockPR: Matlab software for phase retrieval using block circulant measurement constructions and angular synchronization, version 1.0. `https://bitbucket.org/charms/blockpr`, 2014.
[26] W. Johnson and J. Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26:189–206, 1984.
[27] F. Krahmer and R. Ward. New and improved johnson-lindenstrauss embeddings via the restricted isometry property. *SIAM Journal on Mathematical Analysis*, 43(3):1269–1281, 2011.
[28] X. Li and V. Voroninski. Sparse signal recovery from quadratic measurements via convex programming. *SIAM Journal on Mathematical Analysis*, 45(5):3019–3033, 2013.
[29] J. Miao, T. Ishikawa, Q. Shen, and T. Earnest. Extending x-ray crystallography to allow the imaging of non-crystalline materials, cells, and single protein complexes. *Annu. Rev. Phys. Chem.*, 59:387–410, 2008.
[30] R. Millane. Phase retrieval in crystallography and optics. *J. Opt. Soc. Am. A*, 7(3):394–411, 1990.
[31] D. Needell and J. A. Tropp. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3), 2008.
[32] D. Needell and R. Vershynin. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Foundations of Computational Mathematics*, 9:317–334, 2009.
[33] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *IEEE Journal of Selected Topics in Signal Processing*, pages 310–316, 2010.
[34] H. Ohlsson, A. Yang, R. Dong, and S. Sastry. Cprl: An extension of compressive sensing to the phase retrieval problem. In *Proceedings of the 26th Conference on Advances in Neural Information Processing Systems*, pages 1376–1384, 2012.
[35] P. J. P. Netrapalli and S. Sanghavi. Phase retrieval using alternating minimization. In *Advances in Neural Information Processing Systems*, pages 2796–2804, 2013.
[36] P. Schniter and S. Rangan. Compressive phase retrieval via generalized approximate message passing. In *Proc. Allerton Conf. on Communication, Control, and Computing*, 2012.
[37] Y. Shechtman, A. Beck, and Y. C. Eldar. Gespar: Efficient phase retrieval of sparse signals. *IEEE transactions on signal processing*, 62(4):928–938, 2014.
[38] G. J. Tee. Eigenvectors of block circulant and alternating circulant matrices. *New Zealand Journal of Mathematics*, 36:195–211, 2007.

[39] J. Tropp and A. Gilbert. Signal recovery from partial information via orthogonal matching pursuit. *IEEE Trans. Info. Theory*, 53(12):4655–4666, Dec. 2007.
[40] A. Walther. The question of phase retrieval in optics. *Journal of Modern Optics*, 10(1):41–49, 1963.
[41] Y. Wang and Z. Xu. Phase retrieval for sparse signals. *Applied and Computational Harmonic Analysis*, 2014.

Edge Detection from Two-Dimensional Fourier Data using Gaussian Mollifiers

Anne Gelb Guohui Song Aditya Viswanathan Yang Wang

February 8, 2016

Abstract

This paper discusses the detection of edges from two-dimensional truncated Fourier spectral data. Compared to edge detection from pixel data, this is a more challenging problem since we seek accurate *local* information from a small number of often noisy *global* measurements. We propose a highly effective algorithm using a specific class of spectral mollifiers which converges *uniformly* to sharp peaks along the singular support of the function. We provide theoretical guarantees and numerical simulations to show that the resulting edge map is free of spurious edges and oscillations.

1 Introduction

The detection of jump discontinuities in piece-wise smooth functions is an important task in several areas of science and engineering. For example, many image and video processing operations such as segmentation and feature extraction rely on the accurate identification of edges in the underlying image (see for example [1, Chapter 10] for a discussion). Similarly, high-order methods for the numerical solution of PDEs often incorporate jump information when the solution is piece-wise smooth [2, Chapter 9]. Although edge detection is a non-trivial problem (especially when dealing with discrete and/or quantized data, and in the presence of noise), efficient and accurate algorithms such as the (W)ENO schemes, [3, 4] and the Canny edge detector, [5] exist for identifying edge locations when we start with *physical* space or *pixel* data. Certain applications, however, require that we extract edge information starting with *spectral* data. The most common example is magnetic resonance imaging (MRI), where the underlying physics of nuclear magnetic resonance implies that the MR scanner collects samples of the Fourier transform of the specimen being imaged. Identifying edges from such data is a significantly more challenging problem since we seek accurate *local* information from a small number of often noisy *global* measurements.

We begin by illustrating this problem in one dimension. Consider the piece-wise smooth test

function $f : [0, 1) \to \mathbb{R}$

$$f(x) = a(x)\sin(\pi x), \quad a(x) = \begin{cases} \frac{1}{2} & x \in \left[0, \frac{1}{4}\right) \\ 0 & x \in \left[\frac{1}{4}, \frac{1}{2}\right) \\ 1 & x \in \left[\frac{1}{2}, \frac{3}{4}\right) \\ -1 & x \in \left[\frac{3}{4}, 1\right). \end{cases} \tag{1.1}$$

The jump discontinuities in f are completely described by its associated *jump function*, $[f]$, defined as

$$[f](x) := \begin{cases} f(x^+) - f(x^-) & x \in (0, 1) \\ f(0^+) - f(1^-) & x = 0. \end{cases} \tag{1.2}$$

Given the first $2N + 1$ Fourier coefficients of f,

$$\hat{f}(k) = \int_0^1 f(x)e^{-2\pi ikx}dx, \quad k = -N, \ldots, N,$$

how do we identify the locations and values of its jump discontinuities, i.e., how do we approximate $[f]$? The naive approach would be to compute the $2N + 1$ mode Fourier partial sum approximation of f on an equispaced grid

$$S_N f(x_j) = \sum_{|k| \le N} \hat{f}(k)e^{2\pi ikx_j}, \quad x_j = \frac{j}{N}, \, j = 0, \ldots, N - 1,$$

followed by the application of a local differencing scheme such as the (undivided) forward difference operator

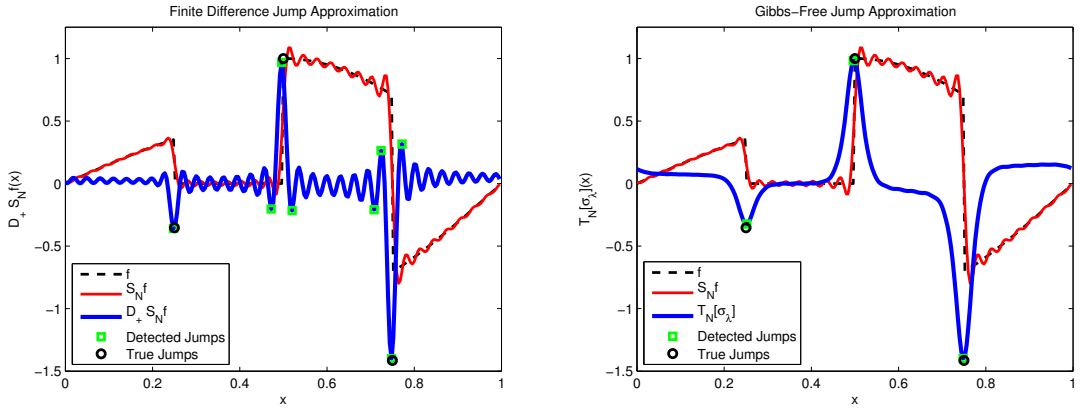
$$D_+ S_N f(x_j) = \begin{cases} S_N f(x_{j+1}) - S_N f(x_j) & j \in [0, N - 2] \\ S_N f(x_0) - S_N f(x_{N-1}) & j = N - 1. \end{cases} \tag{1.3}$$

The results using such an approach are shown in Fig. 1a, where $f, S_N f$ and $D_+ S_N f$ are plotted using dashed, solid (red) and solid(blue) lines respectively. A simple detector function of the form

$$\mathcal{E}(x_j) = \begin{cases} D_+ S_N f(x_j) & |D_+ S_N f(x_j)| > \left|D_+ S_N f(x_{(j\pm1)})\right|, \; D_+ S_N f(x_j) > \gamma \\ 0 & \text{else,} \end{cases} \tag{1.4}$$

is used to extract jump information from $D_+ S_N f$, where γ is a detection threshold. Since $S_N f$ (and consequently, $D_+ S_N f$) is a Fourier approximation of a piece-wise smooth function, it suffers from non-physical Gibbs oscillations. The largest of these (which are 9% of the corresponding jump height) are observed to be of the same order of the smallest jump in Fig. 1a. Unsurprisingly, the detector function (1.4) mistakes these oscillations for legitimate edges. Therefore, the challenge in detecting jump discontinuities from Fourier data is to distinguish these non-physical Gibbs oscillations from legitimate edges, or, to eliminate them entirely.

The latter approach was pursued by Cochran et. al. in [6], where the detection of jump discontinuities from one-dimensional truncated Fourier data using a special class of spectral



(a) (Undivided) Finite Differences

(b) Spectral Mollifiers

Figure 1: Jump Detection from one-dimensional truncated Fourier data. The jumps in test function (1.1) are detected using $2N+1$ Fourier modes, with $N = 32$ and a reconstruction grid of 250 points.

mollifiers was discussed. They proposed to approximate $[f]$ by a sequence of smooth pulses, $Q_N^\sigma(x) = \sum_{\xi\in\mathcal{K}} [f](\xi)\sigma_N(x-\xi)$, where $\mathcal{K}$ is the set of jump locations of f. For increasing N, Q_N^σ is increasingly concentrated at the jumps and σ is drawn from an appropriate class of functions so as to ensure Q_N^σ has no oscillations. Further, it was shown that a mollified Fourier derivative operator of the form

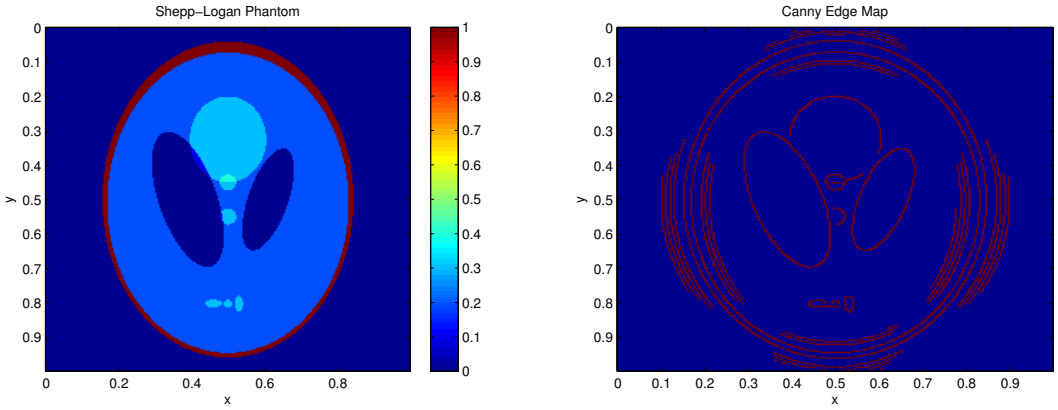
$$T_N[\sigma_{\lambda_N}](x) = 2\pi i \sum_{|k|\le N} k\,\widehat{\sigma_{\lambda_N}}(k)\,\hat{f}(k)e^{2\pi ikx} \tag{1.5}$$

converges uniformly to Q_N^σ for suitable choice of σ and sequence λ_N. A representative result of this method is shown in Fig. 1b, confirming the oscillation-free approximation qualities of $T_N[\sigma_{\lambda_N}]$. The edge detector function (1.4) applied to $T_N[\sigma_{\lambda_N}]$ now contains no spurious responses as was the case in Fig. 1a. We note that the jump approximation (1.5) is a specialization of the more general class of *concentration* edge detectors first introduced by Gelb and Tadmor in [7,8] and refined in [9–11]. These methods generally begin with a jump approximation of the form

$$S_N^\sigma[f](x) = 2\pi i \sum_{|k|\le N} \omega\left(\frac{k}{N}\right)\hat{f}(k)e^{2\pi ikx}, \tag{1.6}$$

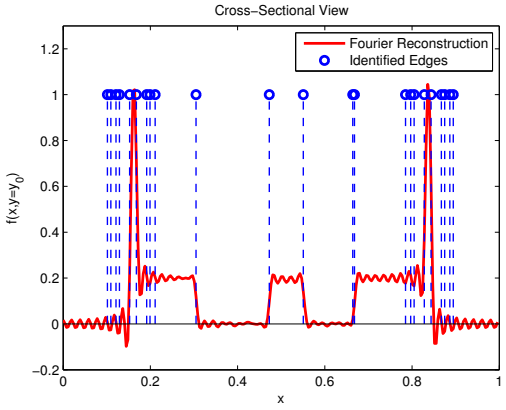
where ω defines a *concentration factor*. The corresponding physical-space *concentration kernels* are typically odd, suitably scaled, smooth and *oscillatory*. The oscillatory nature of these kernels makes it difficult to implement reliable edge detector functions, especially in the presence of noise.

Needless to say, the same issues exist in two dimensions, as illustrated in Fig. 2, where the edges of a Shepp-Logan brain phantom are identified using the Canny edge detector. A Fourier partial sum reconstruction on a 256×256 grid and using 50×50 Fourier modes serves as the input to the Canny edge detector. Fig. 2b plots the generated edge map while Fig. 2c shows a cross-section at the center of the image. The identified edges and the Fourier reconstruction along this cross-section are plotted using dashed and solid lines respectively. The comments and



(a) Shepp-Logan Phantom

(b) Edge Map



(c) Cross-Section

Figure 2: The Canny edge detector applied to a 50×50 mode partial sum Fourier reconstruction of the Shepp-Logan phantom on a 256×256 grid.

observations regarding the Gibbs phenomena in Fig. 1 apply here too. Our objective in this paper is to extend the one-dimensional framework introduced in [6] to the detection of edges from two-dimensional truncated and noisy Fourier data.

It is appropriate at this point to mention other related approaches to this problem and their relative advantages and disadvantages. We start with popular pixel-space edge detectors such as the Sobel, Prewitt or Marr-Hildreth edge detectors (see [1, Chapter 10] for a review) as well as more specialized algorithms such as the Canny edge detector [5]. As mentioned previously, these pixel- space approaches suffer from the tendency to mistake Gibbs oscillations for edges when applied to Fourier data. The method proposed here is more closely related to the two-dimensional *concentration kernel* approaches discussed in [12] and [13]. [12] uses statistical hypothesis testing methods to distinguish true edges from Gibbs oscillations, while [13] uses regularized bump functions and rotation-based post-processing operations to identify edges. The main contribution of this paper is the use of a specific form of spectral mollifier (and associated parameters) as well as a rigorous analysis of the same, demonstrating the oscillation-

free nature of the resulting edge approximation. We note that this framework can be combined with any other post-processing procedures, including Canny-type hysteresis edge tracking.

The rest of this paper is organized as follows: §2 introduces our two-dimensional edge detection scheme. A rigorous analysis examining the convergence of the scheme and confirming the absence of oscillaitons in the approximation is presented §3. §4 provides numerical results, including comparisons to pixel data methods such as the Canny edge detector and existing Fourier data schemes such as the concentration method. Performance in the presence of noise is also examined. Some concluding remarks and future directions are presented in §5.

2 Two Dimensional Edge Detection using Gaussian Mollifiers

We first give a brief introduction to the problem of detecting edges from 2-D Fourier data. Suppose $f : \mathbb{R}^2 \to \mathbb{R}$ is a piece-wise smooth and compactly supported on $[0, 1]^2$. We are given its finite Fourier data: $\hat{f}(\boldsymbol{z})$ for $\boldsymbol{z} = (z_1, z_2) \in S_N := [-N, N]^2 \cap \mathbb{Z}^2$, where

$$\hat{f}(\boldsymbol{z}) = \int_{(x,y)\in\mathbb{R}^2} f(x, y)\mathrm{e}^{-2\pi i z_1 x}\mathrm{e}^{-2\pi i z_2 y} dxdy.$$

We would like to identify all of its discontinuities in $[0, 1]^2$ and the corresponding jump heights that will be defined below.

We next present some assumptions on the function f and define the jump heights at the discontinuities. We will assume the set Γ of all the discontinuities consists of a few finite and disjoint smooth curves. In particular, we could write all the discontinuities in the following two ways:

$$(\alpha_j(y), y), \quad j = 1, 2, \ldots, N_y, \quad , y \in \mathbb{R},$$

and

$$(x, \bar{\alpha}_j(x)), \quad j = 1, 2, \ldots, M_x, \quad x \in \mathbb{R},$$

where N_y is a finite number for all but finitely many y's and M_x is a finite number for all but finitely many x's. We would also assume M_x and N_y are uniformly bounded for all $x \in \mathbb{R}$ and all $y \in \mathbb{R}$. A simple illustration is shown in Figure 3. Since we assume the discontinuities are smooth curves, both $\alpha_j(y)$ and $\bar{\alpha}_j(x)$ are smooth functions locally by the Implicit Function Theorem for almost all y's and for almost all x's respectively. Let f_x and f_y denote the partial derivatives of f at points other than the discontinuities. Note that both f_x and f_y are again piece-wise smooth with the same discontinuities of f. Let

$$[f]_1(x, y) = f(x+, y) - f(x-, y), \quad \text{and} \quad [f]_2(x, y) = f(x, y+) - f(x, y-), \quad (x, y) \in \mathbb{R}^2.$$

We point out that when they are different, one of them must be zero. In particular, $[f]_1(\alpha_j(y), y) = [f]_2(\alpha_j(y), y)$ for all but y's with $\alpha_j'(y) = 0$ or ∞. Consequently, we define the jump height $[f](x, y)$ be either one of them when they are the same, and the nonzero one if they are different.

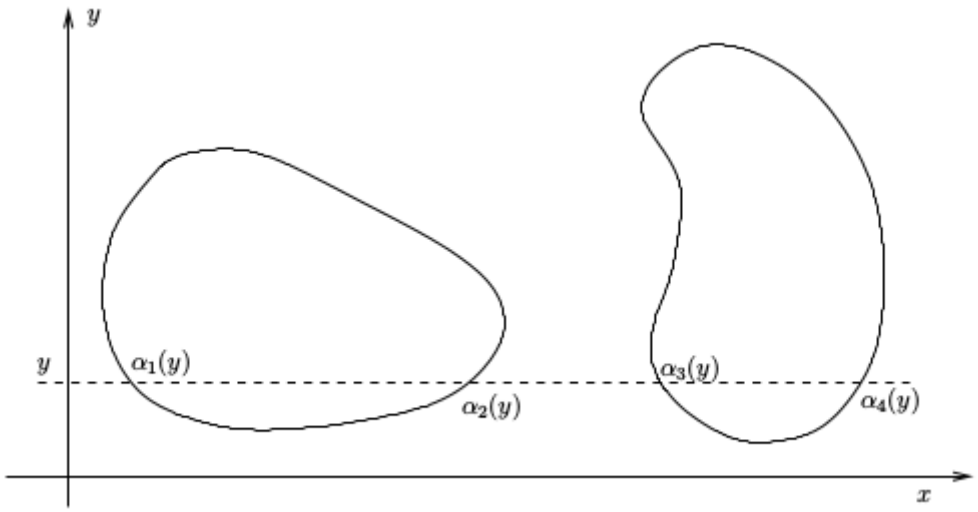


Figure 3: Edge Detection in two dimensions — Principle

We next introduce our edge detector by using the spectral Gaussian mollifiers. For $\lambda > 0$, we define

$$I_{N,\lambda}(x,y) = -2\pi i \sum_{\boldsymbol{z}\in S_N} \hat{f}(\boldsymbol{z}) z_1 \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{\lambda^2}} \mathrm{e}^{2\pi i(z_1 x + z_2 y)} \tag{2.1}$$

and

$$J_{N,\lambda}(x,y) = -2\pi i \sum_{\boldsymbol{z}\in S_N} \hat{f}(\boldsymbol{z}) z_2 \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{\lambda^2}} \mathrm{e}^{2\pi i(z_1 x + z_2 y)}.$$

We will use the following function to detect the edges of f:

$$E_{N,\lambda}(x,y) = \frac{1}{\sqrt{\pi}\lambda} \left[I_{N,\lambda}^2(x,y) + J_{N,\lambda}^2(x,y) \right]^{1/2} \mathrm{sgn}(I_{N,\lambda}(x,y)), \quad (x,y) \in \mathbb{R}^2, \tag{2.2}$$

where $\mathrm{sgn}(t) = 1$ if $t \geq 0$ and $\mathrm{sgn}(t) = -1$ otherwise.

We remark that without the Gaussian mollifier (i.e., $\lambda = \infty$), $I_{N,\lambda}(x,y)$ and $J_{N,\lambda}(x,y)$ would reduce to the partial derivatives of $f(\boldsymbol{x})$, which would yield spikes at the edges in addition to non-physical Gibbs oscillations in their vicinity. We will show in next section that with suitably chosen λ, the function $E_{N,\beta}(x,y)$ in (2.2) is a robust and accurate edge detector.

3 Convergence Analysis

We will present in this section the convergence analysis of the edge detector $E_{N,\lambda}(x,y)$ in (2.2). Specifically, we will present how to choose the parameter λ such that

(1) When (x,y) is away from the edge curves of f, the value of $E_{N,\lambda}(x,y)$ is close to zero.

(2) When (x,y) is on the edge curves of f, the value of $E_{N,\lambda}(x,y)$ is an approximation of the jump height $[f](x,y)$.

(3) The function $E_{N,\lambda}(x,y)$ behaves like sharp "mountains" rather than some oscillated peaks around the edges. That is, the Gibbs oscillation is controlled.

(4) The edge detector $E_{N,\lambda}(x,y)$ is robust with respect to small perturbations/noises on the spectral data $\hat{f}(\boldsymbol{z})$.

We point out that the convolution of f and a Gaussian function is an important resource for locating the edges of f, since the partial derivatives of the convolution would show some singular behaviors (sharp "mountains") around the edges. To this end, we would define this convolution and study the relation between its partial derivatives and our edge detector. We consider the following Gaussian function

$$\phi(x,y) = \pi \mathrm{e}^{-\pi^2(x^2+y^2)}, \quad (x,y) \in \mathbb{R}^2,$$

and for $\lambda \in \mathbb{R}$, let

$$\phi_\lambda(x,y) = \lambda^2 \phi(\lambda x, \lambda y), \quad (x,y) \in \mathbb{R}^2. \tag{3.1}$$

We then convolve f with ϕ_λ:

$$F_\lambda(x,y) = (f * \phi_\lambda)(x,y) = \int_{(s,t)\in\mathbb{R}^2} f(s,t)\phi_\lambda(x-s, y-t)dsdt, \quad (x,y) \in \mathbb{R}^2. \tag{3.2}$$

We will next focus on deriving estimates of the edge detector $I_{N,\lambda}$. The estimates of $J_{N,\lambda}$ could be obtained in a similar way. We shall first show a relation between $I_{N,\lambda}$ and the partial derivative $\frac{\partial F_\lambda}{\partial x}$. To this end, for $(x,y) \in \mathbb{R}^2$ we let

$$Q_\lambda(x,y) = \sum_{\boldsymbol{z}\in\mathbb{Z}^2} \frac{\partial F_\lambda(x+z_1, y+z_2)}{\partial x}, \tag{3.3}$$

and

$$B_N(x,y) = 2\pi i \sum_{\boldsymbol{z}\in\mathbb{Z}^2\setminus S_N} \hat{f}(\boldsymbol{z}) z_1 \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{\lambda^2}} \mathrm{e}^{2\pi i(z_1 x + z_2 y)}. \tag{3.4}$$

Proposition 3.1 *For $(x,y) \in \mathbb{R}^2$, there holds*

$$I_{N,\lambda}(x,y) = Q_\lambda(x,y) + B_N(x,y).$$

Moreover, there exists a positive constant c such that for any $(x,y) \in \mathbb{R}^2$ and $N \in \mathbb{N}$

$$|I_{N,\lambda}(x,y) - Q_\lambda(x,y)| \le c\lambda^3 \mathrm{e}^{-\frac{3N^2}{2\lambda^2}}.$$

Proof: We will prove the equality by a direct computation. To this end, we define the shift of the partial derivatives

$$g_{(x,y)}(s,t) = \frac{\partial F_\lambda(x+s, y+t)}{\partial x}, \quad (s,t) \in \mathbb{R}^2.$$

It follows that

$$Q_\lambda(x,y) = \sum_{\boldsymbol{z}\in\mathbb{Z}^2} g_{(x,y)}(\boldsymbol{z}).$$

On the other hand, a direct computation yields that $\hat{g}_{(x,y)}(\boldsymbol{\xi}) = -2\pi i \hat{f}(\boldsymbol{\xi})\xi_1 \mathrm{e}^{-\frac{\|\boldsymbol{\xi}\|^2}{\lambda^2}} \mathrm{e}^{2\pi i(\xi_1 x+\xi_2 y)}$. By the Poisson summation formula,

$$Q_\lambda(x,y) = \sum_{\boldsymbol{z}\in\mathbb{Z}^2} \hat{g}_{(x,y)}(\boldsymbol{z}) = \sum_{\boldsymbol{z}\in\mathbb{Z}^2} -2\pi i \hat{f}(\boldsymbol{z})\xi_1 \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{\lambda^2}} \mathrm{e}^{2\pi i(z_1 x+z_2 y)}$$

This combined with the definition of $I_{N,\beta}$ in (2.1) and the definition of B_N in (3.4) implies the desired equality.

We next show the inequality. It is enough to show $B_N(x,y)$ is bounded by the right hand side of the inequality. Since $f \in L^2[0,1]$, there exists a positive constant c_0 such that $|\hat{f}(\boldsymbol{z})| \le c_0$. It follows from (3.4) that

$$|B_N(x,y)| \le 2\pi c_0 \sum_{\boldsymbol{z}\in\mathbb{Z}^2\setminus S_N} z_1 \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{\lambda^2}} = 2\pi c_0 \sum_{z_1>N} z_1 \mathrm{e}^{-\frac{z_1^2}{\lambda^2}} \sum_{z_2>N} \mathrm{e}^{-\frac{z_2^2}{\lambda^2}}.$$

It is direct to observe that $\sum_{z_1>N} z_1 \mathrm{e}^{-\frac{z_1^2}{\lambda^2}} \le \int_N^\infty t\mathrm{e}^{-\frac{t^2}{\lambda^2}} dt = \frac{\lambda^2}{2}\mathrm{e}^{-\frac{N^2}{\lambda^2}}$. Moreover, by using the polar coordinates, it follows from a direct computation that $\sum_{z_2>N} \mathrm{e}^{-\frac{z_2^2}{\lambda^2}} \le \frac{\sqrt{\pi}}{2}\lambda \mathrm{e}^{-\frac{N^2}{2\lambda^2}}$. Substituting these two estimates into the above inequality, we have

$$|B_N(x,y)| \le \frac{\pi^{3/2}}{2} c_0 \lambda^3 \mathrm{e}^{-\frac{3N^2}{\lambda^2}},$$

which implied the desired inequality. □

We point out that we could choose appropriate λ depending on N such that $B_N(x,y)$ converges to zero uniformly, which avoids the Gibbs oscillation in the edge detectors. More details will be shown in later results.

We shall next analyze the behavior of Q_λ. In particular, we will show that Q_λ has peaks at the edges by using its relation with the partial derivative $\frac{\partial F_\lambda}{\partial x}$. To this end, we first present a direct computation of $\frac{\partial F_\lambda}{\partial x}$. We let

$$\tilde{I}_\lambda(x,y) = \int_{\mathbb{R}} \sum_{j=1}^{N_t} [f]_1(\alpha_j(t),t)\, \phi_\lambda(x-\alpha_j(t), y-t) dt, \tag{3.5}$$

and

$$H_\lambda(x,y) = \int_{(s,t)\in\mathbb{R}^2} f_x(s,t)\phi_\lambda(x-s,y-t) ds dt. \tag{3.6}$$

We have the following result of the partial derivative $\frac{\partial F_\lambda}{\partial x}$.

Proposition 3.2 *For any $(x,y) \in \mathbb{R}^2$, there holds that*

$$\frac{\partial F_\lambda}{\partial x}(x,y) = \tilde{I}_\lambda(x,y) + H_\lambda(x,y).$$

Moreover, if the edge curves are at least ϵ away from the boundary of $[0,1]^2$, that is, $\sqrt{(x-x^*)^2+(y-y^*)^2} \geq \epsilon$ for all $(x,y) \in [0,1]^2$ with either $x \in \{0,1\}$ or $y \in \{0,1\}$ and for all (x^*,y^*) on the edge curves, then there exists a positive constant c such that for $(x,y) \in [0,1]^2$

$$|Q_\lambda(x,y) - \tilde{I}_\lambda(x,y)| \leq c\pi\lambda^2 \mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + c\pi\lambda^2 \frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \|f_x\|_\infty.$$

Proof: We first show the equality about the decomposition of $\frac{\partial F_\lambda}{\partial x}$. From the definition of ∂F_λ in (3.2), we have

$$\frac{\partial F_\lambda}{\partial x}(x,y) = \int_{(s,t)\in\mathbb{R}^2} f(s,t) \frac{\partial \phi_\lambda}{\partial x}(x-s, y-t) ds dt.$$

A direct calculation of $\frac{\partial \phi_\lambda}{\partial x}$ from (3.1) yields that

$$\frac{\partial F_\lambda}{\partial x}(x,y) = \lambda^3 \int_{t\in\mathbb{R}} \left[\int_{s\in\mathbb{R}} f(s,t) \phi_x(\lambda(x-s), \lambda(y-t)) ds \right] dt.$$

Note that for $t \in \mathbb{R}$, $f(\cdot,t)$ has discontinuities $\alpha_j(t)$ for $1 \leq j \leq N_t$. For simplicity of presentation, we let $\alpha_0(t) = -\infty$ and $\alpha_{N_t+1}(t) = \infty$. It follows that

$$\frac{\partial F_\lambda}{\partial x}(x,y) = \lambda^3 \int_{t\in\mathbb{R}} \left[\sum_{j=1}^{N_t+1} \int_{\alpha_{j-1}(t)}^{\alpha_j(t)} f(s,t) \phi_x(\lambda(x-s), \lambda(y-t)) ds \right] dt.$$

Apply integration by parts and we have

$$\frac{\partial F_\lambda}{\partial x}(x,y) = \lambda^2 \int_{t\in\mathbb{R}} \left[-\sum_{j=1}^{N_t+1} f(s,t) \phi(\lambda(x-s), \lambda(y-t)) \Big|_{\alpha_{j-1}(t)}^{\alpha_j(t)} + \int_{\mathbb{R}} f_x(s,t) \phi(\lambda(x-s), \lambda(y-t)) ds \right] dt.$$

The desires equality follows from a direct calculation from the above equality.

We next estimate the difference of $Q_\lambda(x,y)$ and $\tilde{I}_\lambda(x,y)$. It follows from the definition of Q_λ in (3.3) and the equality shown above that

$$|Q_\lambda(x,y) - \tilde{I}_\lambda(x,y)| \leq \sum_{\boldsymbol{z}\neq 0} |\tilde{I}_\lambda(x+z_1, y+z_2)| + \sum_{\boldsymbol{z}\in\mathbb{Z}^2} |H_\lambda(x+z_1, y+z_2)|. \tag{3.7}$$

We will estimate the two terms in the right hand side of the above inequality separately.

We start with an estimate of first term. Note that both N_t and $[f]_1(\lambda_j(t), t)$ are uniformly bounded for all t. It follows from the definition of $\tilde{I}_\lambda$ in (3.5) that there exists a positive constant c_0 such that

$$\sum_{\boldsymbol{z}\neq 0} |\tilde{I}_\lambda(x+z_1, y+z_2)| \leq c_0 \sum_{\boldsymbol{z}\neq 0} \int_{\mathbb{R}} \phi_\lambda(x+z_1-\alpha_j(t), y+z_2-t) dt.$$

Note that when $\boldsymbol{z} \neq 0$, the point $(x+z_1, y+z_2)$ is not in $[0,1]^2$. By assumption, it is at least ϵ away from the edge curves. In particular, when $\boldsymbol{z} \in E := \{-1,0,1\}^2 \backslash (0,1)$, we have $\left((x+z_1-\alpha_j(t))^2 + (y+z_2-t)^2\right)^{1/2} \geq \epsilon$. On the other hand side, when $|z_1| \geq 2$, the point $(x+z_1, y+z_2)$ is at least $|z_1|-1$ away from the edge curves. When $|z_2| \geq 2$, the point

$(x+z_1, y+z_2)$ is at least $|z_2|-1$ away from the edge curves. Substituting these estimates into ϕ_λ as in (3.1) yields that

$$\begin{aligned}
&\sum_{\boldsymbol{z}\neq 0} |\tilde{I}_\lambda(x+z_1, y+z_2)| \\
\leq\ & \sum_{\boldsymbol{z}\in E} |\tilde{I}_\lambda(x+z_1, y+z_2)| + \sum_{z_1\in\mathbb{Z}, |z_2|\geq 2} |\tilde{I}_\lambda(x+z_1, y+z_2)| + \sum_{|z_1|\geq 2, z_2\in\mathbb{Z}} |\tilde{I}_\lambda(x+z_1, y+z_2)| \\
\leq\ & 5c_0\pi\lambda^2 \mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + 4c_0\pi\lambda^2 \frac{1}{1-\mathrm{e}^{-\pi^2\lambda^2}} \frac{\mathrm{e}^{-\pi^2\lambda^2}}{1-\mathrm{e}^{-\pi^2\lambda^2}},
\end{aligned}$$

which combined with (3.7) implies

$$|Q_\lambda(x,y) - \tilde{I}_\lambda(x,y)| \leq 5c_0\pi\lambda^2 \mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + 4c_0\pi\lambda^2 \frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \sum_{\boldsymbol{z}\in\mathbb{Z}^2} |H_\lambda(x+z_1, y+z_2)|.$$

To show the desired result on $|Q_\lambda(x,y) - \tilde{I}_\lambda(x,y)|$, it remains to prove $\sum_{\boldsymbol{z}\in\mathbb{Z}^2} |H_\lambda(x+z_1, y+z_2)| \leq \|f_x\|_\infty$. Note that $f_x(s,t) = 0$ when $(s,t) \notin [0,1]^2$. It is direct to observe from the definition of H_λ in (3.6) that for any $\boldsymbol{z} \in \mathbb{Z}^2$,

$$\begin{aligned}
|H_\lambda(x+z_1, y+z_2)| &= \left| \int_{(s,t)\in[0,1]^2} f_x(s,t)\phi_\lambda(x+z_1-s, y+z_2-t) ds dt \right| \\
&\leq \|f_x\|_\infty \int_{(s,t)\in[0,1]^2} \phi_\lambda(x+z_1-s, y+z_2-t) ds dt,
\end{aligned}$$

It implies

$$\begin{aligned}
\sum_{\boldsymbol{z}\in\mathbb{Z}^2} |H_\lambda(x+z_1, y+z_2)| &\leq \|f_x\|_\infty \sum_{\boldsymbol{z}\in\mathbb{Z}^2} \int_{(s,t)\in[0,1]^2} \phi_\lambda(x+z_1-s, y+z_2-t) ds dt \\
&= \|f_x\|_\infty \int_{(u,v)\in\mathbb{R}^2} \phi_\lambda(u,v) du dv \\
&= \|f_x\|_\infty,
\end{aligned}$$

which finishes the proof. □

We will continue with the analysis of $\tilde{I}_\lambda$. In particular, we will show that it is concentrated around the edges of f.

Proposition 3.3 *(i) When (x,y) is at least ϵ away from the edges, that is, $dist((x,y),\Gamma) \geq \epsilon$, there exists a positive constant c such that*

$$|\tilde{I}_\lambda(x,y)| \leq c\lambda^2 \mathrm{e}^{-\pi^2\lambda^2\epsilon^2}.$$

(ii) When (x^*, y^*) is on the edge, that is, $x^* = \alpha_{j_0}(y^*)$ for some $j_0 \in \mathbb{N}$, if there exists a $\epsilon > 0$ such that $d((\alpha_j(y), y), (\alpha_{j_0}(y^*), y^*)) = \sqrt{(\alpha_j(y) - \alpha_{j_0}(y^*))^2 + (y-y^*)^2} \geq \epsilon$ for all $j \neq j_0$

and $y \in [0,1]$, then there exists a positive constant c such that

$$\left|\tilde{I}_\lambda(x^*,y^*) - [f]_1(x^*,y^*)\frac{\sqrt{\pi}\lambda}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}}\right| \le c\big(\lambda \mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + \lambda^2\mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + (\lambda\epsilon)^2 + (\lambda\epsilon)^4\big).$$

Proof: (i) Note that both N_t and the jump heights $[f]_1(\alpha_j(t),t)$ are uniformly bounded. Since f is compactly supported on $[0,1]$, it follows from the definition of tI_λ in (3.5) that there exists a positive constant c_0 such that

$$|\tilde{I}_\lambda(x,y)| \le c_0 \int_0^1 \phi_\lambda(x-\alpha_j(t), y-t)dt.$$

By the definition of ϕ_λ in (3.1),

$$|\tilde{I}_\lambda(x,y)| \le c_0 \int_0^1 \lambda^2 \mathrm{e}^{-\pi^2\lambda^2((x-\alpha_j(t))^2+(y-t)^2)}dt.$$

When $dist((x,y),\Gamma) \ge \epsilon$, that is, $(x-\alpha_j(t))^2 + (y-t)^2 \ge \epsilon^2$ for all $t \in [0,1]$,

$$|\tilde{I}_\lambda(x,y)| \le c_0\lambda^2\mathrm{e}^{-\pi^2\lambda^2\epsilon^2}.$$

(ii) By (3.5) we have that

$$\tilde{I}_\lambda(x^*,y^*) = \int_{\mathbb{R}} \sum_{j=1}^{N_t} [f]_1(\alpha_j(t),t)\,\phi_\lambda(\alpha_{j_0}(y^*) - \alpha_j(t), y^* - t)dt. \tag{3.8}$$

It follows from the same argument in (i) that

$$\int_{|t-y^*|\ge\epsilon} \sum_{j=1}^{N_t} [f]_1(\alpha_j(t),t)\,\phi_\lambda(\alpha_{j_0}(y^*) - \alpha_j(t), y^* - t)dt \le c_0\lambda^2\mathrm{e}^{-\pi^2\lambda^2\epsilon^2},$$

and

$$\int_{\mathbb{R}} \sum_{j\ne j_0} [f]_1(\alpha_j(t),t)\,\phi_\lambda(\alpha_{j_0}(y^*) - \alpha_j(t), y^* - t)dt \le c_0\lambda^2\mathrm{e}^{-\pi^2\lambda^2\epsilon^2}.$$

These two inequalities combined with (3.8) yields that

$$\left|\tilde{I}_\lambda(x^*,y^*) - \tilde{I}_{\lambda,\epsilon}(x^*,y^*)\right| \le 2c_0\lambda^2\mathrm{e}^{-\pi^2\lambda^2\epsilon^2}. \tag{3.9}$$

where

$$\tilde{I}_{\lambda,\epsilon}(x^*,y^*) = \int_{|t-y^*|\le\epsilon} [f]_1(\alpha_{j_0}(t),t)\,\phi_\lambda(\alpha_{j_0}(y^*) - \alpha_{j_0}(t), y^* - t)dt. \tag{3.10}$$

We next estimate the integral $\tilde{I}_{\lambda,\epsilon}(x^*,y^*)$. Since both $[f]_1$ and α_{j_0} are smooth functions locally, there exist positive constants c_1 and c_2 such that for $|t-y^*| \le \epsilon$

$$\big|[f]_1(\alpha_{j_0}(t),t) - [f]_1(\alpha_{j_0}(y^*),y^*)\big| \le c_1|t-y^*|, \tag{3.11}$$

and

$$|\alpha_{j_0}(t)-\alpha_{j_0}(y^*)-\alpha'_{j_0}(y^*)(t-y^*)|\le c_2|t-y^*|^2. \tag{3.12}$$

By the definition of ϕ_λ in (3.1),

$$\phi_\lambda(\alpha_{j_0}(y^*)-\alpha_{j_0}(t),y^*-t)=\pi\lambda^2\mathrm{e}^{-\pi^2\lambda^2[(\alpha_{j_0}(y^*)-\alpha_{j_0}(t))^2+(t-y^*)^2]}.$$

Since α_{j_0} is smooth, there exists a positive constant c_3 such that $|\alpha_{j_0}(t)-\alpha_{j_0}(y^*)+\alpha'_{j_0}(y^*)(t-y^*)|\le c_3|t-y^*|$. This combined with (3.12) yields that $\big|(\alpha_{j_0}(t)-\alpha_{j_0}(y^*))^2-(\alpha'_{j_0}(y^*))^2(t-y^*)^2\big|\le c_2c_3|t-y^*|^3$. Substituting it into the above equality and putting

$$\psi_\lambda(t)=\pi\lambda^2\mathrm{e}^{-\pi^2\lambda^2[1+(\alpha'_{j_0}(y^*))^2](t-y^*)^2}, \tag{3.13}$$

we have

$$\Big|\phi_\lambda(\alpha_{j_0}(y^*)-\alpha_{j_0}(t),y^*-t)-\psi_\lambda(t)\Big|\le\psi_\lambda(t)\big(1-\mathrm{e}^{-\pi^2\lambda^2c_2c_3|t-y^*|^3}\big).$$

Since $1-\mathrm{e}^{-x}\le x$ for $x\ge0$,

$$\Big|\phi_\lambda(\alpha_{j_0}(y^*)-\alpha_{j_0}(t),y^*-t)-\psi_\lambda(t)\Big|\le\psi_\lambda(t)\pi^2\lambda^2c_2c_3|t-y^*|^3.$$

We can now estimate $\tilde I_{\lambda,\epsilon}(x^*,y^*)$ as in (3.10) following from the above inequality and (3.11)

$$\begin{aligned}
&\Big|\tilde I_{\lambda,\epsilon}(x^*,y^*)-\int_{|t-y^*|\le\epsilon}[f]_1(\alpha_{j_0}(y^*),y^*)\psi_\lambda(t)dt\Big|\\
\le\;&\Big|\int_{|t-y^*|\le\epsilon}\big([f]_1(\alpha_{j_0}(t),t)-[f]_1(\alpha_{j_0}(y^*),y^*)\big)\psi_\lambda(t)dt\Big|\\
&+\Big|\int_{|t-y^*|\le\epsilon}[f]_1(\alpha_{j_0}(y^*),y^*)(\phi_\lambda(\alpha_{j_0}(y^*)-\alpha_{j_0}(t),y^*-t)-\psi_\lambda(t))dt\Big|\\
\le\;&c_1\Big|\int_{|t-y^*|\le\epsilon}|t-y^*|\psi_\lambda(t)dt\Big|+c_2c_3|[f]_1(\alpha_{j_0}(y^*),y^*)|\Big|\int_{|t-y^*|\le\epsilon}\psi_\lambda(t)\pi^2\lambda^2|t-y^*|^3dt\Big|.
\end{aligned}$$

It is direct to observe from (3.13) that $\psi_\lambda(t)\le\pi\lambda^2\mathrm{e}^{-\pi^2\lambda^2(t-y^*)^2}$. Moreover, there exists a positive constant c_4 such that $c_2c_3|[f]_1(\alpha_{j_0}(y^*),y^*)\le c_4$ for all $y^*\in[0,1]$. Substituting these into the above inequality and having a change of variable $u=t-y^*$ yields that

$$\Big|\tilde I_{\lambda,\epsilon}(x^*,y^*)-\int_{|t-y^*|\le\epsilon}[f]_1(\alpha_{j_0}(y^*),y^*)\psi_\lambda(t)dt\Big|\le2c_1\pi\lambda^2\int_0^\epsilon u\mathrm{e}^{-\pi^2\lambda^2u^2}du+2c_4\pi^3\lambda^4\int_0^\epsilon u^3\mathrm{e}^{-\pi^2\lambda^2u^2}du.$$

Since $\mathrm{e}^{-\pi^2\lambda^2u^2}\le1$ for all $u\ge0$, it follows from a direct computation of the above integrals that

$$\Big|\tilde I_{\lambda,\epsilon}(x^*,y^*)-\int_{|t-y^*|\le\epsilon}[f]_1(\alpha_{j_0}(y^*),y^*)\psi_\lambda(t)dt\Big|\le c_1\pi(\lambda\epsilon)^2+\frac{1}{2}c_4\pi^3(\lambda\epsilon)^4. \tag{3.14}$$

We next estimate the integral in the above inequality. To this end, we let $F(a) = \int_{-a}^{a} \mathrm{e}^{-x^2} dx$. A direction computation from (3.13) gives that

$$\int_{|t-y^*|\le\epsilon} \psi_\lambda(t)dt = \frac{\lambda}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}} F\big(\pi\lambda\sqrt{1+(\alpha'_{j_0}(y^*))^2}\epsilon\big).$$

Note that by using the polar coordinates in the integral, we have the following estimates of $F(a)$: $\pi(1-\mathrm{e}^{-a^2}) \le F^2(a) \le \pi(1-\mathrm{e}^{-2a^2})$, which implies $|F(a) - \sqrt{\pi}| \le \sqrt{\pi}\mathrm{e}^{-a^2}$. Substituting it into the above equation, we have

$$\left|\int_{|t-y^*|\le\epsilon} \psi_\lambda(t)dt - \frac{\sqrt{\pi}\lambda}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}}\right| \le \frac{\sqrt{\pi}\lambda}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}} \mathrm{e}^{-\pi^2(1+(\alpha'_{j_0}(y^*))^2)\lambda^2\epsilon^2} \le \sqrt{\pi}\lambda\mathrm{e}^{-\pi^2\lambda^2\epsilon^2}.$$

Since $[f]_1$ is continuous, there exists a positive constant c_5 such that $|[f]_1(x^*, y^*)| \le c_5$ for all $y^* \in [0,1]$. It implies

$$\left|\int_{|t-y^*|\le\epsilon} [f]_1(\alpha_{j_0}(y^*), y^*)\psi_\lambda(t)dt - [f]_1(\alpha_{j_0}(y^*), y^*)\frac{\sqrt{\pi}\lambda}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}}\right| \le c_5\sqrt{\pi}\lambda\mathrm{e}^{-\pi^2\lambda^2\epsilon^2}.$$

The desired result follows from this combined with (3.9) and (3.14). □

We remark that we could choose appropriate λ and ϵ such that $\tilde{I}_\lambda(x,y)$ will be arbitrarily small when the point (x,y) is away from the edge curves and it will blow up when the point (x,y) is on the edge curve. That is, $\tilde{I}_\lambda(x,y)$ behaves like a "sharp mountain" around the edge curves. We will present the specific choices of λ and ϵ in the later results.

We are now ready to present the edge detection behavior of $I_{N,\lambda}$.

Theorem 3.4 *(i) When (x,y) is at least ϵ away from the edges, that is, $dist((x,y),\Gamma) \ge \epsilon$, there exists a positive constant c such that*

$$\left|\frac{I_{N,\lambda}(x,y)}{\sqrt{\pi}\lambda}\right| \le c\bigg(\lambda^2\mathrm{e}^{-\frac{3N^2}{2\lambda^2}} + \lambda\mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + \lambda\frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \frac{\|f_x\|_\infty}{\lambda}\bigg).$$

(ii) When (x^*, y^*) is on the edge, that is, $x^* = \alpha_{j_0}(y^*)$ for some $j_0 \in \mathbb{N}$, if there exists a $\epsilon > 0$ such that $d((\alpha_j(y), y), (\alpha_{j_0}(y^*), y^*)) \ge \epsilon$ for all $j \ne j_0$ and $y \in [0,1]$, then there exists a positive constant c such that

$$\left|\frac{I_{N,\lambda}(x^*,y^*)}{\sqrt{\pi}\lambda} - \frac{[f]_1(x^*,y^*)}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}}\right| \le c\bigg(\lambda^2\mathrm{e}^{-\frac{3N^2}{2\lambda^2}} + \lambda\frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \frac{\|f_x\|_\infty}{\lambda} + (\lambda+1)\mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + \lambda\epsilon^2 + \lambda^3\epsilon^4\bigg).$$

Proof: It follows immediately from Propositions 3.1, 3.2, and 3.3. □

Similarly, we could obtain the following estimates on $J_{N,\lambda}$.

Theorem 3.5 *(i) When (x,y) is at least ϵ away from the edges, that is, $dist((x,y),\Gamma) \ge \epsilon$,*

there exists a positive constant c such that

$$\left|\frac{J_{N,\lambda}(x,y)}{\sqrt{\pi}\lambda}\right| \le c\bigg(\lambda^2 \mathrm{e}^{-\frac{3N^2}{2\lambda^2}} + \lambda \mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + \lambda\frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \frac{\|f_y\|_\infty}{\lambda}\bigg).$$

(ii) When (x^*, y^*) is on the edge, that is, $y^* = \bar{\alpha}_{l_0}(x^*)$ for some $l_0 \in \mathbb{N}$, if there exists a $\epsilon > 0$ such that $d((x, \bar{\alpha}_l(x)), (x^*, \bar{\alpha}_{l_0}(x^*))) \ge \epsilon$ for all $l \ne l_0$ and $x \in [0,1]$, then there exists a positive constant c such that

$$\left|\frac{J_{N,\lambda(x^*,y^*)}}{\sqrt{\pi}\lambda} - \frac{[f]_2(x^*,y^*)}{\sqrt{1+(\bar{\alpha}'_{l_0}(x^*))^2}}\right| \le c\bigg(\lambda^2\mathrm{e}^{-\frac{3N^2}{2\lambda^2}} + \lambda\frac{\mathrm{e}^{-\pi^2\lambda^2}}{(1-\mathrm{e}^{-\pi^2\lambda^2})^2} + \frac{\|f_y\|_\infty}{\lambda} + (\lambda+1)\mathrm{e}^{-\pi^2\lambda^2\epsilon^2} + \lambda\epsilon^2 + \lambda^3\epsilon^4\bigg).$$

Proof: It follows immediately from Propositions 3.1, 3.2, and 3.3. □

Consequently, we will present the main result of this paper below. In particular, we will show the specific choices of λ and ϵ such that the edge detector $E_{N,\lambda}$ as in (2.2) behaves like oscillation-free sharp "mountains" around the edges.

Theorem 3.6 *If $\lambda = c_0\frac{N}{\log N}$ and $\epsilon = c_1(\frac{N}{\log N})^{-p}$ for some positive constants c_0, c_1 and $\frac{3}{4} < p < 1$, then for large enough N,*

(i) when (x,y) is at least ϵ away from the edges, that is, $dist((x,y),\Gamma) \ge \epsilon$, there exists a positive constant c such that

$$|E_{N,\lambda}(x,y)| \le c\,\frac{\log N}{N};$$

(ii) when (x^*, y^*) is on the edge, that is, $x^* = \alpha_{j_0}(y^*)$ and $y^* = \bar{\alpha}_{l_0}(x^*)$ for some $j_0, l_0 \in \mathbb{N}$, if there exists a $\epsilon > 0$ such that $d((\alpha_j(y), y), (\alpha_{j_0}(y^*), y^*)) \ge \epsilon$ for all $j \ne j_0$ and $y \in [0,1]$ and $d((x, \bar{\alpha}_l(x)), (x^*, \bar{\alpha}_{l_0}(x^*))) \ge \epsilon$ for all $l \ne l_0$ and $x \in [0,1]$, then there exists a positive constant c such that

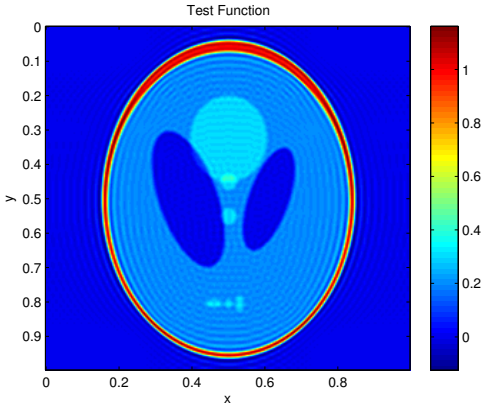
$$\left|E_{N,\lambda}(x^*,y^*) - [f](x^*,y^*)\right| \le c\left(\frac{\log N}{N}\right)^{4p-3}.$$

Proof: (i) It follows from a direct computation from substituting the choices of λ and ϵ into Theorems 3.4, 3.5 and the definition of $E_{N,\lambda}$ in (2.2).

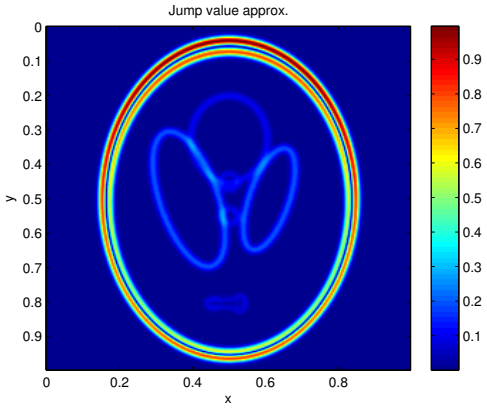
(ii) Note that when $x^* = \alpha_{j_0}(y^*)$ and $y^* = \bar{\alpha}_{l_0}(x^*)$, we have $[f](x^*,y^*) = [f]_1(x^*,y^*) = [f]_2(x^*,y^*)$ and $\left(\frac{1}{\sqrt{1+(\alpha'_{j_0}(y^*))^2}}\right)^2 + \left(\frac{1}{\sqrt{1+(\bar{\alpha}'_{l_0}(x^*))^2}}\right)^2 = 1$. The desired result follows immediately from a direct computation from substituting the choices of λ and ϵ into Theorems 3.4, 3.5 and the definition of $E_{N,\lambda}$ in (2.2). □

4 Numerical Results

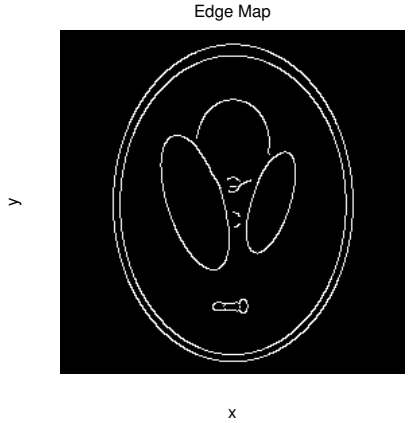
We now present numerical results demonstrating the accuracy of the proposed formulation. Matlab code used to generate the figures in this section can be found at [14]. We begin with Figure 4, where we plot the edge map of a Shepp-Logan phantom on a 256×256 grid given its



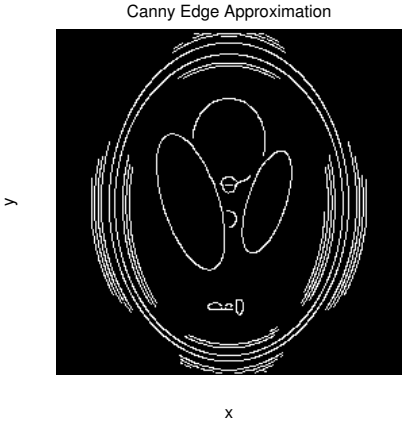
(a) Fourier Reconstruction



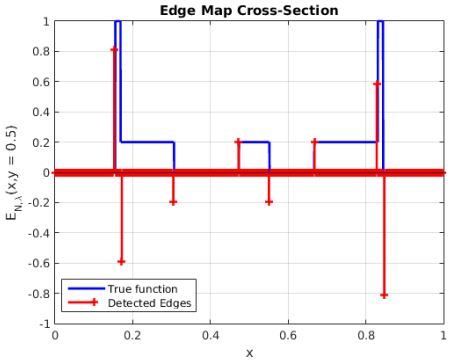
(b) Jump Function Approximation



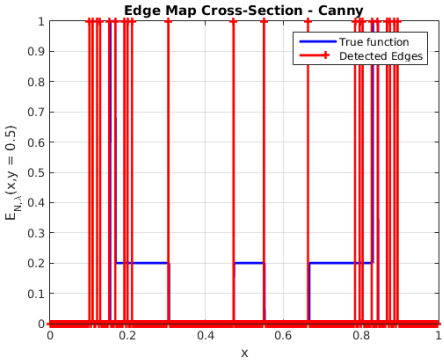
(c) Edge Map



(d) Canny Edge Detector



(e) Cross-Section ($y = 0.5$)



(f) Cross-Section (Canny, $y = 0.5$)

Figure 4: Edge Detection — Shepp-Logan Phantom; $S_N = [-50, 50]^2 \cap \mathbb{Z}^2$ while the equispaced reconstruction grid is of size 256×256 in $[0, 1]^2$.

first 50×50 Fourier modes. Low-resolution measurement acquisitions such as this are common in MR imaging applications For reference, the partial Fourier sum reconstruction (showing significant Gibbs oscillations) is plotted in Figure 4a. Applying the proposed spectral mollifier, the resulting jump funtion approximation is shown in Figure 4b, while the resulting edge map is shown in Figure 4c. Hysteresis edge tracking (similar to that implemented in the Canny edge detector) was used to obtain Figure 4c from Figure 4b. For comparison, we also plot in Figure 4d the results of applying the standard Canny edge detector. Note the presence of a significant number of false positives (see also Figures 4e and 4f for cross-section plots) — these are due to the Gibbs oscillations being spuriously identified as edges by the Canny algorithm. Finally, we note that the proposed method also provides approximations to the jump height (as illustrated in Figure 4e) which may be useful in certain applications such as the solution of PDEs.

Next, we present a higher resolution example in Figure 5, where the edges in the Shepp-Logan are identified starting with the first 200×200 Fourier modes. As before, the results are plotted on a 256×256 equispaced grid. Figure 5a plots the Fourier partial sum reconstruction for reference while Figure 5b plots the jump function approximation. Figures 5c and 5d plot the edge maps generated by the proposed method and the Canny edge detector respectively, while Figures 5e and 5f show the corresponding cross-section plots. In this case, Gibbs oscillations in the Fourier reconstruction are localized to regions close to the true edge locations. Moreover, the standard Canny edge detector does a good job of recognizing and suppressing spurious Gibbs oscillations from true edges. However, note that some of the closely spaced edges are either missing or spuriously identified by the Canny edge detector (see the cross- section plots for an illustration), while the proposed method accurately identifies these.

Figures 4 and 5 have illustrated the performance of the method when we have perfect (noiseless) measurements. We now consider the case where the Fourier modes are corrupted by additive (complex) Gaussian noise; i.e.,

$$\hat{g}(\mathbf{z}) = \hat{f}(\mathbf{z}) + \hat{n}(\mathbf{z}), \quad \mathbf{z} = (z_1, z_2) \in S_N := [-N, N]^2 \cap \mathbb{Z}^2,$$

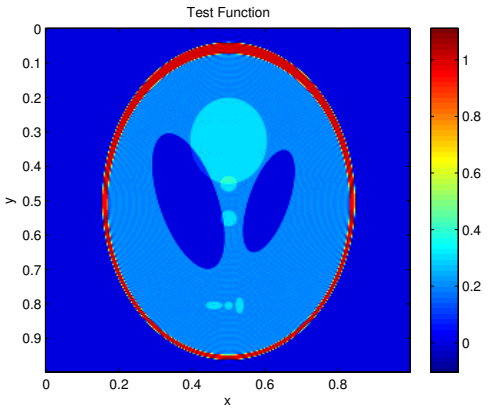
where $\hat{f}$ an $\hat{g}$ denote the true and noise corrupted Fourier coefficients respectively, and $\hat{n}$ denotes additive noise in Fourier space. In Figure 6, the first 50×50 Fourier modes of the Shepp-Logan phantom are corrupted by i.i.d. additive complex Gaussian noise of variance $\frac{1}{2N^2} = 2 \times 10^{-4}$. The equivalent PSNR is

$$\text{PSNR (dB)} = 20 \log_{10} \frac{\text{max. image intensity}}{\sqrt{\text{Mean Square Error}}} = \frac{\max_{i,j} |f(x_i, y_j)|}{\sqrt{\dfrac{1}{M_x M_y} \displaystyle\sum_{i=0}^{M_x - 1} \sum_{j=0}^{M_y - 1} \left[S_N f(x_i, y_j) - S_N g(x_i, y_j) \right]^2}},$$

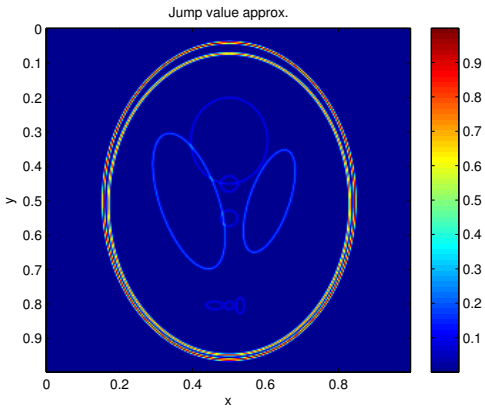
where M_x, M_y are the number of points in the reconstruction grid ($M_x = M_y = 256$ in Figure 6) and $S_N f, S_N g$ are the Fourier partial sum reconstructions of f and g respectively:

$$S_N f(x, y) = \sum_{\mathbf{z} \in \mathbf{S_N}} \hat{f}(\mathbf{z}) e^{2\pi i (z_1 x + z_2 y)}, \qquad S_N g(x, y) = \sum_{\mathbf{z} \in \mathbf{S_N}} \hat{g}(\mathbf{z}) e^{2\pi i (z_1 x + z_2 y)}.$$

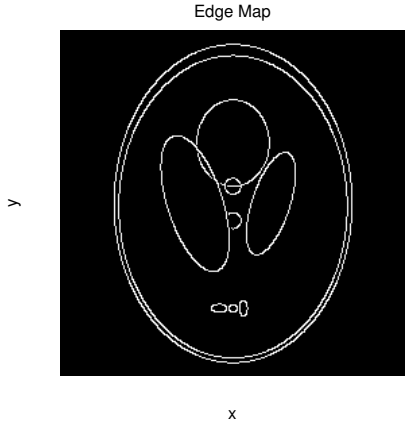
As before the jump function approximation, edge maps using the proposed method and the



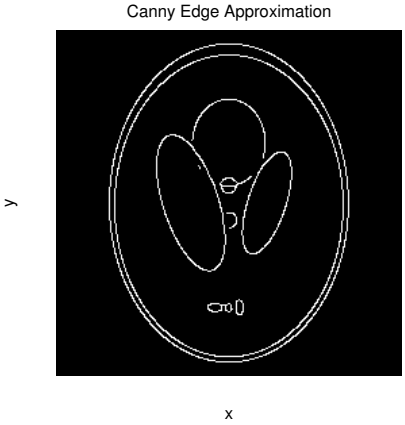
(a) Fourier Reconstruction



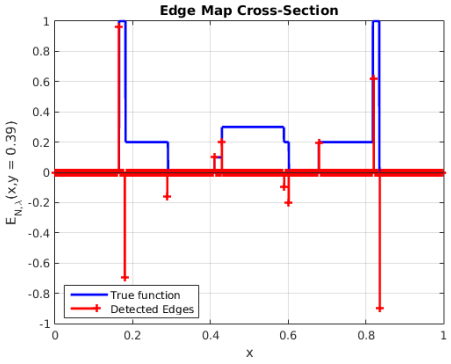
(b) Jump Function Approximation



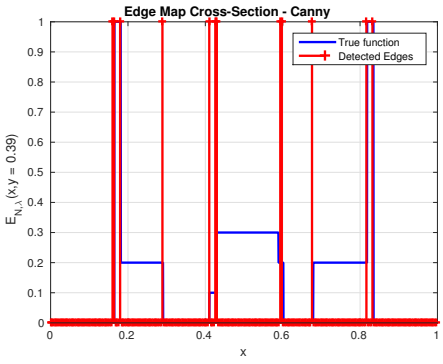
(c) Edge Map



(d) Canny Edge Detector

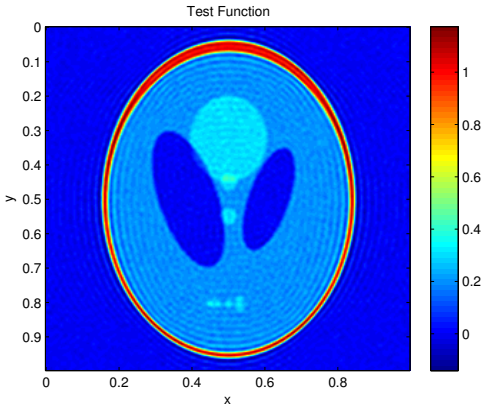


(e) Cross-Section ($y = 0.39$)

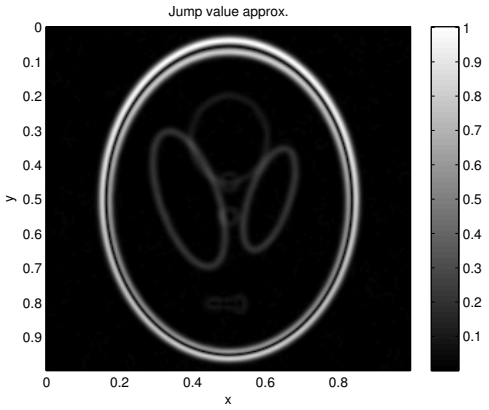


(f) Cross-Section (Canny, $y = 0.39$)

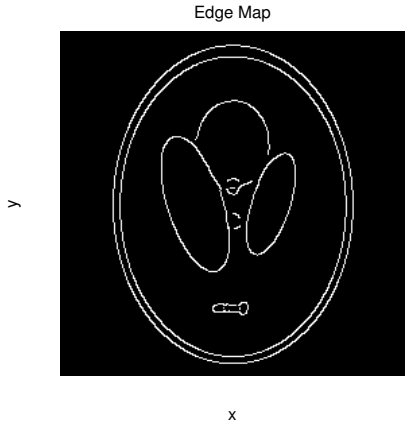
Figure 5: Edge Detection — Shepp-Logan Phantom; $S_N = [-200, 200]^2 \cap \mathbb{Z}^2$ while the equispaced reconstruction grid is of size 256×256 in $[0, 1]^2$.



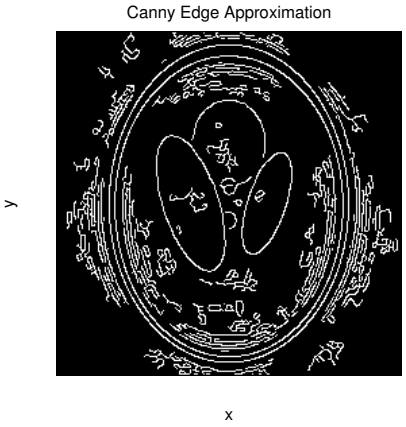
(a) Fourier Reconstruction



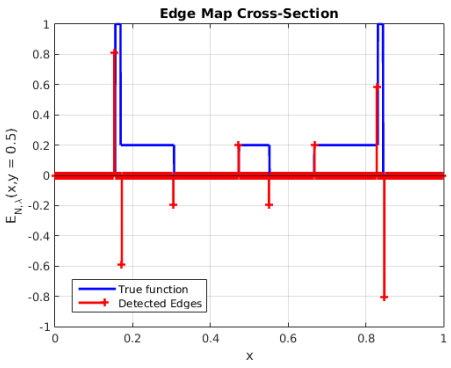
(b) Jump Function Approximation



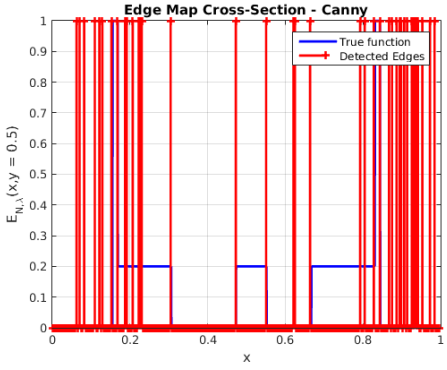
(c) Edge Map



(d) Canny Edge Detector



(e) Cross-Section ($y = 0.5$)



(f) Cross-Section (Canny, $y = 0.5$)

Figure 6: Noisy Edge Detection — Shepp-Logan Phantom; $S_N = [-50, 50]^2 \cap \mathbb{Z}^2$ while the equispaced reconstruction grid is of size 256×256 in $[0, 1]^2$. Additive complex white Gaussian noise of variance 2×10^{-4} (PSNR 36.93 dB) was added to the Fourier modes.

Canny edge detector, and the cross sections of the edge maps are shown in Figures 6a – 6f respectively. We observe that the addition of noise to pre-existing Gibbs oscillations results in the Canny edge detector generating numerous spurious edges, while the proposed method suppresses almost all of these artifacts and generates a near-perfect edge map.

5 Concluding Remarks

In this paper, we have introduced a class of spectral mollifiers for the detection of edges from two-dimensional truncated Fourier data. Recall that the problem of detecting edges from Fourier spectral data is different from and more challenging than the problem of detecting edges from pixel data. Indeed, distinguishing between true edges and Gibbs oscillations is a non-trivial task, especially when we start with a small number of (possibly noise corrupted) Fourier coefficients. We have shown through rigorous analysis that the jump approximations generated using the proposed spectral mollifier are guaranteed to be free of spurious oscillations and edges. Numerical results show that the resulting edge maps are accurate and outperform standard methods such as the Canny edge detector, especially in cases where we have truncated and/or noisy data.

Several interesting avenues for future research exist, including the extension of these results to the case of non-harmonic Fourier data, investigation of the performance of this method for highly incomplete or interrupted data, and the extension of the method to the case of distributed data acquisition.

References

[1] Rafael C. Gonzales and Richard E. Woods. *Digital Image Processing*. Prentice Hall, 3rd edition, 2007.

[2] Jan S. Hesthaven, Sigal Gottlieb, and David Gottlieb. *Spectral Methods for Time-Dependent Problems*. Cambridge University Press, 2007.

[3] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 71(2):231–303, 1987.

[4] Xu-Dong Liu, Stanley Osher, and Tony Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[5] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679–698, 1986.

[6] Doug Cochran, Anne Gelb, and Yang Wang. Edge detection from truncated Fourier data using spectral mollifiers. *Advances in Computational Mathematics*, 38(4):737–762, 2013.

[7] Anne Gelb and Eitan Tadmor. Detection of edges in spectral data. *Applied and Computational Harmonic Analysis*, 7(1):101–135, 1999.

[8] Anne Gelb and Eitan Tadmor. Detection of edges in spectral data II. Nonlinear enhancement. *SIAM Journal on Numerical Analysis*, 38(4):1389–1408, 2000.

[9] Anne Gelb and Eitan Tadmor. Adaptive edge detectors for piecewise smooth data based on the minmod limiter. *Journal of Scientific Computing*, 28(2–3):279–306, 2006.

[10] Eitan Tadmor and Jing Zou. Three novel edge detection methods for incomplete and noisy spectral data. *Journal of Fourier Analysis and Applications*, 14(5–6):744–763, 2008.

[11] Anne Gelb and Dennis Cates. Detection of edges in spectral data III – Refinement of the concentration method. *Journal of Scientific Computing*, 36(1):1–43, 2008.

[12] Alex Petersen, Anne Gelb, and Randall Eubank. Hypothesis testing for Fourier based edge detection methods. *Journal of Scientific Computing*, 51(3):608–630, 2012.

[13] Adam Martinez, Anne Gelb, and Alexander Gutierrez. Edge detection from non-uniform Fourier data using the convolutional gridding algorithm. *Journal of Scientific Computing*, 61(3):490–512, 2014.

[14] Anne Gelb, Guohui Song, Aditya Viswanathan, and Yang Wang. GiFtED: Matlab software for Gibbs-Free Edge Detection, version 1.0. `https://bitbucket.org/charms/gifted`, 2015.

RANDOM MATRICES AND ERASURE ROBUST FRAMES

YANG WANG

Abstract. Data erasure can often occur in communication. Guarding against erasures involves redundancy in data representation. Mathematically this may be achieved by redundancy through the use of frames. One way to measure the robustness of a frame against erasures is to examine the worst case condition number of the frame with a certain number of vectors erased from the frame. The term *numerically erasure-robust frames (NERFs)* was introduced in [9] to give a more precise characterization of erasure robustness of frames. In the paper the authors established that random frames whose entries are drawn independently from the standard normal distribution can be robust against up to approximately 15% erasures, and asked whether there exist frames that are robust against erasures of more than 50%. In this paper we show that with very high probability random frames are, independent of the dimension, robust against any amount of erasures as long as the number of remaining vectors is at least $1 + \delta$ times the dimension for some $\delta_0 > 0$. This is the best possible result, and it also implies that the proportion of erasures can arbitrarily close to 1 while still maintaining robustness. Our result depends crucially on a new estimate for the smallest singular value of a rectangular random matrix with independent standard normal entries.

1. Introduction

Let $\mathbf{H}$ be a Hilbert space. A set of elements $\mathcal{F} = \{\mathbf{f}_n\}$ in $\mathbf{H}$ (counting multiplicity) is called a *frame* if there exist two positive constants C_* and C^* such that for any $\mathbf{v} \in \mathbf{H}$ we have

$$C_*\|\mathbf{v}\|^2 \leq \sum_n |\langle \mathbf{v}, \mathbf{f}_n\rangle|^2 \leq C^*\|\mathbf{v}\|^2. \tag{1.1}$$

The constants C_* and C^* are called the *lower frame bound* and the *upper frame bound*, respectively. A frame is called a *tight frame* if $C_* = C^*$. In this paper we focus mostly on real finite dimensional Hilbert spaces with $\mathbf{H} = \mathbb{R}^n$ and $\mathcal{F} = \{\mathbf{f}_n\}_{j=1}^N$, although we shall also discuss the extendability of the results to the complex case. Let $F = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_N]$. It

1991 *Mathematics Subject Classification.* Primary 42C15.

Key words and phrases. Random matrices, singular values, numerically erasure robust frame (NERF), condition number, restricted isometry property.

Yang Wang was supported in part by the National Science Foundation grant DMS-08135022 and DMS-1043032.

is called the *frame matrix* for $\mathcal{F}$. It is well known that $\mathcal{F}$ is a frame if and only if the $n \times N$ matrix F has rank n. Furthermore, the optimal frames bounds are given by

$$C_* = \sigma_n^2(F), \qquad C^* = \sigma_1^2(F),$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n > 0$ are the singular values of F. Throughout this paper we shall identify without loss of generality a frame $\mathcal{F}$ by its frame matrix.

The main focus of the paper is on the *erasure robustness* property for a frame. This property arise in applications such as communication where data can be lost or corrupted in the process of transmission. Suppose that we have a frame F that is *full spark* in the sense that every n columns of F span $\mathbb{R}^n$, it is theoretically possible to erase up to $N - n$ data from the full set of data $\{\langle \mathbf{v}, \mathbf{f}_j \rangle\}_{j=1}^N$ while still reconstruct the signal $\mathbf{v}$. This is a simple consequence of the property that with the remaining available data $\{\langle \mathbf{v}, \mathbf{f}_j \rangle\}_{j \in S}$ with $|S| \geq n$, $\mathbf{v}$ is uniquely determined because $\text{span}(\mathbf{f}_j : j \in S) = \mathbb{R}^n$. In practice, however, the condition number of the matrix $[\mathbf{f}_j]_{j \in S}$ could be so poor that the reconstruction is numerically unstable against the presence of additive noise in the data. Thus robustness against data loss and erasures is a highly desirable property for a frame. There have been a number of studies that aim to address this important issue.

Among the first studies of erasure-robust frames was given in [10]. It was shown in subsequent studies that that unit norm tight frames are optimally robust against one erasure [**?**] while Grassmannian frames are optimally robust against two erasures [16, 11]. The literature on erasure robustness for frames is quite extensive, see e.g. also [12, 18, 13]. In general, the robustness of a frame F against q-erasures, where $q \leq N - n$, is measured by the maximum of the condition numbers of all $n \times (N - q)$ submatrices of F. More precisely, let $S \subseteq \{1, 2, \ldots, N\}$ and let F_S denote the $n \times |S|$ submatrix of F with columns $\mathbf{f}_j$ for $j \in S$ (in its natural order, although the order of the columns is irrelevant). Then the robustness against q-erasures of F is measured by

$$R(F, q) := \max_{|S| = N - q} \frac{\sigma_1(F_S)}{\sigma_n(F_S)}. \tag{1.2}$$

Of course, the smaller $R(F, q)$ is the more robust F is against q-erasures. In [9], Fickus and Mixon coined the term *numerically erasure robust frame (NERF)*. A frame F is (K, α, β)-NERF if

$$\alpha \leq \sigma_n(F_S) \leq \sigma_1(F_S) \leq \beta \qquad \text{for any } S \subseteq \{1, 2, \ldots, N\},\ |S| = K.$$

Thus in this case $R(F, N-K) \leq \beta/\alpha$. Note that for any full spark $n \times N$ frame matrix F and any $n \leq K \leq N$ there always exist $\alpha, \beta > 0$ such that F is (K, α, β)-NERF. The main goal is to find classes of frames where the bounds α, β, and more importantly, $R(F, N-K) = \beta/\alpha$, are independent of the dimension n while allowing the proportion of erasures $1 - \frac{K}{N}$ as large as possible. The authors studied in [9] the erasure robustness of $F = \frac{1}{\sqrt{n}}A$, where the entries of A are independent random variables of the standard normal $\mathcal{N}(0,1)$ distribution. It was shown that with high probability such a matrix can be good NERFs provided that K is no less than approximately 85% of N. The authors also proved that equiangular frame F in $\mathbb{C}^n$ with $N = n^2 - n + 1$ vectors is a good NERF against up to about 50% erasures. As far as the proportion of erasures is concerned this was the best known result for NERFs. However, the frame requires almost n^2 vectors. The authors posed as an open question whether there exist NERFs with $K < N/2$. A more recent paper [8] explored a deterministic construction based on certain group theoretic techniques. The approach offers more flexibility in the frame design than the far more restrictive equiangular frames.

In this paper we revisit the robustness of random frames. We provide a much stronger result for random frames, showing that for any $\delta > 0$, with very high probability, the frame $F = \frac{1}{\sqrt{n}}A$ is a $((1+\delta)n, \alpha, \beta)$-NERF where α, β depend only on δ and the aspect ratio $\frac{N}{n}$. One version of our result is given by the following theorem.

Theorem 1.1. *Let $F = \frac{1}{\sqrt{n}}A$ where A is $n \times N$ whose entries are independent Gaussian random variables of $N(0,1)$ distribution. Let $\lambda = \frac{N}{n} > 1$. Then for any $0 < \delta_0 < \lambda - 1$ and $\tau_0 > 0$ there exist $\alpha, \beta > 0$ depending only on δ_0, λ and τ_0 such that for any $\delta_0 \leq \delta < \lambda - 1$, the frame F is a $((1+\delta)n, \alpha, \beta)$-NERF with probability at least $1 - e^{-\tau_0 n}$.*

Later in the paper we shall provide more implicit estimates for α, β that will allow us to easily compute them numerically. Note that our result is essentially the best possible, as we cannot go to $\delta_0 = 0$. A corollary of the theorem is that for random Gaussian frames the proportion of erasures $1 - \frac{K}{N}$ can be made arbitrary large while the frames still maintain robustness with overwhelming probability.

Our theorem depends crucially on a refined estimate on the smallest singular value of a random Gaussian matrix. There is a wealth of literature on random matrices. The study of singular values of random matrices has been particularly intense in recent years due

to their applications in compressive sensing for the construction of matrices with the so-called *restricted isometry property* (see e.g.[4, 5, 1, 2]). Random matrices have also been employed for phase retrieval [3], which aims to reconstruct a signal from the magnitudes of its samples. For a very informative and comprehensive survey of the subject we refer the readers to [15, 19], which also contains an extensive list of references (among the notable ones [7, 14, 17]). For the $n \times N$ Gaussian random matrix A the expected value of $\sigma_1(A)$ and $\sigma_n(A)$ are asymptotically $\sqrt{N} + \sqrt{n}$ and $\sqrt{N} - \sqrt{n}$, respectively. Many important results, such as the NERF analysis of random matrices in [9] as well as results on the restricted isometry property in compressive sensing, often utilize known estimates of $\sigma_1(A)$ and $\sigma_n(A)$ based on Hoeffding-type inequalities. One good such estimate is

$$\mathbb{P}\left(\sigma_n(A) < \sqrt{N} - \sqrt{n} - t\right) \leq e^{-\frac{t^2}{2}}, \tag{1.3}$$

see [19]. The problem with this estimate is that even by taking $t = \sqrt{N} - \sqrt{n}$ we only get a bound of $e^{-(\sqrt{\lambda}-1)^2 n/2}$ even though the probability in this case is 0. Thus estimates such as (1.3) that cap the decay rate are often inadequate. When applied to the erasure robustness problem for frames they usually put a cap on the proportion of erasures. To go further we must prove an estimate that will allow the exponent of decay to be much larger. We achieve this goal by proving the following theorem:

Theorem 1.2. *Let A be $n \times N$ whose entries are independent random variables of standard normal $N(0,1)$ distribution. Let $\lambda = \frac{N}{n} > 1$. Then for any $\mu > 0$ there exist constants $c, C > 0$ depending only on μ and λ such that*

$$\mathbb{P}\left(c\sqrt{n} \leq \sigma_n(A) \leq \sigma_1(A) \leq C\sqrt{n}\right) \geq 1 - 3e^{-\mu n}. \tag{1.4}$$

Furthermore, we may take $C = 1 + \sqrt{\lambda} + \sqrt{\mu}$ and $c = \sup_{0<t<1} \varphi(t)$ where

$$\varphi(t) = \frac{t^{\frac{1}{\lambda}}}{L} - \frac{2Ct}{1-t}, \qquad \text{where} \qquad L = \sqrt{\frac{2e}{\lambda}} e^{\frac{\mu}{\lambda}}. \tag{1.5}$$

Acknowledgement. The author would like to thank Radu Balan and Dustin Mixon for very helpful discussions.

2. Smallest Singular Value of a Random Matrix: Nonasymptotic Estimate

We begin with estimates on the extremal singular values of a ranodm matrix A whose entries are independent standard normal random variables. We shall assume throughout

the section that A is $n \times N$ where $\frac{N}{n} = \lambda > 1$. One of the very important estimates is

$$\mathbb{P}\left(\sigma_1(A) > \sqrt{N} + \sqrt{n} + t\right) \leq e^{-\frac{t^2}{2}}, \tag{2.1}$$

see [19]. Our main goal of this section is to prove the estimates for smallest singular value $\sigma_n(A)$ stated in Theorem 1.2. An equivalent formulation of (2.1) is

$$\mathbb{P}\left(\sigma_1(A) > C\sqrt{n}\right) \leq e^{-\frac{(C-1-\sqrt{\lambda})^2}{2}n}, \qquad C \geq 1 + \sqrt{\lambda}. \tag{2.2}$$

Observe that

$$\sigma_n(A) = \min_{\mathbf{v} \in S^{n-1}} \|A^*\mathbf{v}\|,$$

where S^{n-1} denotes the unit sphere in $\mathbb{R}^n$.

Lemma 2.1. *Let $c > 0$. For any $\mathbf{v} \in S^{n-1}$ the probability $\mathbb{P}(\|A^*\mathbf{v}\| \leq c)$ is independent of the choice of $\mathbf{v}$. We have*

$$\mathbb{P}\left(\|A^*\mathbf{v}\| \leq \sqrt{\delta n}\right) \leq \left(\frac{2e\delta}{\lambda}\right)^{\frac{N}{2}} \tag{2.3}$$

for any $\delta > 0$.

Proof. The fact that $\mathbb{P}(\|A^*\mathbf{v}\| \leq c)$ is independent of the choice of $\mathbf{v}$ is a well know fact, which stems from the fact that the entries of PA are again independent standard normal random variables for any orthogonal $n \times n$ matrix P. In particular, one can always find an orthogonal P such that $P\mathbf{v} = e_1$. Thus we may without loss of generality take $\mathbf{v} = e_1$. In this case $\|A^*\mathbf{v}\|^2 = a_{11}^2 + \cdots + a_{1N}^2$ where $[a_{11}, \ldots, a_{1N}]$ denotes the first row of A. Denote $Y_N = a_{11}^2 + \cdots + a_{1N}^2$. Then Y_N has the $\Gamma(\frac{N}{2}, 1)$ distribution, which has the density function

$$\rho(t) = \frac{1}{\Gamma(\frac{N}{2})} e^{-t} t^{\frac{N}{2}-1}, \quad t > 0.$$

Denote $m = \frac{N}{2}$. It follows that

$$\begin{aligned}
\mathbb{P}\left(\|A^*\mathbf{v}\| \leq \sqrt{\delta n}\right) &= \mathbb{P}\left(Y_N \leq \delta n\right) \\
&= \frac{1}{\Gamma(m)} \int_0^{\delta n} e^{-t} t^{m-1} dt \\
&\leq \frac{1}{\Gamma(m)} \int_0^{\delta n} t^{m-1} dt \\
&= \frac{\delta^m n^m}{\Gamma(m)}.
\end{aligned}$$

Note that $\Gamma(m) \geq (\frac{m}{e})^m$ by Stirling's formula. The theorem now follows from $\frac{N}{n} = \lambda$ and $m = \frac{N}{2}$. ∎

A ubiquitous tool in the study of random matrices is an ε-net. F or any $\varepsilon > 0$ an ε-net for S^{n-1} is a set in S^{n-1} such that any point on S^{n-1} is no more than ε distance away from the set. The following result is known and can be found in [19]:

Lemma 2.2. *For any $\varepsilon > 0$ there exists an ε-net $\mathcal{N}_\varepsilon$ in S^{n-1} with cardinality no larger than $(1 + 2\varepsilon^{-1})^n$.*

Proof of Theorem 1.2. Assume that $\sigma_n(A) = b\sqrt{n}$. Then there exists a $\mathbf{v}_0 \in S^{n-1}$ such that $\|A^*\mathbf{v}_0\| = b\sqrt{n}$. Let $\mathcal{N}_\varepsilon$ be an ε-net for S^{n-1} and take $\mathbf{u} \in \mathcal{N}_\varepsilon$ that is the closest to $\mathbf{v}_0$. So $\|\mathbf{u} - \mathbf{v}_0\| \leq \varepsilon$. Thus

$$\|A^*\mathbf{u}\| \leq \|A^*\mathbf{v}_0\| + \|A^*(\mathbf{u} - \mathbf{v}_0)\| \leq b\sqrt{n} + \varepsilon\sigma_1(A). \tag{2.4}$$

Hence

$$\mathbb{P}\left(\sigma_n(A) \leq c\sqrt{n}\right) \leq \sum_{\mathbf{u}\in\mathcal{N}_\varepsilon} \mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A)\right). \tag{2.5}$$

Note that

$$\begin{aligned}\mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A)\right) &= \mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A), \sigma_1(A) \leq C\sqrt{n}\right)\\ &+ \mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A), \sigma_1(A) > C\sqrt{n}\right).\end{aligned}$$

By Lemma 2.1 the first term on the right hand side is bounded from above by

$$\begin{aligned}&\mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A), \sigma_1(A) \leq C\sqrt{n}\right)\\ &\qquad \leq \mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon C\sqrt{n}\right) \leq \left(\frac{2e(c + \varepsilon C)^2}{\lambda}\right)^{\frac{N}{2}}.\end{aligned}$$

By (2.2) the second term on the right hand side is bounded from above by

$$\begin{aligned}&\mathbb{P}\left(\|A^*\mathbf{u}\| \leq c\sqrt{n} + \varepsilon\sigma_1(A), \sigma_1(A) > C\sqrt{n}\right)\\ &\qquad \leq \mathbb{P}\left(\sigma_1(A) > C\sqrt{n}\right) \leq e^{-\frac{(C-1-\sqrt{\lambda})^2}{2}n}.\end{aligned}$$

Thus combining these two upper bounds we obtain the estimate

$$\mathbb{P}\left(\sigma_n(A) \leq c\sqrt{n}\right) \leq \left(1 + \frac{2}{\varepsilon}\right)^n \left(\left(\frac{2e(c + \varepsilon C)^2}{\lambda}\right)^{\frac{N}{2}} + e^{-\frac{(C-1-\sqrt{\lambda})^2}{2}n}\right). \tag{2.6}$$

We would like to bound $\mathbb{P}\left(\sigma_n(A) \leq c\sqrt{n}\right)$ by $2e^{-\mu n}$. All we need then is to choose $\varepsilon, c, C > 0$ so that both upper bound terms in (2.6) are bounded by $e^{-\mu n}$. Note that

$\frac{N}{2} = \frac{\lambda}{2}n$. Hence we only need

$$-\mu \geq \ln(1+2\varepsilon^{-1}) + \frac{\lambda}{2}\Big(\ln 2e - \ln\lambda + 2\ln(c+\varepsilon C)\Big), \tag{2.7}$$

$$-\mu \geq -\frac{1}{2}(C-1-\sqrt{\lambda})^2. \tag{2.8}$$

The equation (2.8) leads to the condition

$$C \geq \sqrt{2\mu} + \sqrt{\lambda} + 1. \tag{2.9}$$

To meet condition (2.7) we set $c = r\varepsilon$. Then $\ln(c+\varepsilon C) = -\ln\varepsilon^{-1} + \ln(r+C)$. Thus (2.7) becomes

$$(\lambda-1)\ln(\varepsilon^{-1}) \geq \mu + \ln(2+\varepsilon) + \frac{\lambda}{2}\ln\Big(\frac{2e(r+C)^2}{\lambda}\Big). \tag{2.10}$$

Clearly, once we fix C and r, say, take $C = \sqrt{2\mu} + \sqrt{\lambda} + 1$ and $r = 1$, $\ln\varepsilon^{-1}$ will be greater than the right hand side of (2.10) for small enough ε because of the condition $\lambda > 1$. Both C, c only depend on λ and μ. The existence part of the theorem is thus proved.

While we have already a good explicit estimate $C = \sqrt{2\mu}+\sqrt{\lambda}+1$, it remains to establish the explicit formula for c. For any fixed r the largest ε is achieved when (2.10) is an equality, namely

$$(\lambda-1)\ln(\varepsilon^{-1}) = \mu + \ln(2+\varepsilon) + \frac{\lambda}{2}\ln\Big(\frac{2e(r+C)^2}{\lambda}\Big),$$

which one can rewrite as

$$\ln(r+C) = -(1-p)\ln\varepsilon - p\ln(2+\varepsilon) - \ln L,$$

where $p = \lambda^{-1}$ and $L = \sqrt{\frac{2e}{\lambda}}e^{\frac{\mu}{\lambda}}$. It follows that

$$r\varepsilon = \frac{1}{L}\Big(\frac{\varepsilon}{2+\varepsilon}\Big)^p - C\varepsilon = \frac{1}{L}t^{\frac{1}{\lambda}} - \frac{2Ct}{1-t},$$

where $t = \frac{\varepsilon}{2+\varepsilon}$. Note that $0 < t < 1$. Now we can take c to be the supreme value of $r\varepsilon$, which yields

$$c = \sup_{0<t<1}\Big\{\frac{t^{\frac{1}{\lambda}}}{L} - \frac{2Ct}{1-t}\Big\}. \tag{2.11}$$

Finally, (1.4) follows from $\mathbb{P}\left(\sigma_n(A) \leq c\sqrt{n}\right) \leq 2e^{-\mu n}$ and (2.2). The proof of the theorem is now complete. ∎

Remark. Although there does not seem to exist an explicit formula for c given in (2.11), there is a very good explicit approximation of it. In general, the t that maximize $\varphi(t)$ is

rather small. So we may approximate $\frac{2Ct}{1-t}$ simply by $2Ct$ and find the maximum of

$$\tilde{\varphi}(t) = \frac{1}{L}t^{\frac{1}{\lambda}} - 2Ct. \tag{2.12}$$

The maximum of $\tilde{\varphi}(t)$ is obtained at $t_0 = (2C\lambda L)^{-\frac{\lambda}{\lambda-1}}$. This t_0 is very close to the actual t that maximizes $\varphi(t)$. Thus

$$\tilde{c} := \varphi(t_0) = \Big(\frac{1}{2C\lambda L^{\lambda}}\Big)^{\frac{1}{\lambda-1}}\Big(1 - \frac{1}{\lambda}\Big) \tag{2.13}$$

has $\tilde{c} \leq c$ and it is a close approximation of the optimal c. Of course, Theorem 1.2 still holds when c is replaced by $\tilde{c}$.

Although Theorem 1.2 is for real Gaussian random matrices, a complex version of it can also be proved with minor modifications. A complex random variable $Z = X + iY$ has the *complex standard normal* distribution if both X and Y have the real complex normal distribution $\mathcal{N}(0,1)$. Theorem 1.2 extends to the following theorem for the complex case:

Theorem 2.3. *Let A be $n \times N$ whose entries are independent random variables of complex standard normal $N(0,1)$ distribution. Let $\lambda = \frac{N}{n} > 1$. Then for any $\mu > 0$ there exist constants $c, C > 0$ depending only on μ and λ such that*

$$\mathbb{P}\left(c\sqrt{n} \leq \sigma_n(A) \leq \sigma_1(A) \leq C\sqrt{n}\right) \geq 1 - 3e^{-\mu n}. \tag{2.14}$$

Furthermore, we may take $C = \sqrt{2} + 2\sqrt{\lambda} + 2\sqrt{\mu}$ and $c = \sup_{0<t<1}\varphi(t)$ where

$$\varphi(t) = \frac{t^{\frac{1}{\lambda}}}{L} - \frac{2Ct}{1-t}, \qquad \text{where} \qquad L = \sqrt{\frac{2e}{\lambda}}e^{\frac{\mu}{2\lambda}}. \tag{2.15}$$

Proof. The proof follows the same argument as in the real case so we only sketch the proof here. In particular we point out the places where the estimates need to be modified.

Write $A = A_R + iA_I$ and set $B = [A_R, A_I]$. Then B is an $n \times 2N$ matrix whose entries are independent real standard normal random variables. It is easy to check that $\sigma_1(A) \leq \sqrt{2}\sigma_1(B)$. Thus by taking $C = 2\sqrt{\lambda} + \sqrt{2} + 2\sqrt{\mu}$ we have via (2.1) that

$$\mathbb{P}\left(\sigma_1(A) \leq C\sqrt{n}\right) \leq e^{-\mu n}. \tag{2.16}$$

The estimate for $\sigma_n(A)$ follows from the same strategy as in the real case. First of all, just like the real case for any $n \times n$ unitary matrix U the entries of UA are still independed complex standard normal random variables. As a result the probability $\mathbb{P}\left(\|A^*\mathbf{v}\| \leq \sqrt{\delta n}\right)$ where $\mathbf{v} \in \mathbb{C}^n$ is a unit vector does not depend on the choice of $\mathbf{v}$. By taking $\mathbf{v} = e_1$ we see

that ($\|A^*\mathbf{v}\|^2$ has the $\Gamma(N,1)$ distribution (as opposed to the $\Gamma(\frac{N}{2},1)$ distribution in the real case). Applying Lemma 2.1 we obtain the equivalent result for the complex case in

$$\mathbb{P}\left(\|A^*\mathbf{v}\| \leq \sqrt{\delta n}\right) \leq \left(\frac{2e\delta}{\lambda}\right)^N. \tag{2.17}$$

Next for the ε-net, we observe that the unit sphere in $\mathbb{C}^n$ is precisely the unit sphere in $\mathbb{R}^{2n}$ if we identify $\mathbb{C}^n$ as $\mathbb{R}^{2n}$. Thus we can find an ε-net $\mathcal{N}_\varepsilon$ of cardinality no more than $(1+2\varepsilon^{-1})^{2n}$. The proof of Theorem 1.2 now goes through with some minor modifications. The most important one is that with (2.16) and (2.17) the inequality condition (2.7) now becomes

$$-\frac{\mu}{2} \geq \ln(1+2\varepsilon^{-1}) + \frac{\lambda}{2}\Big(\ln 2e - \ln\lambda + 2\ln(c+\varepsilon C)\Big),$$

where the constant C is changed to $C = 2\sqrt{\lambda} + \sqrt{2} + 2\sqrt{\mu}$. Substituting this C and $\frac{\mu}{2}$ for μ we prove the theorem. ∎

3. Random Frames as NERFs

Our goal in this section is to establish the robustness of random frames against erasures by proving Theorem 1.1. Here we restate Theorem 1.1 in a a different form for the benefit of simpler notation in the proof.

Theorem 3.1. *Let $F = \frac{1}{\sqrt{n}}A$ where A is $n \times N$ whose entries are drawn independently from the standard normal $\mathcal{N}(0,1)$ distribution. Let $\lambda = \frac{N}{n} > 1$ and $K = pN = p\lambda n$ where $\lambda^{-1} < p \leq 1$. For any $\tau_0 > 0$ there exist constants $\alpha, \beta > 0$ depending only on λ, p and τ_0 such that F is a (K,α,β)-NERF with probability at least $1 - 3e^{-\tau_0 n}$.*

Proof. There exists exactly $\frac{N!}{K!(N-K)!}$ subsets $S \subseteq \{1,2,\ldots,N\}$ of cardinality $|S| = K$. It is well known that

$$\frac{N!}{K!(N-K)!} \leq \frac{N^N}{K^K(N-K)^{N-K}},$$

which can be shown easily by Stirling's Formula or induction on N. Set $s_p = p\ln p^{-1} + (1-p)\ln(1-p)^{-1}$, which has $0 \leq s_p \leq \ln 2$. We have then

$$\frac{N!}{K!(N-K)!} \leq \left(p^{-p}(1-p)^{p-1}\right)^N = e^{\lambda s_p n}. \tag{3.1}$$

Now we set $\mu := \lambda s_p + \tau_0$. Let $C = \sqrt{2\mu} + \sqrt{p\lambda} + 1$ and $c = \sup_{0<t<1}\varphi(t)$ where $\varphi(t)$ is given in (1.5). Let the columns of A be $\{\mathbf{a}_j\}_{j=1}^N$. For any $S \subseteq \{1,2,\ldots,N\}$ we denote by

A_S the submatrix of A whose columns are $\{\mathbf{f}_j : j \in S\}$. Then for $|S| = K = p\lambda n$ we have

$$\mathbb{P}\left(c\sqrt{n} \le \sigma_n(A_S) \le \sigma_1(A_S) \le C\sqrt{n}\right) \ge 1 - 3e^{-\mu n}.$$

by Theorem 1.2. It follows that

$$\begin{aligned}\mathbb{P}\Big(\sigma_n(A_S) \le c\sqrt{n} \text{ or } \sigma_1(A_S) \ge C\sqrt{n} \text{ for some } S \text{ with } |S| = K\Big) \\ \le \sum_{|S|=K} \mathbb{P}\Big(\sigma_n(A_S) \le c\sqrt{n} \text{ or } \sigma_1(A_S) \ge C\sqrt{n}\Big) \\ \le 3e^{(\lambda s_p - \mu)n} \;=\; 3e^{-\tau_0 n}.\end{aligned}$$

It follows that

$$\mathbb{P}\Big(c\sqrt{n} \le \sigma_n(A_S) \le \sigma_1(A_S) \le C\sqrt{n} \text{ for all } S \text{ with } |S| = K\Big) \ge 1 - 3e^{\tau_0 n}.$$

This implies that, by setting $\alpha = c$ and $\beta = C$, $F = \frac{1}{\sqrt{n}}A$ is a (K, α, β)-NERF with probability at least $1 - 3e^{-\tau_0 n}$. ∎

Theorems 1.1 and 3.1 states that random Gaussian frames can be robust with overwhelming probability against erasures of an arbitrary proportion of data from the original data, at least in theory, as long as the number of remaining vectors is at least $(1 + \delta_0)n$ for some $\delta_0 > 0$. In practice one may ask how good the condition numbers are if the erasures reach a high proportion, say, 90% of the data. We show some numerical results below.

Example 1. Let $F = \frac{1}{\sqrt{n}}A$ where A is $n \times N$ whose entries are independent standard normal random variables. Set $\tau_0 = 0.25$. In this experiment we fix $K = 2n$ and $K = 5n$, respectively, and let N vary. As N increases from $N = K$ to $N = 100K$ the proportion of erasure $s = 1 - \frac{K}{N}$ increases from 0 to 99%. We shall use β/α as a measure of robustness since it is an upper bound for the condition number. Clearly, as s increases we should expect β/α to increase. The left plot in Figure 1 shows $\log_2(\beta/\alpha)$ against s for both $K = 2n$ (top curve) and $K = 5n$ (bottom curve). Because the frame is normalized so that each column is on average a unit norm vector, it also makes sense to use the smallest singular value as a measurement of robustness. The right plot in Figure 1 shows $-\log_2(\alpha)$ against s also for both $K = 2n$ (top curve) and $K = 5n$ (bottom curve). Our numerical results show that in the case $K = 2n$, with probability at least $1 - 3e^{-0.5n}$, the condition number is no more than 10232 for 50% erasures and no more than 611675 for 90% erasures. In the case

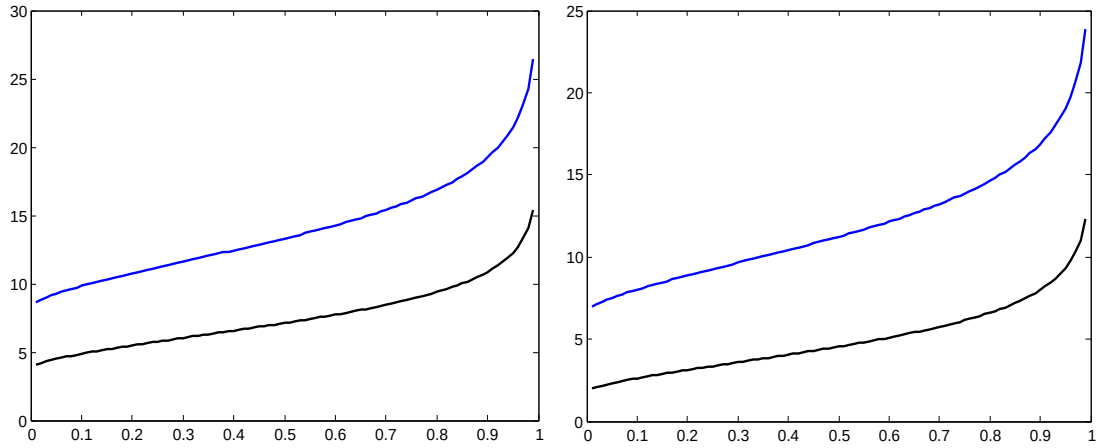


FIGURE 1. Left: $\log_2(\beta/\alpha)$ against the proportion of erasures when N varies from K to $100K$ while K is fixed at $K = 2n$ (top curve) and $K = 5n$ (bottom curve). Right: Same as in the left figure, but for $-\log_2(\alpha)$.

$K = 5n$, the corresponding numbers are 139.88 and 1862.1, respectively. In fact, even with 99% erasures the condition number is no more than 42716.

Example 2. Again we let $F = \frac{1}{\sqrt{n}}A$ where A is $n \times N$ whose entries are independent standard normal random variables, and let $\tau_0 = 0.25$. In this experiment we fix $N = 200n$ and $N = 50n$, respectively, and let K vary so the proportion of erasures $s = 1 - \frac{K}{N}$ varies from 0 to 99% ($N = 200n$ and 0 to 97% ($N = 50n$), respectively. Again we should expect the robustness to go down as we increase s. The left plot in Figure 2 shows $\log_2(\beta/\alpha)$ against s for $N = 50n$ (top curve) and $N = 200n$ (bottom curve). The right plot in Figure 2 shows $-\log_2(\alpha)$ against s also for both $N = 50n$ (top curve) and $N = 200n$ (bottom curve). Our numerical results show that in the case $N = 50n$, with probability at least $1 - 3e^{-0.5n}$, the condition number is no more than 31.7 for 50% erasures and 1862.1 for 90% erasures. In the case $N = 200n$, the corresponding numbers are 23.48 and 315.12, respectively. Even with 95% erasures the condition number is no more than 1312.4.

REFERENCES

[1] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3):253–263, 2008.

[2] E.J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9):589–592, 2008.

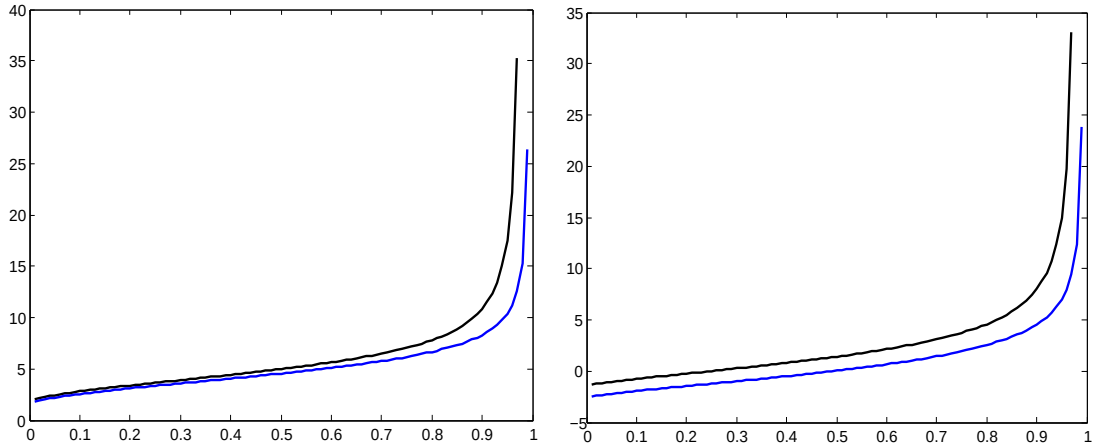


FIGURE 2. Left: $\log_2(\beta/\alpha)$ against the proportion of erasures when K varies while N is fixed at $N = 50n$ (top curve) and $N = 200n$ (bottom curve). Right: Same as in the left figure, but for $-\log_2(\alpha)$.

[3] E.J. Candes, Y. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *arXiv preprint arXiv:1109.0573*, 2011.

[4] E.J. Candes and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[5] E.J. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, 2006.

[6] P.G. Casazza and J. Kovačević. Equal-norm tight frames with erasures. *Advances in Computational Mathematics*, 18(2):387–430, 2003.

[7] A. Edelman. Eigenvalues and condition numbers of random matrices. *SIAM Journal on Matrix Analysis and Applications*, 9(4):543–560, 1988.

[8] M. Fickus, J. Jasper, D.G. Mixon, and J. Peterson. Group-theoretic constructions of erasure-robust frames. *arXiv preprint arXiv:1210.0139*, 2012.

[9] M. Fickus and D.G. Mixon. Numerically erasure-robust frames. *Linear Algebra and its Applications*, 2012.

[10] V.K. Goyal, J. Kovačević, and J.A. Kelner. Quantized frame expansions with erasures. *Applied and Computational Harmonic Analysis*, 10(3):203–233, 2001.

[11] R.B. Holmes and V.I. Paulsen. Optimal frames for erasures. *Linear algebra and its applications*, 377:31–51, 2004.

[12] J. Kovacevic, P.L. Dragotti, and V.K. Goyal. Filter bank frame expansions with erasures. *Information Theory, IEEE Transactions on*, 48(6):1439–1450, 2002.

[13] M. Puschel and J. Kovacevic. Real, tight frames with maximal robustness to erasures. In *Data Compression Conference, 2005. Proceedings. DCC 2005*, pages 63–72. IEEE, 2005.

[14] M. Rudelson and R. Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics*, 62(12):1707–1739, 2009.

[15] M. Rudelson and R. Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *arXiv preprint arXiv:1003.2990*, 2010.

[16] T. Strohmer and R.W. Heath. Grassmannian frames with applications to coding and communication. *Applied and Computational Harmonic Analysis*, 14(3):257–275, 2003.

[17] T. Tao and V. Vu. Random matrices: The distribution of the smallest singular values. *Geometric And Functional Analysis*, 20(1):260–297, 2010.

[18] R. Vershynin. Frame expansions with erasures: an approach through the non-commutative operator theory. *Applied and Computational Harmonic Analysis*, 18(2):167–176, 2005.
[19] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA

E-mail address: ywang@math.msu.edu

ON THE DECAY OF THE SMALLEST SINGULAR VALUE OF SUBMATRICES OF RECTANGULAR MATRICES

YANG LIU AND YANG WANG

Abstract. In this article, we study the decay of the smallest singular value of submatrices that consist of bounded column vectors. We find that that the smallest singular value of submatrices is related to the minimal distance of points to the lines connecting other two points in a bounded point set. Using a technique from integral geometry and from the perspective of combinatorial geometry, we show the decay rate of the minimal distance for the sets of points if the number of the points that are on the boundary of the convex hull of any subset is not too large, relative to the cardinality of the set. In the numeral or computational aspect, we conduct some numerical experiments for many sets of points and analyze the smallest distance for some extremal configurations.

1. Introduction

In recent decades, measurements, frames, and dictionaries (see for instance, [2], [24], and [5]), all of which are essentially matrices, have been studied and used in signal processing, such as compressed sensing, matrix recovery, phase retrieval, and other fields. As the main characteristics of a matrix or linear transformation, the singular values and their generalized forms have been studied in, for instance, [20], [18], [9], [28], and [23]. It is not hard to see that the singular values of a matrix are determined by both the magnitudes and the angles of the row vectors of the matrix.

Rectangular matrices are of the main interest in some recent research (see, for instance, [28] and [29]). Here we call a rectangular matrix a slim matrix if there are more rows than columns in the matrix. Considering the columns of a slim matrix as points in a bounded region in a plane, we show that the matrix problem can be reduced down to a combinatorial problem. If the magnitudes of all the rows of a rectangular matrix are bounded, we can estimate the smallest singular values of submatrices, in terms of the size of the matrix, because there are configurations of matrices whose minimal smallest singular values by the order of a power of the size with some negative exponent. Some estimates on the distances among points in a set or the distances from points to lines that connect other two points in a set of points in a bounded region are established in this article, and the decay rate of these distances, in some sense, essentially determines the the decay rate of the smallest singular values of submatrices with bounded column vectors. The combinatorial geometry problem is to related to Heilbronn's triangle problem (see, for instance, [16] and [4]). There have been some work on developing algorithms to find counter example for Heilbronn's original conjecture, but there does not appear to be any experimentable algorithm for one to find any explicit or concrete sets of points, and

2000 *Mathematics Subject Classification.* 35R30, 35R60, 35Q86, 94B75, 33F05.

Key words and phrases. matrix analysis, duality, singular values, combinatorial geometry.

it would be interesting to see the optimal arrangements of n points in a square or unit disk for Heilbronn's triangle problem and this problem respectively. However, we formulate a conjecture for a slower decay rate, which, as far as we know, is still open.

The main contribution of this paper is to show the connection between the singular value problem and a combinatorial geometry problem. Using a technique from integral geometry and from the perspective of combinatorial geometry, we show the decay rate of the minimal distance for the sets of points if the number of the points that are not on the boundary of the convex hull of any subset is not too large, relative to the cardinality of the set. We also obtain some other results regarding this combinatorial geometry problem in some cases, and so for the minimal smallest singular value of submatrices of rectangular matrices.

This paper is structured as follows: in Section 2, we prove some lemmas on the minimal smallest singular value of slim matrices, and particularly, we show the optimal decay rate for the base case; in Section 3, we prove a duality lemma for a the minimal smallest singular value of matrices of size $n+k$ by n; in Section 4, we undertake extensively studies on the minimal smallest singular value of matrices of size $n+3$ by n, and we obtain some results by using a technique from integral geometry and from the perspective of combinatorial geometry; and in Section 5, we present some numerical experimental results.

2. Some Lemmas on the Minimal Smallest Singular Value

First, we have the following lemma.

Lemma 2.1. *For any real matrix A of size N by n with $N \geq n$, one has*

$$\sigma_n(A) \geq \min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_n(A_S) \tag{2.1}$$

and

$$\sigma_1(A) \geq \min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_1(A_S) \tag{2.2}$$

Proof. For any $S \subseteq \{1,\ldots,n+1\}$ with $|S| = n$,

$$\sigma_n(A_S) = \inf_{v\in\mathbb{R}^n,\|v\|=1} \|A_S v\|; \tag{2.3}$$

and on the other hand,

$$\sigma_n(A) = \inf_{V\subseteq\mathbb{R}^n,\dim(V)=1} \|A|_V\| = \inf_{v\in\mathbb{R}^n,\|v\|=1} \|Av\|. \tag{2.4}$$

Since Av is basically an vector extension of$A_S v$ for every $v \in \mathbb{R}^n, \|v\| = 1$, then

$$\|A_S v\| \leq \|Av\| \tag{2.5}$$

for every $v \in \mathbb{R}^n, \|v\| = 1$. Thus, it follows from (2.3) and (2.4) that

$$\sigma_n(A_S) \leq \sigma_n(A) \tag{2.6}$$

for any $S \subseteq \{1,\ldots,n+1\}$ with $|S| = n$. Hence, we obtain (2.1), and similarly, we also obtain (2.2). □

From the growth rate of the smallest singular value of random matrices established in [3], one can obtain that

$$\sigma_n(A) \to \left(2-\sqrt{2}\right)\sqrt{n} \tag{2.7}$$

for $N = 2n$. On the other hand,

$$\sigma_n(A_S) \leq O\left(\frac{1}{\sqrt{n}}\right). \tag{2.8}$$

Lemma 2.2. *For any $n+1$ by n matrix $A = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_{n+1} \end{bmatrix}$ with $\|\mathbf{a}_i\| \leq 1$, $i = 1, \ldots, n+1$, one has*

$$\min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_n(A_S) \leq \frac{1}{\sqrt{n}} \tag{2.9}$$

Proof. Since $\mathbf{a}_1$, ..., $\mathbf{a}_{n+1}$ are linear dependent, there are c_1, ..., c_{n+1}, such that

$$\sum_{i=1}^{n+1} c_i \mathbf{a}_i = 0 \tag{2.10}$$

with

$$\sum_{i=1}^{n+1} c_i^2 = 1. \tag{2.11}$$

Without loss of generality, assume $c_{n+1} = \min(c_1, \ldots, c_{n+1})$. If $c_{n+1} = 0$, (2.9) is trivial, because there is an S such that A_S is singular. It suffices to consider the case of $c_{n+1} \neq 0$. Therefore,

$$c_{n+1}\mathbf{a}_{n+1} = -\sum_{i=1}^{n} c_i \mathbf{a}_i \tag{2.12}$$

By (2.11),

$$(n+1)\, c_{n+1}^2 \leq \sum_{i=1}^{n+1} c_i^2 = 1. \tag{2.13}$$

It follows that

$$|c_{n+1}| \leq \frac{1}{\sqrt{n+1}} \tag{2.14}$$

and furthermore

$$\frac{\|c_{n+1}\mathbf{a}_{n+1}\|}{\sqrt{1-c_{n+1}^2}} \leq \frac{1}{\sqrt{n+1}} \cdot \frac{\sqrt{n+1}}{\sqrt{n}} = \frac{1}{\sqrt{n}}. \tag{2.15}$$

Since

$$\min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_n(A_S) \leq \frac{\|\sum_{i=1}^n c_i \mathbf{a}_i\|}{\sqrt{\sum_{i=1}^n c_i^2}} = \frac{\|c_{n+1}\mathbf{a}_{n+1}\|}{\sqrt{1-c_{n+1}^2}}, \tag{2.16}$$

thus (2.9) follows from (2.15). □

Remark 2.3. However, one can have

$$\min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_n(A_S) > \frac{1}{n} \tag{2.17}$$

for some matrix A. For example,

$$A^T = \begin{bmatrix} 0.9969 & 0.6688 & 0.1610 \\ -0.0782 & 0.7434 & -0.9870 \end{bmatrix}, \tag{2.18}$$

we have

$$\min_{S\subseteq\{1,\ldots,n+1\},|S|=n} \sigma_n(A_S) = 0.6115 > \frac{1}{2}. \tag{2.19}$$

For matrices of size $n+1$ by n, one can have the following

Lemma 2.4. *For any $n+2$ by n matrix $A = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_{n+2} \end{bmatrix}$ with $\|\mathbf{a}_i\| \leq 1$, $i = 1,\ldots,n+2$, one has*

$$\min_{S\subseteq\{1,\ldots,n+2\},|S|=n} \sigma_n(A_S) \leq \frac{C}{n^{3/2}} \tag{2.20}$$

for some constant $C > 0$.

Proof. It suffices to consider matrices of size $n+2$ by n with rank no less than n. Then for any $\mathbf{z} \in \ker(A)$ with $\|z\| = 1$, we have

$$\begin{aligned} \sigma_n(A_S) &\leq \inf_{\mathbf{z}\in\ker(A)} \frac{\|A_S\mathbf{z}_S\|}{\|\mathbf{z}_S\|} \\ &= \inf_{\mathbf{z}\in\ker(A)} \frac{\left\|z_{i_1}\mathbf{a}_{i_1}+z_{i_2}\mathbf{a}_{i_2}\right\|}{\|\mathbf{z}_S\|} \\ &\leq \inf_{\mathbf{z}\in\ker(A)} \frac{|z_{i_1}|\|\mathbf{a}_{i_1}\|+|z_{i_2}|\|\mathbf{a}_{i_2}\|}{\|\mathbf{z}_S\|} \\ &\leq \inf_{\mathbf{z}\in\ker(A)} \frac{|z_{i_1}|+|z_{i_2}|}{\|\mathbf{z}_S\|} \\ &\leq \inf_{\mathbf{z}\in\ker(A)} \frac{\sqrt{2}\sqrt{z_{i_1}^2+z_{i_2}^2}}{\sqrt{1-\left(z_{i_1}^2+z_{i_2}^2\right)}} \end{aligned} \tag{2.21}$$

where $S = \{1,\ldots,n+2\} \setminus \{i_1, i_2\}$ for all $1 \leq i_1, i_2 \leq n+2$.

Let $\mathbf{b}_1$ and $\mathbf{b}_2$ be an orthonormal basis of $\ker(A)$, $\mathbf{b}_1 = (b_{11},\ldots,b_{1,n+2})$ and $\mathbf{b}_2 = (b_{21},\ldots,b_{2,n+2})$, and denote $\begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix} := B$. Since $\mathbf{z} \in \ker(A)$ with $\|\mathbf{z}\| = 1$, there exist t_1 and t_2 such that

$$\mathbf{z} = t_1\mathbf{b}_1 + t_2\mathbf{b}_2 \tag{2.22}$$

with $t_1^2 + t_2^2 = 1$. Therefore,

$$\begin{aligned} \sqrt{z_{i_1}^2 + z_{i_2}^2} &= \sqrt{\left(t_1 b_{1,i_1} + t_2 b_{2,i_1}\right)^2 + \left(t_1 b_{1,i_2} + t_2 b_{2,i_2}\right)^2} \\ &= \|(t_1,t_2)\, B_{S^c}\| \end{aligned} \tag{2.23}$$

Combining (2.21), we have

$$\sigma_n(A_S) \leq C \inf_{t_1^2+t_2^2=1} \|(t_1,t_2)\, B_{S^c}\| = C\sigma_2(B_{S^c}) \tag{2.24}$$

for some constant $C > 0$ and furthermore,

$$\min_{S\subseteq\{1,\ldots,n+2\},|S|=n} \sigma_n(A_S) \leq C \min_{S\subseteq\{1,\ldots,n+2\},|S|=n} \sigma_2(B_{S^c}). \tag{2.25}$$

Now let $B = (\beta_1,\ldots,\beta_{n+2})$ and normalize the columns of B, then

$$\sigma_2(B_{S^c}) \leq \max\left(\|\beta_{i_1}\|, \|\beta_{i_2}\|\right) \sigma_2\left(\left(\frac{\beta_{i_1}}{\|\beta_{i_1}\|}, \frac{\beta_{i_2}}{\|\beta_{i_2}\|}\right)\right). \tag{2.26}$$

Now we can choose the indices i_1 and i_2, $1 \le i_1, i_2 \le n+2$, such that

$$b_{1,i_1}^2 + b_{2,i_1}^2 + b_{1,i_2}^2 + b_{2,i_2}^2 = \|B_{S^c}\|_F \tag{2.27}$$

is the smallest among all pairs of indices between 1 and $n+2$, but since

$$\sum_{i=1}^{n+2} b_{1,i}^2 + \sum_{i=1}^{n+2} b_{1i}^2 = 2, \tag{2.28}$$

we have

$$b_{1,i_1}^2 + b_{2,i_1}^2 + b_{1,i_2}^2 + b_{2,i_2}^2 \le \frac{4}{n+2}, \tag{2.29}$$

which implies

$$\max\left(\sqrt{b_{1,i_1}^2 + b_{2,i_1}^2}, \sqrt{b_{1,i_2}^2 + b_{2,i_2}^2}\right) \le \frac{2}{\sqrt{n+2}}. \tag{2.30}$$

Therefore, by (2.26), we have

$$\begin{array}{rcl} \min_{S\subseteq\{1,\ldots,n+2\},|S|=n} \sigma_2\left(B_{S^c}\right) & \le & \frac{2}{\sqrt{n+2}}\sigma_2\left(\left(\frac{\beta_{i_1}}{\|\beta_{i_1}\|}, \frac{\beta_{i_2}}{\|\beta_{i_2}\|}\right)\right) \\ & \le & \frac{\sqrt{2}}{\sqrt{n+2}}\left\|\frac{\beta_{i_1}}{\|\beta_{i_1}\|} - \frac{\beta_{i_2}}{\|\beta_{i_2}\|}\right\|. \end{array} \tag{2.31}$$

Considering the geometry of $n+2$ vectors on the unit circle and choose the closest two vectors among the $n+2$ unit vectors, we know

$$\left\|\frac{\beta_{i_1}}{\|\beta_{i_1}\|} - \frac{\beta_{i_2}}{\|\beta_{i_2}\|}\right\| \le 2\sin\frac{\pi}{n+2}. \tag{2.32}$$

Next, we will show the following inequality

$$\min_{S\subseteq\{1,\ldots,n+2\},|S|=n} \sigma_2\left(B_{S^c}\right) \le \frac{2\sqrt{2}}{\sqrt{n+2}}\sin\frac{\pi}{n+2}. \tag{2.33}$$

Suppose that

$$\sigma_2\left(B_{S^c}\right) \ge \frac{2\sqrt{2}}{\sqrt{n+2}}\sin\frac{\pi}{n+2} \tag{2.34}$$

for all $S \subseteq \{1,\ldots,n+2\}$ with $|S| = n$. For any

$$B_{S^c} = (\beta_i, \beta_j), \tag{2.35}$$

we have

$$\begin{array}{rcl} \sigma_2\left(B_{S^c}\right) & \le & \frac{\left\|\frac{\beta_i}{\|\beta_i\|} - \frac{\beta_j}{\|\beta_j\|}\right\|}{\sqrt{\frac{1}{\|\beta_i\|^2} + \frac{1}{\|\beta_j\|^2}}} \\ & = & \frac{\|\beta_i\|\|\beta_j\|\left\|\frac{\beta_i}{\|\beta_i\|} - \frac{\beta_j}{\|\beta_j\|}\right\|}{\sqrt{\|\beta_i\|^2 + \|\beta_j\|^2}} \\ & \le & \min\left(\|\beta_i\|, \|\beta_j\|\right)\left\|\frac{\beta_i}{\|\beta_i\|} - \frac{\beta_j}{\|\beta_j\|}\right\| \end{array} \tag{2.36}$$

for $1 \le i < j \le n+2$. We can actually arrange the indices in β_i, $1 \le i \le n+2$, so that $\frac{\beta_i}{\|\beta_i\|}$, $1 \le i \le n+2$, are in the counterclockwise order in the unit disk. By (2.28), we know that

$$\sum_{i=1}^{n+2} \|\beta_i\|^2 = 2, \tag{2.37}$$

and since any chord is shorter than its corresponding arc on a circle, we have that

$$\sum_{i=1}^{n+2}\left\|\frac{\beta_i}{\|\beta_i\|}-\frac{\beta_{i+1}}{\|\beta_{i+1}\|}\right\|\leq 2\pi, \tag{2.38}$$

assuming $\beta_{n+3}=\beta_1$. From (2.36),

(2.39)
$$\begin{array}{rcl}\min\limits_{S\subseteq\{1,\ldots,n+2\},|S|=n}\sigma_2\left(B_{S^c}\right) & \leq & \min_{1\leq i<j\leq n+2}\left(\min\left(\|\beta_i\|,\|\beta_j\|\right)\left\|\frac{\beta_i}{\|\beta_i\|}-\frac{\beta_j}{\|\beta_j\|}\right\|\right)\\ & \leq & \min_{1\leq i\leq n+2}\left(\|\beta_i\|\left\|\frac{\beta_i}{\|\beta_i\|}-\frac{\beta_{i+1}}{\|\beta_{i+1}\|}\right\|\right)\end{array}$$

From (2.37) and (2.38), one can obtain that

(2.40)
$$\begin{array}{rcl}\min_{1\leq i\leq n+2}\left(\|\beta_i\|\left\|\frac{\beta_i}{\|\beta_i\|}-\frac{\beta_{i+1}}{\|\beta_{i+1}\|}\right\|\right) & \leq & \frac{1}{n+2}\sum_{i=1}^{n+2}\left(\|\beta_i\|\left\|\frac{\beta_i}{\|\beta_i\|}-\frac{\beta_{i+1}}{\|\beta_{i+1}\|}\right\|\right)\\ & \leq & \frac{2\sqrt{2}\pi}{(n+2)^{3/2}}\end{array}$$

and then (2.20) follows. □

3. Duality Lemma for matrices of size $n+k$ by n

First we have the following duality lemma in general.

Lemma 3.1. *For any matrix A of size m by n with $m\geq n$ with all rows normalized to 1, one has*

$$\min_{S\subseteq\{1,\ldots,m\},|S|=n}\sigma_n\left(A_S\right)\leq C\min_{T\subseteq\{1,\ldots,m\},|T|=m-n}\sigma_n\left(B_T\right) \tag{3.1}$$

for some constant $C>0$, where B consists of any orthogonal basis of $\ker\left(A\right)$.

Proof. Then for any $\mathbf{z}\in\ker\left(A\right)$ with $\|z\|=1$, we have

$$\begin{array}{rcl}\sigma_n\left(A_S\right) & \leq & \inf_{\mathbf{z}\in\ker(A)}\frac{\|A_S\mathbf{z}_S\|}{\|\mathbf{z}_S\|}\\ & = & \inf_{\mathbf{z}\in\ker(A)}\frac{\left\|z_{i_1}\mathbf{a}_{i_1}+z_{i_2}\mathbf{a}_{i_2}\right\|}{\|\mathbf{z}_S\|}\\ & \leq & \inf_{\mathbf{z}\in\ker(A)}\frac{\left|z_{i_1}\right|\left\|\mathbf{a}_{i_1}\right\|+\left|z_{i_2}\right|\left\|\mathbf{a}_{i_2}\right\|}{\|\mathbf{z}_S\|}\\ & \leq & \inf_{\mathbf{z}\in\ker(A)}\frac{\|z_{S^c}\|_1}{\|\mathbf{z}_S\|}\\ & \leq & \inf_{\mathbf{z}\in\ker(A)}\frac{\sqrt{2}\|z_{S^c}\|_2}{\sqrt{1-\|z_{S^c}\|_2^2}}.\end{array} \tag{3.2}$$

Let $\mathbf{b}_1$ and $\mathbf{b}_2$ be an orthonormal basis of $\ker\left(A\right)$, $\mathbf{b}_1=(b_{11},\ldots,b_{1,n+2})$ and $\mathbf{b}_2=(b_{21},\ldots,b_{2,n+2})$, and denote $\begin{pmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{pmatrix}:=B$. Since $\mathbf{z}\in\ker\left(A\right)$ with $\|\mathbf{z}\|=1$, there exist t_1 and t_2 such that

$$\mathbf{z}=\mathbf{t}B_{S^c} \tag{3.3}$$

with $t_1^2+t_2^2=1$. Therefore,

$$\|z_{S^c}\|_2 \quad = \quad \|\mathbf{t}B_{S^c}\| \tag{3.4}$$

Combining (3.2), we have

$$\sigma_n\left(A_S\right)\leq C\inf_{\mathbf{t}\in\mathbb{S}^{m-n}}\|\mathbf{t}B_{S^c}\|=C\sigma_{m-n}\left(B_{S^c}\right) \tag{3.5}$$

for some constant $C > 0$, where $\mathbb{S}^{m-n}$ is the unit sphere in $\mathbb{R}^{m-n+1}$, and furthermore,

$$\min_{S\subseteq\{1,\ldots,m\},|S|=n} \sigma_n\left(A_S\right) \leq C \min_{T\subseteq\{1,\ldots,m\},|T|=m-n} \sigma_n\left(B_T\right). \tag{3.6}$$

□

Remark 3.2. In matrix theory and operator theory, the image of an operator is regards as to be dual its kernel or null space. Here this duality is in a similar essence to relationship between the restricted isometry property, Johnson-Lindenstrauss embedding, and the null space property in signal processing, including compressed sensing, phase retrieval, and others (see for instance, [27], [17], and [26]).

4. Decay rate for matrices of size $n+3$ by n

Let $P_1, \ldots, P_n$ be in the unit disk on the plane, and $d(i,j,k)$ be the distance of the point P_i to the line connecting other two points P_j and P_k , $1 \leq i,j,k \leq n$. In this section, we want to study the decay of $\min_{1\leq i,j,k\leq n} d(i,j,k)$, as $n \to \infty$.

First, let us prove the following lemma on the decay order of at least $O\left(\frac{1}{n}\right)$.

Lemma 4.1. *Let P_1, ..., P_n be a set of points in the unit disk on the plane. Suppose that P_1, ..., P_n are on the boundary of the convex hull of the point set $\{P_1, \ldots, P_n\}$ and $d(i,j,k)$ is the distance of the point P_i to the line connecting other two points P_j and P_k, $1 \leq i,j,k \leq n$, then*

$$\min_{1\leq i,j,k\leq n} d(i,j,k) \leq \frac{C}{n} \tag{4.1}$$

for some absolute constant C, $C > 0$, independent of n.

Proof. Let us cover the unit disk by parallel stripes of width $\frac{8}{n}$, then the unit disk can be covered by $\left\lceil \frac{n}{4} \right\rceil$ such stripes. By the pigeonhole principle, there exist at least 3 points P_{i_0}, $P_{j_0}$and P_{k_0}which locate in the same strip, thus we have

$$\min_{1\leq i,j,k\leq n} d(i,j,k) \leq d(i_0,j_0,k_0) \leq \frac{8}{n}. \tag{4.2}$$

□

Next, we prove the following lemma.

Lemma 4.2. *Let P_1, ..., P_n be a set of points in the unit disk on the plane. Suppose that P_1, ..., P_n are on the boundary of the convex hull of the point set $\{P_1, \ldots, P_n\}$ and $d(i,j,k)$ is the distance of the point P_i to the line connecting other two points P_j and P_k, $1 \leq i,j,k \leq n$, then*

$$\min_{1\leq i,j,k\leq n} d(i,j,k) \leq \frac{C}{n^2} \tag{4.3}$$

for some absolute constant C, $C > 0$, independent of n.

Proof. Without loss of generality, we assume that the points P_1, P_2, ..., and P_n are in the counterclockwise order in the unit disk. Firstly, if P_1, ..., P_n are the vertices of a convex polygon $\mathbf{P}$, then by the Crofton formula in integral geometry or geometric probability (see for instance [15], [22], and [30]),

$$\text{perimeter}\left(\mathbf{P}\right) = \frac{1}{2}\int_0^{2\pi}\int_0^1 n_{\mathbf{P}}\left(\theta, r\right) dr d\theta, \tag{4.4}$$

where $n_P(\theta, r)$ is the intersection number of the the polygon and the oriented line which has a distance r to the origin and has an angle θ to the positive horizontal axis. Let $\mathbf{C}$ be the unit circle, again by the Crofton formula, we know

$$\text{perimeter}(\mathbf{C}) = \frac{1}{2}\int_0^{2\pi}\int_0^1 n_{\mathbf{C}}(\theta, r)\, dr d\theta. \tag{4.5}$$

But since the polygon $\mathbf{P}$ is convex, then

$$n_{\mathbf{P}}(\theta, r) \le 2 = n_{\mathbf{C}}(\theta, r), \tag{4.6}$$

and it follows from (4.4) and (4.5) that

$$\text{perimeter}(\mathbf{P}) \le \text{perimeter}(\mathbf{C}) = \frac{1}{2}\int_0^{2\pi}\int_0^1 2 dr d\theta = 2\pi. \tag{4.7}$$

Thus the sum of the boundary edges of the polygon

$$\sum_{i=1}^{n} \left\|\overline{P_iP_{i+1}}\right\| \le 2\pi. \tag{4.8}$$

Now let us connect the vertices by edges $\overline{P_1P_3}$, $\overline{P_2P_4}$, $\ldots$, $\overline{P_{n-1}P_1}$, and $\overline{P_nP_2}$, then we have

$$\sum_{i=1}^{n}\left(\angle P_{i+2}P_iP_{i+1} + \angle P_{i+1}P_{i+2}P_i\right) = n\pi - (n-2)\pi = 2\pi \tag{4.9}$$

assuming $P_{n+1} = P_1$ and $P_{n+2} = P_2$, because there are n triangles and the sum of the interior angles of the polygon is $(n-2)\pi$. Furthermore, since

$$\sum_{i=1}^{n}\left(\sin\left(\angle P_{i+2}P_iP_{i+1}\right) + \sin\left(\angle P_{i+1}P_{i+2}P_i\right)\right) \le \sum_{i=1}^{n}\left(\angle P_{i+2}P_iP_{i+1} + \angle P_{i+1}P_{i+2}P_i\right) \tag{4.10}$$

therefore, we have

$$\sum_{i=1}^{n}\left(\sin\left(\angle P_{i+2}P_iP_{i+1}\right) + \sin\left(\angle P_{i+1}P_{i+2}P_i\right)\right) \le 2\pi. \tag{4.11}$$

By Cauchy–Schwarz inequality,

$$\begin{aligned} &\textstyle\sum_{i=1}^{n}\left\|\overline{P_iP_{i+1}}\right\|^{\frac{1}{2}}\left(\sin^{\frac{1}{2}}\left(\angle P_{i+2}P_iP_{i+1}\right) + \sin^{\frac{1}{2}}\left(\angle P_{i+1}P_{i+2}P_i\right)\right) \\ \le\ &\left(\textstyle\sum_{i=1}^{n} 2\left\|\overline{P_iP_{i+1}}\right\|\right)\left(\sum_{i=1}^{n}\left(\sin\left(\angle P_{i+2}P_iP_{i+1}\right) + \sin\left(\angle P_{i+1}P_{i+2}P_i\right)\right)\right). \end{aligned} \tag{4.12}$$

It follow from (4.8) and (4.11) that

$$\sum_{i=1}^{n}\left\|\overline{P_iP_{i+1}}\right\|^{\frac{1}{2}}\left(\sin^{\frac{1}{2}}\left(\angle P_{i+2}P_iP_{i+1}\right) + \sin^{\frac{1}{2}}\left(\angle P_{i+1}P_{i+2}P_i\right)\right) \le 4\pi\cdot 2\pi = 8\pi^2. \tag{4.13}$$

Since there are actually $2n$ terms in the above sum, then we have

$$\min_{1\le i\le n}\left\|\overline{P_iP_{i+1}}\right\|^{\frac{1}{2}}\sin^{\frac{1}{2}}\left(\angle P_{i+2}P_iP_{i+1}\right) \le \frac{8\pi^2}{2n} = \frac{4\pi^2}{n} \tag{4.14}$$

or

$$\min_{1\le i\le n}\left\|\overline{P_iP_{i+1}}\right\|^{\frac{1}{2}}\sin^{\frac{1}{2}}\left(\angle P_{i+1}P_{i+2}P_i\right) \le \frac{8\pi^2}{2n} = \frac{4\pi^2}{n}. \tag{4.15}$$

Notice that

(4.16)
$$d(i+1,i,i+2) = \left\|\overline{P_iP_{i+1}}\right\| \sin\left(\angle P_{i+2}P_iP_{i+1}\right) = \left\|\overline{P_{i+1}P_{i+2}}\right\| \sin\left(\angle P_{i+1}P_{i+2}P_i\right),$$

thus by (4.14) and (4.15) we know that

$$\min_{1\le i\le n} d(i+1,i,i+2) \le \frac{16\pi^4}{n^2} \tag{4.17}$$

and the claim in Lemma 4.2 follows, in the case that P_1, ..., P_n are the vertices of a convex polygon.

Secondly, if a point is on the boundary edges of a convex hull of the point set but is not a vertices of the convex polygon, then the distance of the point to the edge which the point is on is zero. Thus the claim in Lemma 4.2 automatically holds in this case. □

Remark 4.3. In the proof of the above lemma, we have used a technique from integral geometry. For generalized theory of it, one can refer to, for instance, [31], [10], [21], and [1].

From this lemma, we can derive the following corollary immediately.

Lemma 4.4. *Let P_1, ..., P_n be a set of points in the unit disk on the plane. Suppose that P_{i_1}, ..., $P_{i_{n-s}}$, $0 \le s \le n-4$, are on the boundary of the convex hull of the point set $\left\{P_{i_1},\ldots,P_{i_{n-s}}\right\}$ and $d(i,j,k)$ is the distance of the point P_i to the line connecting other two points P_j and P_k, $1 \le i,j,k \le n$, then*

$$\min_{1\le i,j,k\le n} d(i,j,k) \le \frac{C}{(n-s)^2} \tag{4.18}$$

for some absolute constant C, $C>0$, independent of n. In particular, if $s \le \left\lfloor \frac{n}{2} \right\rfloor$, we have

$$\min_{1\le i,j,k\le n} d(i,j,k) \le \frac{4C}{n^2}. \tag{4.19}$$

More generally, if $s \le \lfloor \lambda n \rfloor$ for some absolute constant λ, $\lambda>0$, independent of n, then

$$\min_{1\le i,j,k\le n} d(i,j,k) \le \frac{C}{n^2}. \tag{4.20}$$

for some absolute constant C, $C>0$, independent of n.

Remark 4.5. Note that $s \le n-4$, because by the Sylvester–Gallai theorem (see for instance [6] and [14]), if all the points are not collinear, there is a line which passes through exactly two of the points, but (4.3) will trivially hold if there exist three points in the point set that are colinear and here we only need to consider the sets of n points which have exactly $\frac{n(n-1)}{2}$ ordinary lines, on which one can refer to [11], and also by the Erdős–Szekeres theorem (see for instance [8] and [25]), any set of n generic points, $n \ge 4$, in the plane has at least 4 points that are the vertices of a convex quadrilateral.

In [7] and [13], a set of 2^{n-2} points that contains no convex n-gon was constructed. We will analyze the minimal distance $\min_{1\le i,j,k\le n} d(i,j,k)$ for this extremal case. Let

$$S_{k,l} := \left\{ \left(x, y_{k,l}\left(x\right)\right) : 1 \le x \le \binom{k+l-2}{k-1} \right\} \tag{4.21}$$

and define $y_{k,l}(x)$ inductively as follows:

(1) $y_{k,1}(1) = y_{1,l}(1) = 1$;
(2) if $k > 1$, $l > 1$, then

$$y_{k,l}(x) = y_{k,l-1}(x) \tag{4.22}$$

for $1 \le x \le \binom{k+l-3}{k-1}$ and

$$y_{k,l}(x) = y_{k-1,l}\left(x - \binom{k+l-3}{k-1}\right) + \alpha_{k,l} \tag{4.23}$$

for $\binom{k+l-3}{k-1} < x \le \binom{k+l-2}{k-1}$, where

$$\alpha_{k,l} = \binom{k+l-2}{k-1} \max\left(y_{k,l-1}\left(\binom{k+l-3}{k-1}\right), y_{k-1,l}\left(\binom{k+l-3}{k-2}\right)\right). \tag{4.24}$$

From the inductive definition of $y_{k,l}(x)$, we know that $y_{k,l}$ linearly depends on $y_{k,l-1}$ and $y_{k-1,l}$. By [13], $y_{k,l}(x)$ is monotone increasing with respect to x for $1 \le x \le \binom{k+l-3}{k-1}$. But $y_{k,l}(x)$ increases dramatically when x becomes large.

Now let us consider $S_{n,n}$, the cardinality of $S_{n,n}$

$$|S_{n,n}| = \binom{2n-2}{n-1}. \tag{4.25}$$

To preserve the convexity and concavity of subsets in$S_{n,n}$ and confine it into the unit square, we use a similarity transformation

$$\mathcal{T} = \begin{pmatrix} \frac{((n-1)!)^2}{(2n-2)!} & 0 \\ 0 & \frac{1}{y_{n,n}\left(\binom{2n-2}{n-1}\right)} \end{pmatrix}, \tag{4.26}$$

and then $\mathcal{T}(S_{n,n}) \subset [0,1]^2$. Since $S_{n,n}$ is one of the components of the set of $N = 2^{2n-2}$ points R_N that contains no convex n-gon, $\mathcal{T}(S_{n,n})$ is the one of the components of the set of $N = 2^{2n-2}$ points in $[0,1]^2$ that contains no convex n-gon. From the figure 4.1, we can see that the minimal distance $\min_{1\le i,j,k\le n} d(i,j,k)$ in $S_{n,n}$ multiplied by $N^2 = 2^{4n-4}$ is very likely bounded, that implies the minimal distance $\min_{1\le i,j,k\le n} d(i,j,k)$ in the set of $N = 2^{2n-2}$ points R_N should decay at the rate of at least $O\left(\frac{1}{N^2}\right)$.

Considering the configurations of n points in the unit disk, we have the following lemma first.

Lemma 4.6. *Let $\mathbb{D}$ be the unit disk, then*

$$\min_{1\le i,j,k\le n} d(v_i, v_j, v_k) \le 2\sin^2\frac{\pi}{n} \tag{4.27}$$

for all $v_1, v_2, \ldots\ldots, v_n \in \mathbb{D}$ for $n = 3$ and 4. Therefore

$$\min_{1\le i,j,k\le n} d(v_i, v_j, v_k) \le \frac{2\pi^2}{n^2} \tag{4.28}$$

for all $v_1, v_2, \ldots\ldots, v_n \in \mathbb{D}$ for $n = 3$ and 4.

Proof. For $n = 3$, there are three points v_1, v_2 and v_3 in $\mathbb{D}$. Without loss of generality, we can assume that the side $\overline{v_1v_2}$ is the longest side and v_1 and v_2 lie on the boundary of $\mathbb{D}$, denoted as $\partial\mathbb{D}$, because one can use translations and rotations.

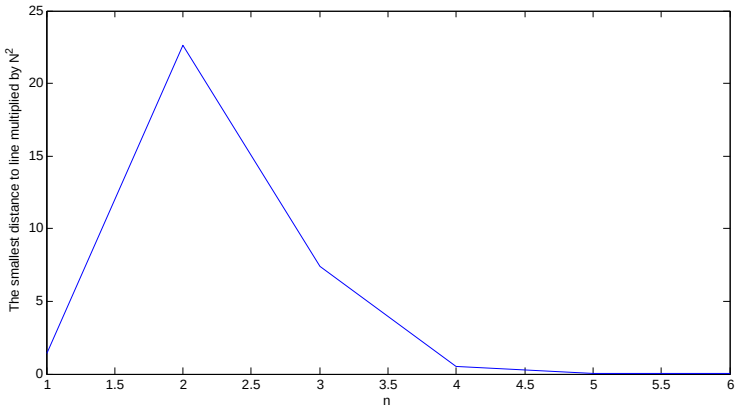


FIGURE 4.1. Plotted above are the smallest distances in $S_{n,n}$ multiplied by 2^{4n-4}.

Let v_3' be the intersection of the line parallel to the side $\overline{v_1v_2}$ and its perpendicular bisector. Then we have

$$d(v_3', v_1, v_2) = d(v_3, v_1, v_2) \tag{4.29}$$

and then the minimal heights of the triangle $\triangle v_1v_2v_3$ and $\triangle v_1v_2v_3'$ are equal, because

$$\left\|\overrightarrow{v_1v_3'}\right\| = \left\|\overrightarrow{v_2v_3'}\right\| \leq \max\left(\left\|\overrightarrow{v_2v_3}\right\|, \left\|\overrightarrow{v_1v_3}\right\|\right) \leq \left\|\overrightarrow{v_1v_2}\right\|, \tag{4.30}$$

in other words, $\overline{v_1v_2}$ is also the longest side of $\triangle v_1v_2v_3'$, and the areas of the triangle $\triangle v_1v_2v_3$ and $\triangle v_1v_2v_3'$ are equal.

Now, let us move v_3' along the perpendicular bisector of the side $\overline{v_1v_2}$ towards the direction in which the height increases, until it touches the boundary $\partial\mathbb{D}$ at a point denoted by v_3''. Let the distance from a point $v_3(t)$ on the perpendicular bisector of the side $\overline{v_1v_2}$ to the side $\overline{v_1v_2}$ be t, then the minimal height of the triangle $\triangle v_1v_2v_3(t)$

(4.31)

$$\frac{t\left\|\overrightarrow{v_1v_2}\right\|}{\max\left(\left\|\overrightarrow{v_1v_2}\right\|, \sqrt{t^2+\left(\frac{\left\|\overrightarrow{v_1v_2}\right\|}{2}\right)^2}\right)} = \begin{cases} t, & 0 < t < \frac{\sqrt{3}\left\|\overrightarrow{v_1v_2}\right\|}{2} \\ \frac{t\left\|\overrightarrow{v_1v_2}\right\|}{\sqrt{t^2+\left(\frac{\left\|\overrightarrow{v_1v_2}\right\|}{2}\right)^2}}, & t \geq \frac{\sqrt{3}\left\|\overrightarrow{v_1v_2}\right\|}{2} \end{cases}$$

increases as t increases. Thus the minimal height of the triangle $\triangle v_1v_2v_3''$ is greater than or equal to that of the triangle $\triangle v_1v_2v_3'$.

Then, we can do a regularization for the $\triangle v_1v_2v_3''$ whose vertices all lie on $\partial\mathbb{D}$. If one of the vertices does not bisect the arc ending with the other two vertices, and without loss of generality, we can assume that v_3'' does not bisect the arc ending with v_1 and v_2, then move v_3'' to the midpoint of the are, and then the new triangle lying on $\partial\mathbb{D}$ has a great minimal height, by comparing trigonometric functions. Thus, the equilateral triangle lying on $\partial\mathbb{D}$ has the greatest minimal height. This finishes the proof for the case of $n = 3$.

For $n = 4$, there are two cases to consider, but we will be able to find the maximum of the minimal heights for both cases. The first case is that one of the four points is in the interior of the convex hall of the other three points. Let's assume that, v_4 is in the interior of the convex hall of the other three points v_1, v_2 and v_3. Then if we fix v_1, v_2 and v_3, the maximum of the minimal heights for this case is reached when v_4 is at the center of the incircle of the triangle $\triangle v_1v_2v_3$, because otherwise, the minimal height $\min_{1\leq i,j,k\leq 4} d(v_i, v_j, v_k)$ would be less than the radius of the incircle of the triangle $\triangle v_1v_2v_3$. Using an argument similar to the case of $n = 3$, we can show that in this case,

$$\min_{1\leq i,j,k\leq 4} d(v_i, v_j, v_k) \leq \frac{1}{2} \leq 2\sin^2\frac{\pi}{4}. \tag{4.32}$$

The second case is that the four points are all on the boundary of the convex hull of the point set $\{v_1, v_2, v_3, v_4\}$. One can always find a rectangle R inside the quadrilateral which has the same minimal height of the triangles of the rectangle R as the minimal height of the triangles of the quadrilateral. By translations and dilations, one can obtain another rectangle R' on $\partial\mathbb{D}$ of which the minimal height of the triangles is no less than minimal height of the triangles of the rectangle R. Through maximizing a simple function, one can get that

$$\min_{1\leq i,j,k\leq 4} d(v_i, v_j, v_k) \leq 2\sin^2\frac{\pi}{4} \tag{4.33}$$

in this case.

In general, if all the points are on the boundary of the convex hull of the point set $\{v_1, v_2, \ldots, v_n\}$, we have □

Lemma 4.7. *Let $\mathbb{D}$ be the unit disk, then*

$$\min_{1\leq i,j,k\leq n} d(v_i, v_j, v_k) \leq 2\sin^2\frac{\pi}{n} \tag{4.34}$$

for all $v_1, v_2, \ldots\ldots, v_n \in \mathbb{D}$ if all the points are n the boundary of the convex hull of the point set $\{v_1, v_2, \ldots, v_n\}$. Therefore

$$\min_{1\le i,j,k\le n} d(v_i, v_j, v_k) \le \frac{2\pi^2}{n^2} \tag{4.35}$$

for all $v_1, v_2, \ldots\ldots, v_n \in \mathbb{D}$ if all the points are on the boundary of the convex hull of the point set $\{v_1, v_2, \ldots, v_n\}$.

Proof. If all the points are on the boundary of the convex hull of the point set $\{v_1, v_2, \ldots, v_n\}$, we can move the points $\{v_1, v_2, \ldots, v_n\}$ towards the boundary and have a convex n-gon whose vertices $\{v_1', v_2', \ldots, v_n'\}$are on $\partial\mathbb{D}$ whose perimeter is no less than that of the n-gon $\{v_1, v_2, \ldots, v_n\}$, because suppose that a vertex $v_{i_0}$is not on the boundary $\partial\mathbb{D}$, then the level set

$$\{v \in \mathbb{D} : \|\overrightarrow{vv_{i_0-1}}\| + \|\overrightarrow{vv_{i_0+1}}\| = \|\overrightarrow{v_{i_0}v_{i_0-1}}\| + \|\overrightarrow{v_{i_0}v_{i_0+1}}\|\}, \tag{4.36}$$

where v_{i_0-1} and v_{i_0+1} (assuming $v_{n+1} = v_1$) are the adjacent vertices of v_{i_0}, is a ellipse. Connect the center of the disk $\mathbb{D}$ and v_{i_0} by a ray and extend the ray till it intersects the boundary $\partial\mathbb{D}$ at v_{i_0}', then

$$\left\|\overrightarrow{v_{i_0}'v_{i_0-1}}\right\| + \left\|\overrightarrow{v_{i_0}'v_{i_0+1}}\right\| \ge \|\overrightarrow{v_{i_0}v_{i_0-1}}\| + \|\overrightarrow{v_{i_0}v_{i_0+1}}\|. \tag{4.37}$$

Thus

$$\text{perimeter}\left(\overline{v_1'v_2'\ldots v_n'}\right) \ge \text{perimeter}\left(\overline{v_1v_2\ldots v_n}\right). \tag{4.38}$$

Let θ_i be the central angle of the chord $v_i'v_{i+1}'$, assuming $v_{n+1}' = v_1'$. Then

$$\text{perimeter}\left(\overline{v_1'v_2'\ldots v_n'}\right) = 2\sum_{i=1}^{n}\sin\frac{\theta_i}{2} \le 2n\sin\left(\frac{\sum_{i=1}^{n}\theta_i}{2n}\right) = 2n\sin\frac{\pi}{n} \tag{4.39}$$

by the concavity of the sine function. Combining (4.38) and (4.39), we have

$$\text{perimeter}\left(\overline{v_1v_2\ldots v_n}\right) \le 2n\sin\frac{\pi}{n}. \tag{4.40}$$

Let's denote the angle between $\overrightarrow{v_iv_{i+1}}$ and $\overrightarrow{v_iv_{i+2}}$ by α_i and the angle between $\overrightarrow{v_{i+2}v_{i+1}}$ and $\overrightarrow{v_{i+2}v_i}$ by β_i, assuming $v_{n+1} = v_1$ and $v_{n+2} = v_2$, then

$$\sum_{i=1}^{n}\alpha_i + \sum_{i=1}^{n}\beta_i = 2\pi \tag{4.41}$$

and furthermore, we have

$$\sum_{i=1}^{n}\sin\alpha_i + \sum_{i=1}^{n}\sin\beta_i \le 2n\sin\left(\frac{\sum_{i=1}^{n}\alpha_i + \sum_{i=1}^{n}\beta_i}{2n}\right) = 2n\sin\frac{\pi}{n} \tag{4.42}$$

again by the concavity of the sine function. Let $s_i := \|\overrightarrow{v_iv_{i+1}}\|$, $x_i := \sin\alpha_i$ and $y_i := \sin\beta_i$ for $i = 1, \ldots, n$, then

$$\sum_{i=1}^{n}x_i + \sum_{i=1}^{n}y_i \le 2n\sin\frac{\pi}{n} \tag{4.43}$$

and

$$\sum_{i=1}^{n}s_i \le 2n\sin\frac{\pi}{n}. \tag{4.44}$$

Define

$$F := \sum_{i=1}^{n} s_i (x_i + y_i) - \lambda \left(\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i - c_1 \right) - \mu \left(\sum_{i=1}^{n} s_i - c_2 \right), \tag{4.45}$$

where $0 \leq c_1 \leq 2n \sin \frac{\pi}{n}$ and $0 \leq c_2 \leq 2n \sin \frac{\pi}{n}$. Solving the system of equations,

$$\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i = c_1, \tag{4.46}$$

and

$$\sum_{i=1}^{n} s_i = c_2, \tag{4.47}$$

and

$$\partial_{x_i} F = 0, \tag{4.48}$$

that is $\lambda = s_i$, and

$$\partial_{s_i} F = 0, \tag{4.49}$$

that is

$$\mu = x_i + y_i, \tag{4.50}$$

we get $s_i = \frac{c_2}{n}$ and

$$x_i + y_i = \frac{c_1}{n} \tag{4.51}$$

for $i = 1, \ldots, n$. By the method of Lagrange multipliers with multiple constraints (see for instance, [19] and [12]),

$$\sum_{i=1}^{n} s_i (x_i + y_i) \leq \frac{c_1 c_2}{n} \leq 4n \sin^2 \frac{\pi}{n}, \tag{4.52}$$

which implies

$$2n \min \left(\min_{1 \leq i \leq n} s_i x_i, \min_{1 \leq i \leq n} s_i y_i \right) \leq 4n \sin^2 \frac{\pi}{n}. \tag{4.53}$$

Thus, there exists an i_0, $1 \leq i_0 \leq n$, such that either

$$s_{i_0} x_{i_0} \leq 2 \sin^2 \frac{\pi}{n} \tag{4.54}$$

or

$$s_{i_0} y_{i_0} \leq 2 \sin^2 \frac{\pi}{n}, \tag{4.55}$$

in other words, either

$$\|\overrightarrow{v_{i_0} v_{i_0+1}}\| \sin \alpha_{i_0} \leq 2 \sin^2 \frac{\pi}{n} \tag{4.56}$$

or

$$\|\overrightarrow{v_{i_0} v_{i_0+1}}\| \sin \beta_{i_0} \leq 2 \sin^2 \frac{\pi}{n}, \tag{4.57}$$

which implies (4.27) as desired. □

Now, let us consider the complementary probability that any point does not fall into the stripes around the lines connecting the preceding points. To obtain the conditional probability each time when a point is dropped into the disk, one needs to have a lower bound of the covering area. This approach calculates the the covering area of the stripes which have overlaps, but to find the covering area, it would depend on the configurations. For example, when the fourth point is dropped into the disk, there would be a difference on the next conditional probability whether the point is dropped into the interior of the region formed by the three preceding points or the exterior of the region. More precisely, if there are 4 random points, then there will be 7 overlaps (including one of them overlapped by three stripes) among the stripes if three points form a triangle whose interior contains the other point, whereas there will be only 4 overlaps among the stripes if 4 points form a quadrilateral. So the covering area depends on the configuration of the points in the unit square or unit disk.

Furthermore, one would need to have a significantly small probability estimate on the minimal distance greater than $\frac{C}{n^2}$ or more strongly $\frac{C}{n^3}$ in order to show that the probability that the minimal distance is less than $\frac{C}{n^2}$ is significantly high. Thus, if one uses the probability approach, the covering area of the stripes may be estimated. But the obstruction caused by configurations or convexity is still the main hard part to solve the problem completely by soft analysis or by quasi-exact hard analysis.

Let's look into the subdivisions of the unit square now. Let S be a set of n points in the unit square. Let q_n be the maximum of the minimal distance from any point of S to the line joining any other two points of S, in which the maximum is taken over all configurations of n points in the unit square, and $p_n = nq_n$. Suppose S_0 is the configuration that achieves the maximum, and divide the unit square into 4^k sub-regions of equal area and equal shape, by using the midpoints of the edges, with a suitable arrangement of the boundaries so that every point belongs to only one sub-square. We have the following lemma regarding the behavior of p_n.

Lemma 4.8. *Suppose that a sub-region contains no more than $\frac{n}{4^k+l}$ points of S_0 for some $l > 0$. Then there exists an n_1, such that $\frac{(4^k+l-1)n}{(4^k-1)(4^k+l)} < n_1 < n$ and $p_n \leq \frac{(4^k-1)(4^k+l)}{2^k(4^k+l-1)} p_{n_1}$.*

Proof. By pigeonhole principle, there exists an sub-region Q that contains at least $\left\lfloor \frac{(4^k+l-1)n}{(4^k-1)(4^k+l)} \right\rfloor$ points of S_0. Let n_1 be the number of points of S_0 that falls into Q. Then

$$q_n \leq \min_{v_i,v_j,v_k \in Q} d(v_i, v_j, v_k) \leq \frac{1}{2^k} q_{n_1} = \frac{1}{2^k n_1} p_{n_1} \leq \frac{\left(4^k-1\right)\left(4^k+l\right)}{2^k\left(4^k+l-1\right)n} p_{n_1}. \tag{4.58}$$

Thus it follows that

$$p_n \leq \frac{\left(4^k-1\right)\left(4^k+l\right)}{2^k\left(4^k+l-1\right)} p_{n_1}. \tag{4.59}$$

□

Let us continue considering the subdivisions of the unit square.

Lemma 4.9. *Let $v_1, \ldots, v_n$ be a set of n points in the unit square on the plane, and connect all pairs of points by line segments. Given any ε, $0 < \varepsilon < 1$, there exist more than $\left\lfloor \frac{n}{2} - \frac{4}{\varepsilon^2} \right\rfloor$ distinct line segments whose length is less than ε.*

Proof. Let us divide the unit square into 4^k sub-squares of equal area, by using the midpoints of the edges, with a suitable arrangement of the boundaries so that every point belongs to only one sub square and connect every pair of points in the same sub square by line segment.

For any given ε, $0 < \varepsilon < 1$, there exists an k such that

$$\frac{\sqrt{2}}{\varepsilon} < 2^k < \frac{2\sqrt{2}}{\varepsilon}. \tag{4.60}$$

Let n_i be the number of points in the i-th sub-square, $i = 1, \ldots, 4^k$, then $n = \sum_{i=1}^{4^k} n_i$, and the total number of the line segments in the sub-squares is

$$\sum_{i=1}^{4^k} \frac{n_i\,(n_i - 1)}{2} \geq \frac{1}{2} \sum_{i}^{4^k} (n_i - 1) = \frac{n - 4^k}{2}, \tag{4.61}$$

since $\frac{n_i(n_i-1)}{2} = 0$ if $n_i = 0$ or 1. Furthermore, by (4.60),

$$\frac{n - 4^k}{2} > \frac{n}{2} - \frac{4}{\varepsilon^2}. \tag{4.62}$$

Thus, the total number of line segments in the sub-squares is greater than $\left\lfloor \frac{n}{2} - \frac{4}{\varepsilon^2} \right\rfloor$, and the length of each line segment is less than ε, since the length of each side of the sub-squares is $\frac{\sqrt{2}}{2^k}$ that is less than ε by (4.60). □

On the angles, one has the following lemma.

Lemma 4.10. *For any $\alpha > 0$, among the angles between the $\left\lfloor \frac{n}{2} - \frac{4}{\varepsilon^2} \right\rfloor$ distinct lines, there exist at least $\left\lfloor \frac{\alpha\left(n\varepsilon^2 - 8\right)}{2\varepsilon^2(\alpha+\pi)} \right\rfloor$ angles less than α.*

Proof. Take any point in the plane as the vertex of the angle π and divide the angle into $\left\lfloor \frac{\pi}{\alpha} + 1 \right\rfloor$ smaller angles of equal degree. We can do parallel transports on the lines so that they pass through the vertex of the angle π. Then by the pigeonhole principle, there must be $\left\lfloor \frac{\alpha\left(n\varepsilon^2 - 8\right)}{2\varepsilon^2(\alpha+\pi)} \right\rfloor$ lines falling into the same angle, which is less than α. □

Considering the edge and angle, one has

Lemma 4.11. *If the smallest angle and edge are adjacent, then*

$$\min_{1 \leq i,j,k \leq n} d(v_i, v_j, v_k) \leq \frac{C}{n \log n} \tag{4.63}$$

for a constant $C > 0$.

Proof. Choose $\varepsilon = \frac{1}{\log n}$ and $\alpha = \frac{8}{n}$, then

$$\left\lfloor \frac{n}{2} - \frac{4}{\varepsilon^2} \right\rfloor \geq 1 \tag{4.64}$$

for $n \geq 6$ and

$$\left\lfloor \frac{\alpha\left(n\varepsilon^2 - 8\right)}{2\varepsilon^2\left(\alpha + \pi\right)} \right\rfloor \geq 1 \tag{4.65}$$

for $n \geq 15$. Therefore,

$$\min_{1 \leq i,j,k \leq n} d(v_i, v_j, v_k) \leq \frac{1}{\log n} \sin \frac{8}{n} \leq \frac{8}{n \log n} \tag{4.66}$$

for $n \geq 15$, and then (4.63) follows. □

We used quasi-exact hard analysis to obtain the decay rate. However, the tools or techniques in hard analysis may be used to obtain the same order of decay but probably better constant in the decay rate. From the perspective of hard analysis, based on numerical experiment results, we formulate the following conjecture for a slower decay rate.

Conjecture 4.12. *Let P_1, ..., P_n be a set of points in the unit disk on the plane and $d(i,j,k)$ be the distance of the point P_i to the line connecting other two points P_j and P_k, $1 \leq i,j,k \leq n$, then*

$$\min_{1 \leq i,j,k \leq n} d(i,j,k) \leq \frac{C}{n^{1+\varepsilon_0}} \tag{4.67}$$

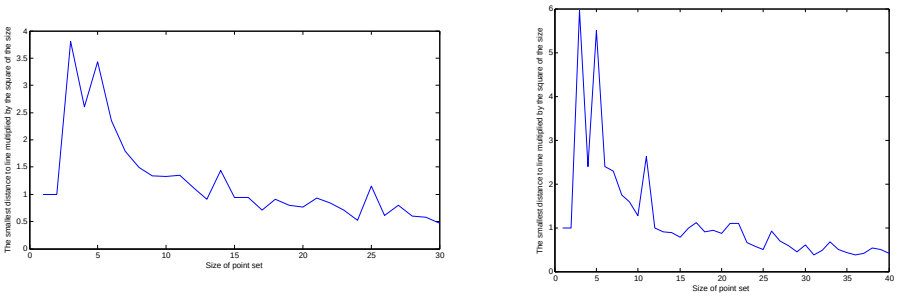
for some absolute constant C, $C > 0$, independent of n and some $\varepsilon_0 > 0$.

5. Numerical Experiments

In this section, we would like to present some numerical experimental results.

In the first and second numerical experiments, we use MATLAB to randomly generate n points in a unit square $[0,1]^2$ whose two coordinates are independent and identically distributed copies of uniformly distributed random variables and then compute the minimal the distance of a point to the line connecting other two points. For each matrix size n, we repeat this procedure n^2times to include n^2 sets of points of size n, and then take the maximum of the minimal distance over the n^2 repeats of randomly generating n points, due to the configurations increase greatly as the size of the point increases. After that, we multiply the maximum of the minimal distance by n^2 to compare the decay rate with $\frac{1}{n^2}$. From the figure Figure 5.1a on page 18 and Figure 5.1b on page 18, we can see that $n^2 \min_{1 \leq i,j,k \leq n} d(i,j,k)$ is bounded, as n increases, so $\min_{1 \leq i,j,k \leq n} d(i,j,k)$ decays mostly at at the order of at least $O\left(\frac{1}{n^2}\right)$ if the points are generated by normal random variables.

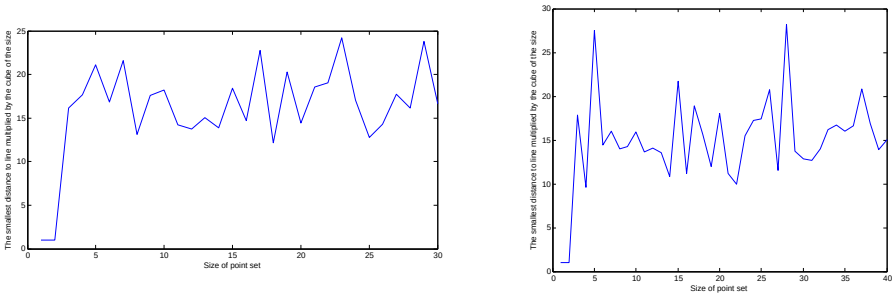
In the third and fourth numerical experiments, we use MATLAB to randomly generate n points in a unit square $[0,1]^2$ whose two coordinates are independent and identically distributed copies uniformly distributed random variables and then compute the minimal the distance of a point to the line connecting other two points. For each matrix size n, we repeat this procedure n^2 times to include n^2 sets of points of size n, and then take the maximum of the minimal distance over the n^2 repeats of randomly generating n points, due to the configurations increase greatly as the size of the point increases. After that, we multiply the maximum of the minimal distance by n^3 to compare the decay rate with $\frac{1}{n^3}$. From the figure Figure 5.2a on page 19 and Figure 5.2b on page 19, we can see that $n^3 \min_{1 \leq i,j,k \leq n} d(i,j,k)$ is bounded, as n increases, so $\min_{1 \leq i,j,k \leq n} d(i,j,k)$ decays with high probability at at the order of $O\left(\frac{1}{n^3}\right)$ mostly if the points are generated by normal random variables.



(A) Size of the point sets up to 30 (B) Size of the point sets up to 40

FIGURE 5.1. Plotted above are the smallest distances to lines multiplied by the square of the sizes of point sets, in which the two coordinates of points are independent and identically distributed copies of uniformly distributed random variables

In the fifth numerical experiment, we use MATLAB to randomly generate n points in a unit square $[0,1]^2$ whose two coordinates are independent and identically distributed copies uniformly distributed random variables and then compute the minimal the distance of a point to the line connecting other two points. For each matrix size n, we repeat this procedure 80 times to include n^2 sets of points of size n, and then take the maximum of the minimal distance over the 80 repeats of randomly generating n points, due to the configurations increase greatly as the size of the point increases. After that, we multiply the maximum of the minimal distance by n^3 to compare the decay rate with $\frac{1}{n^3}$. From Figure 5.3a on page 20, we can see that $n^3 \min_{1\leq i,j,k\leq n} d(i,j,k)$ is bounded, as n increases, so $\min_{1\leq i,j,k\leq n} d(i,j,k)$ decays with high probability at at the order of $O\left(\frac{1}{n^3}\right)$ mostly if the points are generated by normal random variables. In the sixth numerical experiment, we use MATLAB to randomly generate n points in a unit square $[0,1]^2$ whose two coordinates are independent and identically distributed copies uniformly distributed random variables and then compute the minimal the distance of a point to the line connecting other two points. For each matrix size n, we repeat this procedure 100 times to include 100 sets of points of size n, and then take the maximum of the minimal distance over the n^2 repeats of randomly generating n points, due to the configurations increase greatly as the size of the point increases. After that, we multiply the maximum of the minimal distance by n^3 to compare the decay rate



(A) Size of the point sets up to 30 (B) Size of the point sets up to 40

FIGURE 5.2. Plotted above are the smallest distances to lines multiplied by the square of the sizes of point sets, in which the two coordinates of points are independent and identically distributed copies uniformly distributed random variables.

with $\frac{1}{n^3}$. From Figure 5.3b on page 20, we can see that $n^3 \min_{1\leq i,j,k\leq n} d(i,j,k)$ is bounded, as n increases, so $\min_{1\leq i,j,k\leq n} d(i,j,k)$ decays with high probability at at the order of $O\left(\frac{1}{n^3}\right)$ mostly if the points are generated by normal random variables.

ACKNOWLEDGMENT

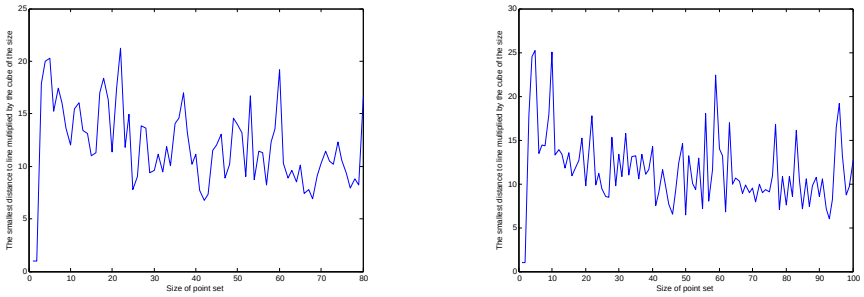
The authors would like to thank Prof. J. Schenker for some helpful discussion on the case of n by $n+2$ in 2.4. The authors would also like to thank the High Performance Computer Center (HPCC) at Michigan State University for the supercomputing service and Dr. Benjamin Ong for some technical support, which has helped us perform some numerical experiments in this research. Y. Liu is partially supported by the Air Force Office of Scientific Research under grant AFOSR 9550-12-1-0455.

REFERENCES

[1] Semyon Alesker. Continuous rotation invariant valuations on convex sets. *Annals of Mathematics*, 149:977–1005, 1999.

[2] Lorne Applebaum, Stephen D Howard, Stephen Searle, and Robert Calderbank. Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery. *Applied and Computational Harmonic Analysis*, 26(2):283–290, 2009.

[3] ZD Bai and YQ Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *The annals of Probability*, pages 1275–1294, 1993.



(A) Size of the point sets up to 80 (B) Size of the point sets up to 100

FIGURE 5.3. Plotted above are the smallest distances to lines multiplied by the square of the sizes of point sets, in which the two coordinates of points are independent and identically distributed copies uniformly distributed random variables.

[4] Gill Barequet and Alina Shaikhet. Heilbronn's triangle problem. In *Proceedings of the twenty-third annual symposium on Computational geometry*, pages 127–128. ACM, 2007.
[5] Emmanuel J Candes, Yonina C Eldar, Deanna Needell, and Paige Randall. Compressed sensing with coherent and redundant dictionaries. *Applied and Computational Harmonic Analysis*, 31(1):59–73, 2011.
[6] H. Coxeter. Introduction to geometry. 1969.
[7] P. ERD& and G. Szekeres. On some extremum problems in elementary geometry. In *Annales Univ. Sci. Budapest*, pages 3–4, 1960.
[8] P. Erdös and G. Szekeres. A combinatorial problem in geometry. *Compositio Mathematica*, 2:463–470, 1935.
[9] Gene H Golub and Christian Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420, 1970.
[10] Paul Goodey and Ralph Howard. Processes of flats induced by higher dimensional processes. *Advances in Mathematics*, 80(1):92–109, 1990.
[11] B. Green and T. Tao. On sets defining few ordinary lines. *arXiv preprint arXiv:1208.4714*, 2012.
[12] Jean-Baptiste Hiriart-Urruty and Claude Lemarechal. Abstract duality for practitioners. In *Convex Analysis and Minimization Algorithms II*, volume 306 of *Grundlehren der mathematischen Wissenschaften*, pages 137–193. Springer Berlin Heidelberg, 1993.
[13] JG Kalbfleisch and RG Stanton. On the maximum number of coplanar points containing no convex n-gons. *UTILITAS MATHEMATICA*, pages 235–235, 1995.
[14] L.M. Kelly. A resolution of the sylvester-gallai problem of j.-p. serre. *Discrete & Computational Geometry*, 1(1):101–104, 1986.

[15] D.A. Klain and G.C. Rota. *Introduction to geometric probability*. Cambridge University Press, 1997.
[16] Janos Komlos, Janos Pintz, and Endre Szemeredi. On heilbronns triangle problem. *Journal of the London Mathematical Society*, 24(2):385–396, 1981.
[17] M.J. Lai and Y. Liu. The null space property for sparse recovery from multiple measurement vectors. *Applied and Computational Harmonic Analysis*, 30(3):402–406, 2011.
[18] M.J. Lai and Y. Liu. The probabilistic estimates on the largest and smallest q-singular values of random matrices. *Mathematics of Computation*, 84(294):1775–1794, 2015.
[19] Claude Lemaréchal. Lagrangian relaxation. In *Computational Combinatorial Optimization*, pages 112–156. Springer, 2001.
[20] Alexander E Litvak, Alain Pajor, Mark Rudelson, and Nicole Tomczak-Jaegermann. Smallest singular value of random matrices and geometry of random polytopes. *Advances in Mathematics*, 195(2):491–523, 2005.
[21] Yang Liu. On the range of cosine transform of distributions for torus-invariant complex minkowski spaces. *Far East Journal of Mathematical Sciences*, 39(2):137–157, 2010.
[22] Yang Liu. On explicit holmes-thompson area formula in integral geometry. *accepted for publication*, 2011.
[23] Yang Liu. The probabilistic estimates of the largest strictly convex p-singular value of pregaussian random matrices. *Journal of Mathematics and Statistics*, DOI: 10.3844/jmssp.2015, 2015.
[24] Stéphane Mallat. *A wavelet tour of signal processing*. Academic press, 1999.
[25] W. Morris and V. Soltan. The erdos-szekeres problem on points in convex position-a survey. *Bulletin of the American Mathematical Society*, 37(4):437–458, 2000.
[26] Henrik Ohlsson, Allen Yang, Roy Dong, and Shankar Sastry. Cprl–an extension of compressive sensing to the phase retrieval problem. In *Advances in Neural Information Processing Systems*, pages 1367–1375, 2012.
[27] Robert Qiu and Michael Wicks. *Cognitive Networked Sensing and Big Data*. Springer, 2013.
[28] Mark Rudelson and Roman Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics*, 62(12):1707–1739, 2009.
[29] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. In *Proceedings of the International Congress of Mathematicians*, 2010.
[30] L.A. Santaló. *Integral geometry and geometric probability*. Cambridge University Press, 2004.
[31] Rolf Schneider and Wolfgang Weil. *Stochastic and integral geometry*. Springer Science & Business Media, 2008.

Department of Mathematics, Michigan State University, East Lansing, MI 48824-1027
E-mail address: yliu@math.msu.edu

Department of Mathematics, Michigan State University, East Lansing, MI 48824-1027
E-mail address: ywang@math.msu.edu

A DISTRIBUTED AND INCREMENTAL SVD ALGORITHM FOR AGGLOMERATIVE DATA ANALYSIS ON LARGE NETWORKS

M. A. IWEN* AND B. W. ONG†

Abstract. In this paper it is shown that the SVD of a matrix can be constructed efficiently in a hierarchical approach. The proposed algorithm is proven to recover the singular values and left singular vectors of the input matrix A if its rank is known. Further, the hierarchical algorithm can be used to recover the d largest singular values and left singular vectors with bounded error. It is also shown that the proposed method is stable with respect to roundoff errors or corruption of the original matrix entries. Numerical experiments validate the proposed algorithms and parallel cost analysis.

Key words. Singular value decomposition; low-rank approximations; distributed computing; incremental SVD

AMS subject classifications. 15-A23, 65-F20

1. Introduction.

The singular value decomposition (SVD) of a matrix,

$$A = U\Sigma V^*, \tag{1}$$

has applications in many areas including principal component analysis [13], the solution to homogeneous linear equations, and low-rank matrix approximations. If A is a complex matrix of size $D \times N$, then the factor U is a unitary matrix of size $D \times D$ whose first nonzero entry in each column is a positive real number, [1] Σ is a rectangular matrix of size $D \times N$ with non-negative real numbers (known as singular values) ordered from largest to smallest down its diagonal, and V^* (the conjugate transpose of V) is also a unitary matrix of size $N \times N$. If the matrix A is of rank $d < \min(D, N)$, then a reduced SVD representation is possible:

$$A = \hat{U}\hat{\Sigma}\hat{V}^*, \tag{2}$$

where $\hat{\Sigma}$ is a $d \times d$ diagonal matrix with positive singular values, $\hat{U}$ is an $D \times d$ matrix with orthonormal columns, and $\hat{V}$ is a $d \times N$ matrix with orthonormal columns.

The SVD of A is typically computed in three stages: a bidiagonal reduction step, computation of the singular values, and then computation of the singular vectors. The bidiagonal reduction step is computationally intensive, and is often targeted for parallelization. A serial approach to the bidiagonal reduction is the Golub–Kahan bidiagonalization algorithm [9], which reduces the matrix A to an upper-bidiagonal matrix by applying a series of Householder reflections alternately, applied from the left and right. Low-level parallelism is possible by distributing matrix-vector multiplies, for example by using the cluster computing framework Spark [21]. Using this form of low-level parallelism for the SVD has been implemented in the Spark project

*Department of Mathematics and Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, (markiwen@math.msu.edu). M. A. Iwen was supported in part by NSF DMS-1416752. Computational resources were provided by the Institute for Cyber-Enabled Research @ MSU.

†Department of Mathematical Sciences, Michigan Technological University, Houghton, MI (ongbw@mtu.edu). B. W. Ong was supported in part by AFOSR FA9550-12-1-0455. Computational resources were provided by Superior, the high-performance computing cluster @ MTU.

[1] This last condition on U guarantees that the SVD of $A \in \mathbb{C}^{N\times N}$ will be unique whenever AA^* has no repeated eigenvalues.

MLlib [18], and Magma [1], which develops its own framework to leverage GPU accelerators and hybrid manycore systems. Alternatively, parallelization is possible on an algorithmic level; it is possible by applying independent reflections simultaneously, for example [15] maps the bidiagonalization algorithm to graphical processing (GPU) units, and [14] executes the bidiagonalization on a distributed cluster. Load balancing is an issue for such parallel algorithms however, because the number of off-diagonal columns (or rows) to eliminate get successively smaller. More recently, two-stage approaches have been proposed and utilized in high-performance implementations for the bidiagonal reduction [16, 10]. The first stage reduces the original matrix to a banded matrix, the second stage subsequently reduces the banded matrix to the desired upper-bidiagonal matrix. A heroic effort to optimize the algorithms to hide latency and cache misses was discussed and implemented [10]. Parallelization is also possible if one uses a probabilistic approach to approximating the SVD [11].

In this paper, we are concerned with finding the SVD of highly rectangular matrices, $N \gg D$. In many applications where such problems are posed, one typically cares about the singular values and the left singular vectors. For example, this work was motivated by the SVDs required in Geometric Multi-Resolution Analysis (GMRA) [2]; the higher-order singular value decomposition (HOSVD) [7] of a tensor requires the computation of n SVDs of very rectangular matrices, where n is the number of tensor modes. Similarly, tensor train factorization algorithms [19] for tensors require the computation of many very rectangular SVDs. In fact, the SVDs of distributed and highly rectangular matrices of data appear in many big-date era machine learning applications. To find the SVD of highly rectangular matrices, many methods have focused on randomized techniques; [17] provides a recent survey of these techniques.

Alternatively, one can take an incremental approach to computing the SVD of an input matrix. Such methods have the advantage that they can be used to help efficiently analyze datasets which (rapidly) evolve over time. Examples of such methods include [5], which computes the SVD of a matrix by adding one column at a time, or more generally, one can add blocks of a matrix at each time. In [4] a block-incremental approach for estimating the dominant singular values and vectors of a highly rectangular matrix is described. It is based on a QR factorization of blocks from the input matrix, which can be done efficiently in parallel. In fact, the QR decomposition can be computed using a communication-avoiding QR (CAQR) factorization [8], which utilizes a tree-reduction approach.

Our approach is similar in spirit to [8], but differs in that we utilize a block decomposition approach that utilizes a partial SVD rather than a full QR factorization. This is advantageous if the application only requires the singular values and/or left singular vectors as in tensor factorization [7, 19] and GMRA applications [2]. Another approach would be to compute the eigenvalue decomposition of the Gram matrix, AA^* [3]. Although computing the Gram matrix in parallel is straightforward using the block inner product, a downside to this approach is a loss of numerical precision, and the general availability of the entire matrix A, which one may not have easy access to (i.e., computation of the Gram matrix, AA^*, is not easily achieved in an incremental and distributed setting).

The remainder of the paper is laid out as follows: In Section 2, we motivate incremental approaches to constructing the SVD before introducing the hierarchical algorithm. Theoretical justifications are given to show that the algorithm exactly recovers the singular values and left singular vectors if the rank of the matrix A is known. An error analysis is also used to show that the hierarchical algorithm can be used to recover the d largest singular values and left singular vectors with bounded

error, and that the algorithm is stable with respect to roundoff errors or corruption of the original matrix entries. In Section 3, numerical experiments validate the proposed algorithms and parallel cost analysis.

2. An Incremental (hierarchical) SVD Approach. The overall idea behind the proposed approach is relatively simple. We require a *distributed* and *incremental* approach for computing the singular values and left singular vectors of all data stored across a large distributed network. This can be achieved by, e.g., performing an incremental partial SVD separately on each network node by occasionally combining each node's previously computed partial SVD representation of its past data with a new partial SVD of its more recent data. The result of this approach will be that each separate network node always contains a fairly accurate approximation of it's cumulative data over time. Of course, these separate partial SVDs must then be merged together in order to understand the network data as a whole. Toward this end, neighboring node's partial SVD approximations can be combined hierarchically in order to compute a global partial SVD of the data stored across the entire network.

Note that the accuracy of the entire approach will be determined by the accuracy of the (hierarchical) partial SVD merging technique, which is ultimately what leads to the proposed method being both incremental and distributed. Theoretical analysis of this partial SVD merging technique is the primary purpose of this section. In particular, we prove the proposed partial SVD merging scheme is both numerically robust to data and/or roundoff errors, and also accurate even when the rank of the overall data matrix A is underestimated and/or purposefully reduced.

2.1. Mathematical Preliminaries. Let $A \in \mathbb{C}^{D\times N}$ be a highly rectangular matrix, with $N \gg D$. Further, let $A^i \in \mathbb{C}^{D\times N_i}$ with $i = 1, 2, \ldots, M$, denote the block decomposition of A, i.e., $A = \left[A^1|A^2|\cdots|A^M\right]$.

DEFINITION 1. *For any matrix $A \in \mathbb{C}^{D\times N}$, $(A)_d \in \mathbb{C}^{D\times N}$ is an optimal rank d approximation to A with respect to Frobenius norm $\|\cdot\|_{\mathrm{F}}$ if*

$$\inf_{B\in\mathbb{C}^{D\times N}} \|B - A\|_F = \|(A)_d - A\|_F, \text{ subject to } \operatorname{rank}(B) \le d.$$

Further, if A has the SVD decomposition $A = U\Sigma V^*$, then $(A)_d = \sum_{i=1}^{d} u_i\sigma_i v_i^*$, where u_i and v_i are singular vectors that comprise U and V respectively, and σ_i are the singular values.

This first lemma proves that partial SVDs of blocks of our original data matrix, $A \in \mathbb{C}^{D\times N}$, can combined block-wise into a new reduced matrix B which has the same singular values and left singular vectors as the original A. This basic lemma can be considered as the simplest merging method for either constructing an incremental SVD approach (different blocks of A have their partial SVDs computed at different times, which are subsequently merged into B), a distributed SVD approach (different nodes of a network compute partial SVDs of different blocks of A separately, and then send them to a single master node for combination into B), or both.

LEMMA 2. *Suppose that $A \in \mathbb{C}^{D\times N}$ has rank $d \in \{1, \ldots, D\}$, and let $A^i \in \mathbb{C}^{D\times N_i}, i = 1, 2, \ldots, M$ be the block decomposition of A, i.e., $A = \left[A^1|A^2|\cdots|A^M\right]$. Since A^i has rank at most d, each block has a reduced SVD representation,*

$$A^i = \sum_{j=1}^{d} u_j^i \sigma_j^i (v_j^i)^* = \hat{U}^i \hat{\Sigma}^i \hat{V}^{i*}, \quad i = 1, 2, \ldots, M.$$

Let $B := \left[\hat{U}^1\hat{\Sigma}^1|\hat{U}^2\hat{\Sigma}^2|\cdots|\hat{U}^M\hat{\Sigma}^M\right]$. *If* A *has the reduced SVD decomposition,* $A = \hat{U}\hat{\Sigma}\hat{V}^*$, *and* B *has the reduced SVD decomposition,* $B = \hat{U}'\hat{\Sigma}'\hat{V}'^*$, *then* $\hat{\Sigma} = \hat{\Sigma}'$, *and* $\hat{U} = \hat{U}'W$, *where* W *is a unitary block diagonal matrix satisfying* $\hat{U} = \hat{U}'W$. *If none of the nonzero singular values are repeated then* $\hat{U} = \hat{U}'$ *(i.e.,* W *is the identity when all the nonzero singular values of* A *are unique).*

Proof. The singular values of A are the (non-negative) square root of the eigenvalues of AA^*. Using the block definition of A,

$$AA^* = \sum_{i=1}^{M} A^i(A^i)^* = \sum_{i=1}^{M} \hat{U}^i\hat{\Sigma}^i(\hat{V}^i)^*(\hat{V}^i)(\hat{\Sigma}^i)^*(\hat{U}^i)^* = \sum_{i=1}^{M} \hat{U}^i\hat{\Sigma}^i(\hat{\Sigma}^i)^*(U^i)^*$$

Similarly, the singular values of B are the (non-negative) square root of the eigenvalues of BB^*.

$$BB^* = \sum_{i=1}^{M} (\hat{U}^i\hat{\Sigma}^i)(\hat{U}^i\hat{\Sigma}^i)^* = \sum_{i=1}^{M} \hat{U}^i\hat{\Sigma}^i(\hat{\Sigma}^i)^*(\hat{U}^i)^*$$

Since $AA^* = BB^*$, the singular values of B must be the same as the singular values of A. Similarly, the left singular vectors of both A and B will be eigenvectors of AA^* and BB^*, respectively. Since $AA^* = BB^*$ the eigenspaces associated with each (possibly repeated) eigenvalue will also be identical so that $\hat{U} = \hat{U}'W$. The block diagonal unitary matrix W (with one unitary $h \times h$ block for each eigenvalue that is repeated h-times) allows for singular vectors associated with repeated singular values to be rotated in the matrix representation $\hat{U}$. ∎

We now propose and analyze a more useful SVD approach which takes the ideas present in Lemma 2 to their logical conclusion.

2.2. An Incremental (Hierarchical) SVD Algorithm. The idea is to leverage the result in Lemma 2 by computing (in parallel) the SVD of the blocks of A, concatenating the scaled left singular vectors of the blocks to form a proxy matrix B, and then finally recovering the singular values and left singular vectors of the original matrix A by finding the SVD of the proxy matrix. A visualization of these steps are shown in Figure 1. Provided the proxy matrix is not very large, the computational and memory bottleneck of this algorithm is in the simultaneous SVD computation of the blocks A^i. If the proxy matrix is sufficiently large that the computational/memory overhead is significant, a multi-level hierarchical generalization is possible through repeated application of Lemma 2. Specifically, one could generate multiple proxy matrices by concatenating subsets of scaled left singular vectors obtained from the SVD of blocks of A, find the SVD of the proxy matrices and concatenate those singular vectors to form a new proxy matrix, and then finally recover the singular values and left singular vectors of the original matrix A by finding the SVD of the proxy matrix. A visualization of this generalization is shown in Figure 2 for a two-level parallel decomposition. A general q-level algorithm is described in Algorithm 1.

2.3. Theoretical Justification. In this section we will introduce some additional notation for the sake of convenience. For any matrix $A \in \mathbb{C}^{D\times N}$ with SVD $A = U\Sigma V^*$, we will let $\overline{A} := U\Sigma = AV \in \mathbb{C}^{D\times N}$. [2] Given this notation Lemma 2 can be rephrased as follows:

[2] It is important to note that $\overline{A}$ is not necessarily uniquely determined by A if, e.g., A is rank deficient and/or has repeated singular values. In these types of cases many pairs of unitary U and

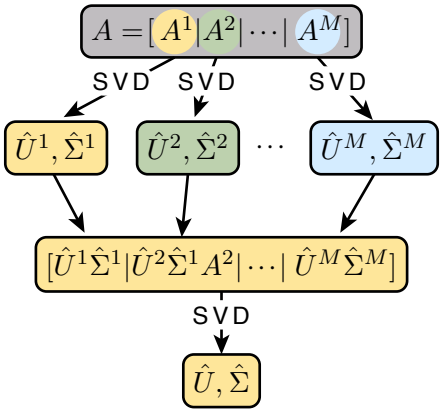


FIG. 1. *Flowchart for a simple (one-level) distributed parallel SVD algorithm. The different colors represent different processors completing operations in parallel.*

Algorithm 1 A q-level, distributed SVD Algorithm for Highly Rectangular $A \in \mathbb{C}^{D\times N}$, $N \gg D$.

Input: q (# levels),
n (# local SVDs to concatenate at each level),
$d \in \{1, \ldots, D\}$ (intrinsic dimension),
$A^{1,i} := A^i \in \mathbb{C}^{D\times N_i}$ for $i = 1, 2, \ldots, M$ (block decomposition of A; algorithm assumes $M = n^q$ – generalization is trivial)

Output: $U' \in \mathbb{C}^{D\times d} \approx$ the first d columns of U, and $\Sigma' \in \mathbb{R}^{d\times d} \approx (\Sigma)_d$.

1: **for** $p = 1, \ldots, q$ **do**
2: Compute (in parallel) the SVDs of $A^{p,i} = U^{p,i}\Sigma^{p,i}\left(V^{p,i}\right)^*$ for $i = 1, 2, \ldots, M/n^{(p-1)}$, unless the $U^{p,i}\Sigma^{p,i}$ are already available from a previous run.
3: Set $A^{p+1,i} := \left[\left(U^{p,(i-1)n+1}\Sigma^{p,(i-1)n+1}\right)_d \,\middle|\, \cdots \,\middle|\, \left(U^{p,in}\Sigma^{p,in}\right)_d\right]$ for $i = 1, 2, \ldots, M/n^p$.
4: **end for**
5: Compute the SVD of $A^{q+1,1}$
6: Set $U' :=$ the first d columns of $U^{q+1,1}$, and $\Sigma' := \left(\Sigma^{q+1,1}\right)_d$.

COROLLARY 1. *Suppose that $A \in \mathbb{C}^{D\times N}$ has rank $d \in \{1, \ldots, D\}$, and let $A^i \in \mathbb{C}^{D\times N_i}, i = 1, 2, \ldots, M$ be the block decomposition of A, i.e., $A = \left[A^1 | A^2 | \cdots | A^M\right]$. Since A^i has rank at most d for all $i = 1, 2, \ldots, M$, we have that*

$$\overline{(A)_d} = \overline{A} = \overline{\left[\overline{A^1} \mid \overline{A^2} \mid \cdots \mid \overline{A^M}\right]} = \overline{\left[\overline{(A^1)_d} \mid \overline{(A^2)_d} \mid \cdots \mid \overline{(A^M)_d}\right]}.$$

175

V may appear in a valid SVD of A. Below, one can consider $\overline{A}$ to be AV *for any such valid unitary matrix* V. Similarly, one can always consider statements of the form $\overline{A} = \overline{B}$ as meaning that A and B are equivalent up to multiplication by a unitary matrix on the right. This inherent ambiguity does not effect the results below in a meaningful way.

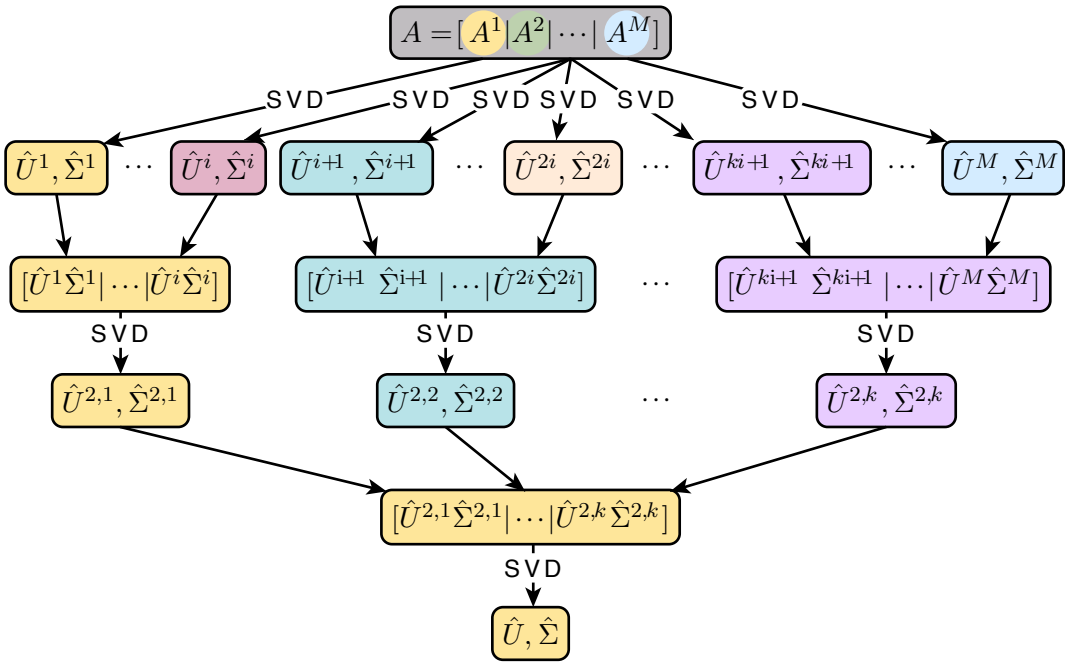


FIG. 2. *Flowchart for a two-level hierarchical parallel SVD algorithm. The different colors represent different processors completing operations in parallel.*

We can now prove that Algorithm 1 is guaranteed to recover $\overline{A}$ when the rank of A is known. The proof follows by inductively applying Corollary 1.

THEOREM 1. *Suppose that $A \in \mathbb{C}^{D\times N}$ has rank $d \in \{1,\ldots,D\}$. Then, Algorithm 1 is guaranteed to recover an $A^{q+1,1} \in \mathbb{C}^{D\times N}$ such that $\overline{A^{q+1,1}} = \overline{A}$.*

Proof. We prove the theorem by induction on the level p. To establish the base case we note that

$$\overline{A} \;=\; \overline{\left[\overline{(A^{1,1})_d} \;\middle|\; \overline{(A^{1,2})_d} \;\middle|\; \cdots \;\middle|\; \overline{(A^{1,M})_d}\right]} \;=\; \overline{\left[\overline{A^{1,1}} \;\middle|\; \overline{A^{1,2}} \;\middle|\; \cdots \;\middle|\; \overline{A^{1,M}}\right]}$$

holds by Corollary 1. Now, for the purpose of induction, suppose that

$$\overline{A} \;=\; \overline{\left[\overline{(A^{p,1})_d} \;\middle|\; \overline{(A^{p,2})_d} \;\middle|\; \cdots \;\middle|\; \overline{(A^{p,M/n^{(p-1)}})_d}\right]} \;=\; \overline{\left[\overline{A^{p,1}} \;\middle|\; \overline{A^{p,2}} \;\middle|\; \cdots \;\middle|\; \overline{A^{p,M/n^{(p-1)}}}\right]}$$

holds for some some $p \in \{1,\ldots,q\}$. Then, we can use the induction hypothesis and repartition the blocks of $\overline{A}$ to see that

$$\begin{aligned}
\overline{A} \;&=\; \overline{\left[\overline{(A^{p,1})_d} \;\middle|\; \overline{(A^{p,2})_d} \;\middle|\; \cdots \;\middle|\; \overline{(A^{p,M/n^{(p-1)}})_d}\right]} \\
&=\; \overline{\left[\cdots \;\middle|\; \left[(\overline{A^{p,(i-1)n+1}})_d \ldots (\overline{A^{p,in}})_d\right] \;\middle|\; \cdots\right]},\ i = 1,\ldots,M/n^p \\
&=\; \overline{\left[A^{p+1,1} \;\middle|\; A^{p+1,2} \;\middle|\; \cdots \;\middle|\; A^{p+1,M/n^p}\right]}, \qquad (3)
\end{aligned}$$

where we have utilized the definition of $A^{p+1,i}$ from line 3 of Algorithm 1 to get (3). Applying Corollary 1 to the matrix in (3) now yields

$$\overline{A} = \overline{\left[\overline{A^{p+1,1}} \mid \overline{A^{p+1,2}} \mid \cdots \mid \overline{A^{p+1,M/n^p}}\right]}.$$

Finally, we finish by noting that each $\overline{A^{p+1,i}}$ will have rank at most d since $\overline{A}$ is of rank d. Hence, we will also have $\overline{A} = \overline{\left[\overline{(A^{p+1,1})_d} \mid \overline{(A^{p+1,2})_d} \mid \cdots \mid \overline{(A^{p+1,M/n^p})_d}\right]}$, finishing the proof. ∎

Our next objective is to understand the accuracy of Algorithm 1 when it is called with a value of d that is less than rank of A. To begin we need a more general version of Lemma 2.

LEMMA 3. *Suppose $A^i \in \mathbb{C}^{D\times N_i}, i = 1, 2, \ldots, M$. Further, suppose matrix A has block components $A = \left[A^1|A^2|\cdots|A^M\right]$, and B has block components $B = \left[(A^1)_d|(A^2)_d|\cdots|(A^M)_d\right]$. Then, $\|(B)_d - A\|_{\mathrm{F}} \leq \|(B)_d - B\|_{\mathrm{F}} + \|B - A\|_{\mathrm{F}} \leq 3\|(A)_d - A\|_{\mathrm{F}}$ holds for all $d \in \{1, \ldots, D\}$.*

Proof. We have that

$$\begin{aligned}\|(B)_d - A\|_{\mathrm{F}} &\leq \|(B)_d - B\|_{\mathrm{F}} + \|B - A\|_{\mathrm{F}} \\ &\leq \|(A)_d - B\|_{\mathrm{F}} + \|B - A\|_{\mathrm{F}} \\ &\leq \|(A)_d - A\|_{\mathrm{F}} + 2\|B - A\|_{\mathrm{F}}.\end{aligned}$$

Now letting $(A)_d^i \in \mathbb{C}^{D\times N_i}, i = 1, 2, \ldots, M$ denote the i^{th} block of $(A)_d$, we can see that

$$\begin{aligned}\|B - A\|_{\mathrm{F}}^2 &= \sum_{i=1}^{M} \|(A^i)_d - A^i\|_{\mathrm{F}}^2 \\ &\leq \sum_{i=1}^{M} \|(A)_d^i - A^i\|_{\mathrm{F}}^2 \\ &= \|(A)_d - A\|_{\mathrm{F}}^2.\end{aligned}$$

Combining these two estimates now proves the desired result. ∎

We can now use Lemma 3 to prove a theorem that that will help us to bound the error produced by Algorithm 1 when d is chosen to be less than the rank of rank of A. It improves over Lemma 3 (in our setting) by not implicitly assuming to have access to any information regarding the right singular vectors of the blocks of A. It also demonstrates that the proposed method is stable with respect to additive errors by allowing (e.g., roundoff) errors, represented by Ψ, to corrupt the original matrix entries. Note that Theorem 2 is a strict generalization of Corollary 1. Corollary 1 is recovered from it when Ψ is chosen to be the zero matrix, and d is chosen to be the rank of A.

THEOREM 2. *Suppose that $A \in \mathbb{C}^{D\times N}$ has block components $A^i \in \mathbb{C}^{D\times N_i}, i = 1, 2, \ldots, M$, so that $A = \left[A^1|A^2|\cdots|A^M\right]$. Let $B = \left[\overline{(A^1)_d} \mid \overline{(A^2)_d} \mid \cdots \mid \overline{(A^M)_d}\right]$, $\Psi \in \mathbb{C}^{D\times N}$, and $B' = B + \Psi$. Then, there exists a unitary matrix W such that*

$$\left\|\overline{(B')_d} - AW\right\|_{\mathrm{F}} \leq 3\sqrt{2}\|(A)_d - A\|_{\mathrm{F}} + \left(1 + \sqrt{2}\right)\|\Psi\|_{\mathrm{F}}$$

holds for all $d \in \{1, \ldots, D\}$.

Proof. Let $A' = \left[\overline{A^1} \mid \overline{A^2} \mid \cdots \mid \overline{A^M}\right]$. Note that $\overline{A'} = \overline{A}$ by Corollary 1. Thus, there exists a unitary matrix W'' such that $A' = \overline{A}W''$. Using this fact in combination with the unitary invariance of the Frobenius norm, one can now see that

$$\left\|(B')_d - A'\right\|_{\mathrm{F}} \;=\; \left\|(B')_d - \overline{A}W''\right\|_{\mathrm{F}} \;=\; \left\|\overline{(B')_d} - \overline{A}W'\right\|_{\mathrm{F}} = \left\|\overline{(B')_d} - AW\right\|_{\mathrm{F}}$$

for some unitary matrixes W' and W. Hence, it suffices to bound $\left\|(B')_d - A'\right\|_{\mathrm{F}}$.

Proceeding with this goal in mind we can see that

$$\begin{aligned}
\left\|(B')_d - A'\right\|_{\mathrm{F}} &\leq \left\|(B')_d - B'\right\|_{\mathrm{F}} + \left\|B' - B\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}} \\
&= \sqrt{\sum_{j=d+1}^{D} \sigma_j^2(B+\Psi)} \;+\; \left\|\Psi\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}} \\
&= \sqrt{\sum_{j=1}^{\lceil \frac{D-d}{2} \rceil} \sigma_{d+2j-1}^2(B+\Psi) + \sigma_{d+2j}^2(B+\Psi)} \;+\; \left\|\Psi\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}} \\
&\leq \sqrt{\sum_{j=1}^{\lceil \frac{D-d}{2} \rceil} \left(\sigma_{d+j}(B) + \sigma_j(\Psi)\right)^2 + \left(\sigma_{d+j}(B) + \sigma_{j+1}(\Psi)\right)^2} \;+\; \left\|\Psi\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}}
\end{aligned}$$

where the last inequality results from an application of Weyl's inequality to the first term (see, e.g., Theorem 3.3.16 in [12]). Utilizing the triangle inequality on the first term now implies that

$$\begin{aligned}
\left\|(B')_d - A'\right\|_{\mathrm{F}} &\leq \sqrt{\sum_{j=d+1}^{D} 2\sigma_j^2(B)} \;+\; \sqrt{\sum_{j=1}^{D} 2\sigma_j^2(\Psi)} + \left\|\Psi\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}} \\
&\leq \sqrt{2}\left(\left\|(B)_d - B\right\|_{\mathrm{F}} + \left\|B - A'\right\|_{\mathrm{F}}\right) + \left(1+\sqrt{2}\right)\left\|\Psi\right\|_{\mathrm{F}}.
\end{aligned}$$

Applying Lemma 3 to bound the first two terms now concludes the proof after noting that $\left\|(A')_d - A'\right\|_{\mathrm{F}} = \left\|(A)_d - A\right\|_{\mathrm{F}}$. ∎

This final theorem bounds the total error of Algorithm 1 with respect to the true matrix A up to right multiplication by a unitary matrix. The structure of its proof is similar to that of Theorem 1.

THEOREM 3. *Let $A \in \mathbb{C}^{D\times N}$ and $q \geq 1$. Then, Algorithm 1 is guaranteed to recover an $A^{q+1,1} \in \mathbb{C}^{D\times N}$ such that $\overline{\left(A^{q+1,1}\right)_d} = AW + \Psi$, where W is a unitary matrix, and $\left\|\Psi\right\|_{\mathrm{F}} \leq \left(\left(1+\sqrt{2}\right)^{q+1} - 1\right)\left\|(A)_d - A\right\|_{\mathrm{F}}$.*

Proof. Within the confines of this proof we will always refer to the approximate matrix $A^{p+1,i}$ from line 3 of Algorithm 1 as

$$B^{p+1,i} := \left[\overline{\left(B^{p,(i-1)n+1}\right)_d} \;\middle|\; \cdots \;\middle|\; \overline{\left(B^{p,in}\right)_d}\right],$$

for $p = 1, \ldots, q$, and $i = 1, \ldots, M/n^p$. Conversely, A will always refer to the original (potentially full rank) matrix with block components $A = \left[A^1 | A^2 | \cdots | A^M\right]$, where $M = n^q$. Furthermore, $A^{p,i}$ will always refer to the error free block of the original

matrix A whose entries correspond to the entries included in $B^{p,i}$. [3] Thus, $A = \left[A^{p,1}|A^{p,2}|\cdots|A^{p,M/n^{(p-1)}}\right]$ holds for all $p = 1,\ldots,q+1$, where

$$A^{p+1,i} := \left[A^{p,(i-1)n+1} \;\middle|\cdots\middle|\; A^{p,in}\right]$$

for all $p = 1,\ldots,q$, and $i = 1,\ldots,M/n^p$. For $p = 1$ we have $B^{1,i} = A^i = A^{1,i}$ for $i = 1,\ldots,M$ by definition as per Algorithm 1. Our ultimate goal is to bound the renamed $\overline{(B^{q+1,1})_d}$ matrix from lines 5 and 6 of Algorithm 1 with respect to the original matrix A. We will do this by induction on the level p. More specifically, we will prove that

1. $\overline{(B^{p,i})_d} = A^{p,i}W^{p,i} + \Psi^{p,i}$, where
2. $W^{p,i}$ is always a unitary matrix, and
3. $\|\Psi^{p,i}\|_{\mathrm{F}} \leq \left(\left(1+\sqrt{2}\right)^p - 1\right)\left\|(A^{p,i})_d - A^{p,i}\right\|_{\mathrm{F}}$,

holds for all $p = 1,\ldots,q+1$, and $i = 1,\ldots,M/n^{(p-1)}$.

Note that conditions $1-3$ above are satisfied for $p = 1$ since $B^{1,i} = A^i = A^{1,i}$ for all $i = 1,\ldots,M$ by definition. Thus, there exist unitary $W^{1,i}$ for all $i = 1,\ldots,M$ such that

$$\overline{(B^{1,i})_d} \;=\; \overline{(A^{1,i})_d} \;=\; \left(A^{1,i}\right)_d W^{1,i} \;=\; A^{1,i}W^{1,i} + \left(\left(A^{1,i}\right)_d - A^{1,i}\right)W^{1,i},$$

where $\Psi^{1,i} := \left(\left(A^{1,i}\right)_d - A^{1,i}\right)W^{1,i}$ has $\|\Psi^{1,i}\|_{\mathrm{F}} = \left\|\left(A^{1,i}\right)_d - A^{1,i}\right\|_{\mathrm{F}} \leq \sqrt{2}\left\|\left(A^{1,i}\right)_d - A^{1,i}\right\|_{\mathrm{F}}$. ∎

Now suppose that conditions $1-3$ hold for some $p \in \{1,\ldots,q\}$. Then, one can see from condition 1 that

$$\begin{aligned}
B^{p+1,i} &:= \left[\overline{\left(B^{p,(i-1)n+1}\right)_d} \;\middle|\cdots\middle|\; \overline{(B^{p,in})_d}\right] \\
&= \left[A^{p,(i-1)n+1}W^{p,(i-1)n+1} + \Psi^{p,(i-1)n+1} \;\middle|\cdots\middle|\; A^{p,in}W^{p,in} + \Psi^{p,in}\right] \\
&= \left[A^{p,(i-1)n+1}W^{p,(i-1)n+1} \;\middle|\cdots\middle|\; A^{p,in}W^{p,in}\right] + \left[\Psi^{p,(i-1)n+1} \;\middle|\cdots\middle|\; \Psi^{p,in}\right] \\
&= \left[A^{p,(i-1)n+1} \;\middle|\cdots\middle|\; A^{p,in}\right]\tilde{W} + \tilde{\Psi},
\end{aligned}$$

where $\tilde{\Psi} := \left[\Psi^{p,(i-1)n+1} \;\middle|\cdots\middle|\; \Psi^{p,in}\right]$, and

$$\tilde{W} := \left(\begin{array}{c:c:c:c}
W^{p,(i-1)n+1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline
\mathbf{0} & W^{p,(i-1)n+2} & \mathbf{0} & \mathbf{0} \\ \hdashline
\mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} \\ \hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & W^{p,in}
\end{array}\right).$$

Note that $\tilde{W}$ is unitary since its diagonal blocks are all unitary by condition 2. Therefore, we have $B^{p+1,i} = A^{p+1,i}\tilde{W} + \tilde{\Psi}$.

We may now bound $\left\|\left(B^{p+1,i}\right)_d - A^{p+1,i}\tilde{W}\right\|_{\mathrm{F}}$ using a similar argument to that

[3] That is, $B^{p,i}$ is used to approximate the singular values and left singular vectors of $A^{p,i}$ for all $p = 1,\ldots,q+1$, and $i = 1,\ldots,M/n^{p-1}$

employed in the proof of Theorem 2.

$$\begin{aligned}
\left\|\left(B^{p+1,i}\right)_d - A^{p+1,i}\tilde{W}\right\|_{\mathrm{F}} &\leq \left\|\left(B^{p+1,i}\right)_d - B^{p+1,i}\right\|_{\mathrm{F}} + \left\|B^{p+1,i} - A^{p+1,i}\tilde{W}\right\|_{\mathrm{F}} \\
&= \sqrt{\sum_{j=d+1}^{D} \sigma_j^2\left(A^{p+1,i}\tilde{W} + \tilde{\Psi}\right)} \; + \; \|\tilde{\Psi}\|_{\mathrm{F}} \\
&\leq \sqrt{\sum_{j=d+1}^{D} 2\sigma_j^2\left(A^{p+1,i}\tilde{W}\right)} \; + \; \sqrt{\sum_{j=1}^{D} 2\sigma_j^2(\tilde{\Psi})} + \|\tilde{\Psi}\|_{\mathrm{F}} \\
&= \sqrt{2}\left\|A^{p+1,i} - \left(A^{p+1,i}\right)_d\right\|_{\mathrm{F}} + \left(1+\sqrt{2}\right)\|\tilde{\Psi}\|_{\mathrm{F}}. \qquad (4)
\end{aligned}$$

Appealing to condition 3 in order to bound $\|\tilde{\Psi}\|_{\mathrm{F}}$ we obtain

$$\begin{aligned}
\|\tilde{\Psi}\|_{\mathrm{F}}^2 = \sum_{j=1}^{n} \|\Psi^{p,(i-1)n+j}\|_{\mathrm{F}}^2 &\leq \left(\left(1+\sqrt{2}\right)^p - 1\right)^2 \sum_{j=1}^{n} \left\|(A^{p,(i-1)n+j})_d - A^{p,(i-1)n+j}\right\|_{\mathrm{F}}^2 \\
&\leq \left(\left(1+\sqrt{2}\right)^p - 1\right)^2 \sum_{j=1}^{n} \left\|(A^{p+1,i})_d^j - A^{p,(i-1)n+j}\right\|_{\mathrm{F}}^2,
\end{aligned}$$

where $(A^{p+1,i})_d^j$ denotes the block of $(A^{p+1,i})_d$ corresponding to $A^{p,(i-1)n+j}$ for $j = 1, \ldots, n$. Thus, we have that

$$\begin{aligned}
\|\tilde{\Psi}\|_{\mathrm{F}}^2 &\leq \left(\left(1+\sqrt{2}\right)^p - 1\right)^2 \sum_{j=1}^{n} \left\|(A^{p+1,i})_d^j - A^{p,(i-1)n+j}\right\|_{\mathrm{F}}^2 \\
&= \left(\left(1+\sqrt{2}\right)^p - 1\right)^2 \left\|(A^{p+1,i})_d - A^{p+1,i}\right\|_{\mathrm{F}}^2. \qquad (5)
\end{aligned}$$

Combining (4) and (5) we can finally see that

$$\begin{aligned}
\left\|\left(B^{p+1,i}\right)_d - A^{p+1,i}\tilde{W}\right\|_{\mathrm{F}} &\leq \left[\sqrt{2} + (1+\sqrt{2})\left(\left(1+\sqrt{2}\right)^p - 1\right)\right]\left\|\left(A^{p+1,i}\right)_d - A^{p+1,i}\right\|_{\mathrm{F}} \\
&= \left(\left(1+\sqrt{2}\right)^{p+1} - 1\right)\left\|\left(A^{p+1,i}\right)_d - A^{p+1,i}\right\|_{\mathrm{F}}. \qquad (6)
\end{aligned}$$

Note that $\left\|\left(B^{p+1,i}\right)_d - A^{p+1,i}\tilde{W}\right\|_{\mathrm{F}} = \left\|\overline{\left(B^{p+1,i}\right)_d} - A^{p+1,i}W^{p+1,i}\right\|_{\mathrm{F}}$ where $W^{p+1,i}$ is unitary. Hence, we can see that conditions 1 - 3 hold for $p+1$ with $\Psi^{p+1,i} := \overline{(B^{p+1,i})_d} - A^{p+1,i}W^{p+1,i}$. ∎

Having proven that the method is accurate for low rank A, we are now free to consider it's computational costs.

2.4. Parallel Cost Model and Collectives. To analyze the parallel communication cost of the hierarchical SVD algorithm, the $\alpha - \beta - \gamma$ model for distributed–memory parallel computation [6] is used. The parameters α and β respectively represent the latency cost and the transmission cost of sending a "word" between two processors. In our presentation, a word will refer to a vector of doubles in $\mathbb{R}^D$, i.e., a vector of size $D \times 1$. The parameter γ represents the time for one floating point operation (FLOP).

The q-level hierarchical Algorithm 1 seeks to find the d largest singular values and left singular vectors of a matrix A. If the matrix A is decomposed into $M = n^q$ blocks, where n being the number of local SVD's being concatenated at each level, the send/receive communication cost for the algorithm is is

$$q\left(\alpha + d\left(n-1\right)\beta\right),$$

assuming that the data is already distributed on the compute nodes and no scatter command is required. If the (distributed) right singular vectors are needed, then a broadcast of the left singular vectors to all nodes incurs a communication cost of $\alpha + d\,M\,\beta$.

Suppose A is a $D \times N$ matrix, $N \gg D$. The sequential SVD is typically performed in two phases: bidiagonalization (which requires $(2N\,D^2 + 2D^3)$ flops) followed by diagonalization (negligible cost). If M processing cores are available to compute the q-level hierarchical SVD method in Algorithm 1, and the matrix A is decomposed into $M = n^q$ blocks, where n is again the number of local SVD's being concatenated at each level. The potential parallel speedup can be approximated by

$$\frac{(2ND^2 + 2D^3)\gamma}{\gamma(2(N/M)D^2 + 2D^3) + q(2dnD^2 + 2D^3) + q(\alpha + d(n-1)\beta)}. \tag{7}$$

3. Numerical Validation. In the first experiment, the left singular vectors and the singular values of a matrix A ($D = d = 800$, $N = 1,152,000$) are found. We utilize a shared memory system which has 6 TB of memory and eight sockets, each equipped with a twelve-core Intel E7-8857v2 processor, for a total of 96 processing cores. Since the input data and memory storage required by the SVD algorithms fit in memory on this specialized compute node, we performed a strong scaling study of our one-level distributed SVD algorithm, benchmarked against the LAPACK SVD routine, `dgesvd`, implemented in the threaded Intel MKL library. In a pre-processing step, the matrix A is decomposed with each block of A stored in separate HDF5 files, hosted on a high-speed Lustre server capable of 6GB/s read/write i/o. The observed speedup is reported in Figure 3. In the blue curve, the observed speedup is reported for a varying number of MKL worker threads. In the red curve, the speedup is reported for a varying number of worker threads i, applied to an appropriate decomposition of the matrix. Each worker uses the same Intel MKL library to compute the SVD of the decomposed matrices (each using a single thread), the proxy matrix is assembled, and the master thread computes the SVD of the proxy matrix using the Intel MKL library, again with a single thread. Each numerical experiment is run four times, and the average walltime used to compute the observed speedup. The parallel performance of our distributed SVD is far superior, this in spite of the fact that our algorithm was implemented using MPI 2.0 and does not leverage the inter-node communication savings that is possible with newer MPI implementations. The dip in performance when more than 48 cores are used is likely attributed to non-uniform memory access on this large shared-memory node.

In the second experiment, we perform a weak scaling study, where the size of the input matrix A is varied depending on the number of worker nodes, $A = 2000 \times 32,000M$, where M is the number of compute cores. The experiment was conducted on a shared high-performance cluster (other users may be running computationally intensive processes on the same node, communication heavy processes on the network, or i/o heavy processes taxing the shared file systems), leading to some variability in the study. Each data point in Figure 4 is computed using the average walltime from

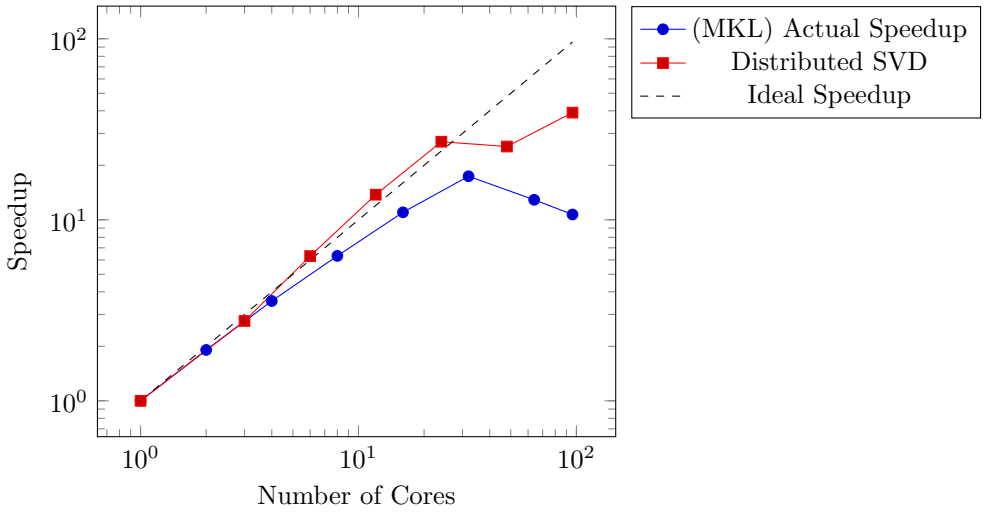


FIG. 3. *Strong scaling study of the* `dgesvd` *function in the threaded Intel MKL library (blue) and the proposed distributed SVD algorithm (red). The input matrix is of size* $800 \times 1,152,000$. *The slightly "better than ideal" speedup is likely due to better utilization of cache in each socket.*

five numerical experiments. Additionally, the network is constructed using a fat-tree topology that is oversubscribed by a ratio of 2:1, resulting in further variability based on the compute resources that were allocated for each numerical experiment. The theoretical peak efficiency is computed using equation 7, assuming negligible communication overhead.

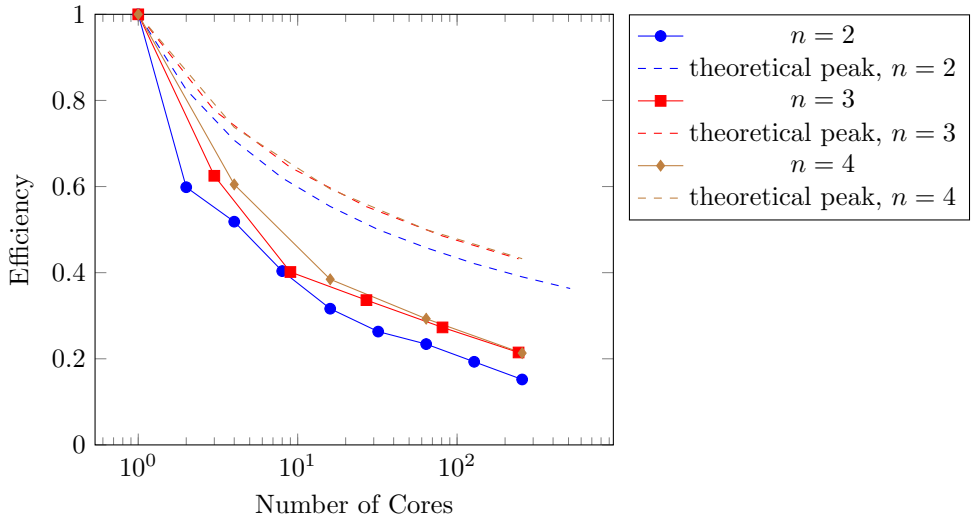


FIG. 4. *Weak scaling study of the hierarchical SVD algorithm. The input matrix is of size* $2000 \times (32000M)$, *where* M *is the processing cores used in the computation. The observed efficiency is plotted for various* n*'s (number of scaled singular vectors concatenated at each hierarchical level). There is a slight efficiency gain when increasing* n, *until the communication cost dominates, or the size of the proxy matrix becomes significantly large.*

In the last experiment, we repeat the weak scaling study (where the size of the input matrix A is varied depending on the number of worker nodes, $A = 2000 \times 32,000M$, where M is the number of compute cores, but utilize a priori knowledge that the rank of A is much less than the ambient dimension. Specifically, we construct a data set with $d = 100 \ll 2000$. The hierarchical SVD performs more efficiently if the intrinsic dimension of the data can be estimated a priori. There is a loss of efficiency when more than 64 cores are utilized. This is likely attributed to the network topology of the assigned computational resources.

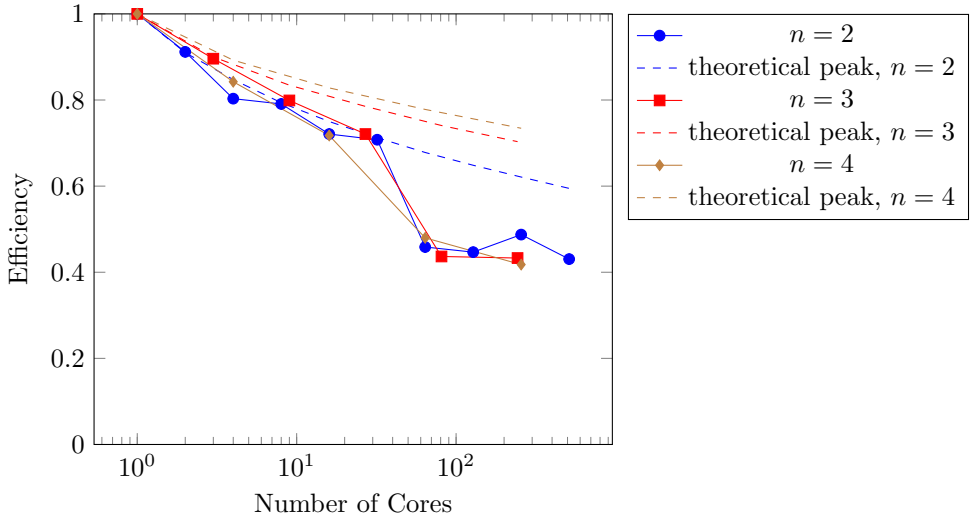


FIG. 5. *Weak scaling study of the hierarchical SVD algorithm applied to data with intrinsic dimension much lower than the ambient dimension. The input matrix is of size* $2000 \times (32000M)$*, where* M *is the processing cores used in the computation. The intrinsic dimension is* $d = 100 \ll 2000$*. The observed efficiency is plotted for various* n*'s (number of scaled singular vectors concatenated at each hierarchical level). As expected, the theoretical and observed efficiency are better if the intrinsic dimension is known (or can be estimated) a priori.*

4. Concluding Remarks and Acknowledgments. In this paper, we show that the SVD of a matrix can be constructed efficiently in a hierarchical approach. Our algorithm is proven to recover exactly the singular values and left singular vectors if the rank of the matrix A is known. Further, the hierarchical algorithm can be used to recover the d largest singular values and left singular vectors with bounded error. We also show that the proposed method is stable with respect to roundoff errors or corruption of the original matrix entries. Numerical experiments validate the proposed algorithms and parallel cost analysis.

Although not shown in the paper, the right singular vectors can be computed efficiently (in parallel) if desired, once the left singular vectors and singular values are known. The master process broadcasts the left singular vectors and singular values to each process. Then columns of the right singular vectors can be constructed by computing $\frac{1}{\sigma_j}(A^i)^* u_j$, were A^i is the block of A residing on process i, and (σ_j, u_j) is the j^{th} singular value and left singular vector respectively. The authors note that the practicality of the hierarchical algorithm is questionable for sparse input matrices, since the assembled proxy matrices as posed will be dense. Further investigation in this direction is required, but beyond the scope of this paper. Lastly, the hierarchical

algorithm has a map–reduce flavor that will lend itself well to a map reduce framework such as Apache Hadoop [20] or Apache Spark [22].

REFERENCES

[1] E. Agullo, J. Demmel, J. Dongarra, B. Hadri, J. Kurzak, J. Langou, H. Ltaief, P. Luszczek, and S. Tomov, *Numerical linear algebra on emerging architectures: The plasma and magma projects*, Journal of Physics: Conference Series, 180 (2009), p. 012037, http://stacks.iop.org/1742-6596/180/i=1/a=012037.

[2] W. K. Allard, G. Chen, and M. Maggioni, *Multi-scale geometric methods for data sets II: Geometric multi-resolution analysis*, Appl. Comput. Harmon. Anal., 32 (2012), pp. 435–462, doi:10.1016/j.acha.2011.08.001, http://dx.doi.org/10.1016/j.acha.2011.08.001.

[3] W. Austin, G. Ballard, and T. G. Kolda, *Parallel Tensor Compression for Large-Scale Scientific Data*, ArXiv e-prints, (2015), arXiv:1510.06689.

[4] C. G. Baker, K. A. Gallivan, and P. Van Dooren, *Low-rank incremental methods for computing dominant singular subspaces*, Linear Algebra Appl., 436 (2012), pp. 2866–2888, doi:10.1016/j.laa.2011.07.018, http://dx.doi.org/10.1016/j.laa.2011.07.018.

[5] J. R. Bunch and C. P. Nielsen, *Updating the singular value decomposition*, Numer. Math., 31 (1978/79), pp. 111–129, doi:10.1007/BF01397471, http://dx.doi.org/10.1007/BF01397471.

[6] E. Chan, M. Heimlich, A. Purkayastha, and R. van de Geijn, *Collective communication: theory, practice, and experience*, Concurrency and Computation: Practice and Experience, 19 (2007), pp. 1749–1783, doi:10.1002/cpe.1206, http://dx.doi.org/10.1002/cpe.1206.

[7] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278 (electronic), doi:10.1137/S0895479896305696, http://dx.doi.org/10.1137/S0895479896305696.

[8] J. Demmel, L. Grigori, M. Hoemmen, and J. Langou, *Communication-optimal parallel and sequential QR and LU factorizations*, SIAM J. Sci. Comput., 34 (2012), pp. A206–A239, doi:10.1137/080731992, http://dx.doi.org/10.1137/080731992.

[9] G. Golub and W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis, 2 (1965), pp. 205–224, doi:10.1137/0702016, http://dx.doi.org/10.1137/0702016, arXiv:http://dx.doi.org/10.1137/0702016.

[10] A. Haidar, J. Kurzak, and P. Luszczek, *An improved parallel singular value algorithm and its implementation for multicore hardware*, in Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC '13, New York, NY, USA, 2013, ACM, pp. 90:1–90:12, doi:10.1145/2503210.2503292, http://doi.acm.org/10.1145/2503210.2503292.

[11] N. Halko, P. G. Martinsson, and J. A. Tropp, *Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288, doi:10.1137/090771806, http://dx.doi.org/10.1137/090771806.

[12] R. A. Horn and C. R. Johnson, *Topics in matrix analysis*, Cambridge University Press, Cambridge, 1994. Corrected reprint of the 1991 original.

[13] I. T. Jolliffe, *Principal component analysis*, Springer Series in Statistics, Springer-Verlag, New York, second ed., 2002.

[14] F. Liu and F. Seinstra, *Adaptive parallel householder bidiagonalization*, in Euro-Par 2009 Parallel Processing, H. Sips, D. Epema, and H.-X. Lin, eds., vol. 5704 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2009, pp. 821–833, doi:10.1007/978-3-642-03869-3_76, http://dx.doi.org/10.1007/978-3-642-03869-3_76.

[15] F. Liu and F. J. Seinstra, *Gpu-based parallel householder bidiagonalization*, in Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, HPDC '10, New York, NY, USA, 2010, ACM, pp. 288–291, doi:10.1145/1851476.1851512, http://doi.acm.org/10.1145/1851476.1851512.

[16] H. Ltaief, P. Luszczek, and J. Dongarra, *High-performance bidiagonal reduction using tile algorithms on homogeneous multicore architectures*, ACM Trans. Math. Softw., 39 (2013), pp. 16:1–16:22, doi:10.1145/2450153.2450154, http://doi.acm.org/10.1145/2450153.2450154.

[17] P. Ma, M. W. Mahoney, and B. Yu, *A statistical perspective on algorithmic leveraging*, J. Mach. Learn. Res., 16 (2015), pp. 861–911.

[18] X. Meng, J. K. Bradley, B. Yavuz, E. R. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. B. Tsai, M. Amde, S. Owen, D. Xin, R. Xin, M. J. Franklin, R. Zadeh, M. Zaharia, and A. Talwalkar, *MLlib: Machine Learning in Apache Spark*, CoRR, abs/1505.06807

(2015), http://arxiv.org/abs/1505.06807.
[19] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317, doi:10.1137/090752286, http://dx.doi.org/10.1137/090752286.
[20] T. White, *Hadoop: The Definitive Guide*, O'Reilly Media, Inc., 1st ed., 2009.
[21] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica, *Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing*, in Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation, NSDI'12, Berkeley, CA, USA, 2012, USENIX Association, pp. 2–2, http://dl.acm.org/citation.cfm?id=2228298.2228301.
[22] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, *Spark: Cluster computing with working sets*, in Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing, HotCloud'10, Berkeley, CA, USA, 2010, USENIX Association, pp. 10–10, http://dl.acm.org/citation.cfm?id=1863103.1863113.

Appl. Comput. Harmon. Anal. • • • (• • • •) • • •–• • •

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

Robust sparse phase retrieval made easy

Mark Iwen [a,*,1], Aditya Viswanathan [b], Yang Wang [c,2]

[a] Department of Mathematics and Department of ECE, Michigan State University, United States
[b] Department of Mathematics, Michigan State University, United States
[c] Department of Mathematics, The Hong Kong University of Science and Technology, Hong Kong

ARTICLE INFO

Article history:
Received 20 October 2014
Received in revised form 25 April 2015
Accepted 19 June 2015
Available online xxxx
Communicated by Thomas Strohmer

ABSTRACT

In this short note we propose a simple two-stage sparse phase retrieval strategy that uses a near-optimal number of measurements, and is both computationally efficient and robust to measurement noise. In addition, the proposed strategy is fairly general, allowing for a large number of new measurement constructions and recovery algorithms to be designed with minimal effort.

1. Introduction

Herein we consider the phase retrieval problem of reconstructing a given vector $\mathbf{x} \in \mathbb{C}^N$ from noisy magnitude measurements of the form

$$b_i := |\langle \mathbf{p}_i, \mathbf{x} \rangle|^2 + n_i, \tag{1}$$

where $\mathbf{p}_i \in \mathbb{C}^N$ is a measurement vector, and $n_i \in \mathbb{R}$ represents arbitrary measurement noise, for $i = 1, \ldots, M$. In particular, we focus on the setting where the dimension N is either very large, or else the number of measurements allowed, M, is otherwise severely restricted. In either case, our inability to gather the $M = \mathcal{O}(N)$ measurements required for the recovery of $\mathbf{x}$ in general [20] forces us to consider the possibility of approximating $\mathbf{x}$ using only $M \ll N$ magnitude measurements, if possible. This is the situation motivating the *compressive phase retrieval problem* (see, e.g., [30,31,26,24,34,15,32,35]), in which one attempts to accurately approximate $\mathbf{x} \in \mathbb{C}^N$ using only $M = o(N)$ magnitude measurements (1) under the assumption that $\mathbf{x}$ is either sparse, or compressible.

* Corresponding author.
E-mail addresses: markiwen@math.msu.edu (M. Iwen), aditya@math.msu.edu (A. Viswanathan), yangwang@ust.hk (Y. Wang).

[1] M.A. Iwen was supported in part by NSF DMS-1416752 and NSA H98230-13-1-0275.
[2] Y. Wang was partially supported by NSF DMS-1043032 and AFOSR FA9550-12-1-0455.

http://dx.doi.org/10.1016/j.acha.2015.06.007
1063-5203/

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

One question regarding the compressive phase retrieval problem is how many measurements are needed to allow for stable reconstruction of $\mathbf{x}$. Clearly, compressive phase retrieval requires at least as many measurements as the corresponding classical compressive sensing problem since one is given less information. Hence, stable compressive phase retrieval requires at least $\mathcal{O}(s\log(N/s))$ magnitude measurements[3] – but *can it be done with* $M = \mathcal{O}(s\log(N/s))$ *measurements*? It is shown in [15] that stable compressive phase retrieval is indeed achievable with $M = \mathcal{O}(s\log(N/s))$ measurements for *real* $\mathbf{x}$ if the entries of $\mathbf{p}_i$ are real independent and identically distributed (i.i.d.) Gaussians. However, this question was unresolved in the complex case. In this note we extend the result to the complex case. Furthermore, we do so in a constructive way by providing a computational procedure which can stably reconstruct complex $\mathbf{x}$ using only $\mathcal{O}(s\log(N/s))$ magnitude measurements.

Unlike previous sparse phase retrieval approaches, we propose a generic two-stage solution technique consisting of (i) using the phase retrieval technique of one's choice to recover compressive sensing measurements of $\mathbf{x}$, $\mathcal{C}\mathbf{x} \in \mathbb{C}^m$, followed by (ii) utilizing the compressive sensing method of one's choice in order to approximate $\mathbf{x}$ from the recovered measurements $\mathcal{C}\mathbf{x}$. As we shall see, the generic nature of the proposed sparse phase retrieval procedure not only allows for a relatively large number of measurement matrices and recovery algorithms to be used, but also allows robust recovery guarantees for the sparse phase retrieval problem to be proven in the complex setting essentially "for free" by combining existing robust recovery results from the compressive sensing literature with robust recovery results for the standard phase retrieval setting. As a result, we are able to show that $\mathcal{O}(s\log(N/s))$ magnitude measurements suffice in order to recover a large class of compressible vectors with the same quality of error guarantee as commonly achieved in the compressive sensing literature. Finally, numerical experiments demonstrate that the proposed approach is also both efficient and robust in practice.

2. Background

In this section we briefly recall selected results from the existing literature on compressive sensing [14,17] and phase retrieval [3,2,12,11,1,16]. Let $\|\mathbf{x}\|_0$ denote the number of nonzero entries in a given $\mathbf{x} \in \mathbb{C}^N$, and $\|\mathbf{x}\|_p$ denote the standard ℓ_p-norm of $\mathbf{x}$ for all $p \geq 1$, i.e., $\|\mathbf{x}\|_p := \left(\sum_{n=1}^N |x_n|^p\right)^{1/p}$ for all $\mathbf{x} \in \mathbb{C}^N$.

2.1. Compressive sensing

Compressive sensing methods deal with the construction of an $m \times N$ measurement matrix, $\mathcal{C}$, with m minimized as much as possible subject to the constraint that an associated approximation algorithm, $\Delta_{\mathcal{C}} : \mathbb{C}^m \to \mathbb{C}^N$, can still accurately approximate any given vector $\mathbf{x} \in \mathbb{C}^N$. More precisely, compressive sensing methods allow one to minimize m, the number of rows in $\mathcal{C}$, as a function of s and N such that

$$\|\Delta_{\mathcal{C}}(\mathcal{C}\mathbf{x}) - \mathbf{x}\|_p \leq C_{p,q} \cdot s^{\frac{1}{p}-\frac{1}{q}} \left(\inf_{\mathbf{z}\in\mathbb{C}^N, \|\mathbf{z}\|_0 \leq s} \|\mathbf{x}-\mathbf{z}\|_q\right) \tag{2}$$

holds for all $\mathbf{x} \in \mathbb{C}^N$ in various fixed ℓ_p,ℓ_q norms, $1 \leq q \leq p \leq 2$, for an absolute constant $C_{p,q} \in \mathbb{R}$ (e.g., see [13,17]). Note that this implies that $\mathbf{x}$ will be recovered exactly if it contains only s nonzero entries. Similarly, $\mathbf{x}$ will be accurately approximated by $\Delta_{\mathcal{C}}(\mathcal{C}\mathbf{x})$ any time its ℓ_q-norm is dominated by its largest s entries.

There are a wide variety of measurement matrices $\mathcal{C} \in \mathbb{C}^{m\times N}$ with $m = \mathcal{O}(s\log(N/s))$ that have associated approximation algorithms, $\Delta_{\mathcal{C}}$, which are computationally efficient, numerically robust, and able to achieve error guarantees of the form (2) for all $\mathbf{x} \in \mathbb{C}^N$. For example, this is true of "most" random matrices $\mathcal{C} \in \mathbb{C}^{m\times N}$ with i.i.d. subgaussian random entries [4,17]. Similarly, one may construct such a $\mathcal{C} \in \mathbb{C}^{m\times N}$ with

[3] See, e.g., Chapter 10 of [17] concerning the minimal number of measurements required for stable compressive sensing.

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

high probability by selecting a set of $m = \mathcal{O}(s \log^4 N)$ rows uniformly at random from an $N \times N$ discrete Fourier transform matrix (or, more generally, from any "sufficiently flat" $N \times N$ unitary matrix) [17]. In either case, one may then use a large number of approximation algorithms, $\Delta_{\mathcal{C}}$, that will achieve error guarantees along the lines of (2), including convex optimization techniques [8–10], iterative hard thresholding [7], (regularized) orthogonal matching pursuit [33,25,28,29], and the CoSaMP algorithm [27], to name just a few.

More generally, any matrix with the *robust null space property* [13] will have an associated approximation algorithm that is both computationally efficient and numerically robust. Let $S \subseteq \{1, 2, \ldots, N\}$, and $\mathbf{x} \in \mathbb{C}^N$. Then, $\mathbf{x}_S$ will denote $\mathbf{x}$ with all entries not in S set to zero. That is,

$$(x_S)_j := \begin{cases} 0, & \text{if } j \notin S, \\ x_j, & \text{if } j \in S. \end{cases}$$

The robust null space property can now be defined as follows.

Definition 1. Let $s, m, N \in \mathbb{N}$ be such that $s < m < N$. We will say that the matrix $\mathcal{C} \in \mathbb{C}^{m \times N}$ satisfies the ℓ_2-robust null space property of order s with constants $0 < \rho < 1$ and $\tau > 0$ if

$$\|\mathbf{x}_S\|_1 \leq \rho \|\mathbf{x}_{S^c}\|_1 + \tau \|\mathcal{C}\mathbf{x}\|_2$$

holds for all $\mathbf{x} \in \mathbb{C}^N$ and $S \subset \{1, 2, \ldots, N\}$ with cardinality $|S| \leq s$, where S^c denotes the complement of S.

In particular, the following robust compressive sensing result for matrices with the null space property is a restatement of Theorem 4.22 from [17].

Theorem 1. *Suppose that the matrix $\mathcal{C} \in \mathbb{C}^{m \times N}$ satisfies the ℓ_2-robust null space property of order s with constants $0 < \rho < 1$ and $\tau > 0$. Then, for any $\mathbf{x} \in \mathbb{C}^N$, the vector*

$$\tilde{\mathbf{x}} := \underset{\mathbf{z} \in \mathbb{C}^N}{\arg\min} \|\mathbf{z}\|_1 \text{ subject to } \|\mathcal{C}\mathbf{z} - \mathbf{y}\|_2 \leq \eta, \tag{3}$$

where $\mathbf{y} := \mathcal{C}\mathbf{x} + \mathbf{e}$ for some $\mathbf{e} \in \mathbb{C}^m$ with $\|\mathbf{e}\|_2 \leq \eta$, will satisfy

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_2 \leq \frac{C}{\sqrt{s}} \cdot \left(\inf_{\mathbf{z} \in \mathbb{C}^N, \|\mathbf{z}\|_0 \leq s} \|\mathbf{x} - \mathbf{z}\|_1 \right) + D\eta \tag{4}$$

for some constants $C, D \in \mathbb{R}^+$ that only depend on ρ and τ.

Many matrices exist with the ℓ_2-robust null space property including, e.g., "most" randomly constructed subgaussian and subsampled discrete Fourier transform matrices (as per above). Thus, in some sense it is not difficult to find a matrix $\mathcal{C} \in \mathbb{C}^{m \times N}$ to which Theorem 1 will apply. Furthermore, $\tilde{\mathbf{x}}$ from (3) can be computed efficiently via convex optimization techniques. See [17] for details.

2.2. Phase retrieval

Noisy phase retrieval problems involve the reconstruction of a given vector $\mathbf{x} \in \mathbb{C}^N$, up to a global phase factor, from magnitude measurements of the form

$$b_i := |\langle \mathbf{p}_i, \mathbf{x} \rangle|^2 + n_i, \tag{5}$$

where $\mathbf{p}_i \in \mathbb{C}^N$ and $n_i \in \mathbb{R}$ for $i = 1, \ldots, M$. Vectorizing (5) yields

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

$$\mathbf{b} := |\mathcal{P}\mathbf{x}|^2 + \mathbf{n}, \tag{6}$$

where $\mathbf{b}, \mathbf{n} \in \mathbb{R}^M$, $\mathcal{P} \in \mathbb{C}^{M\times N}$, and $|\cdot|^2 : \mathbb{C}^M \to \mathbb{R}^M$ computes the component-wise squared magnitude of each vector entry. Thus, the primary objective of phase retrieval is to construct a recovery algorithm, $\Phi_{\mathcal{P}} : \mathbb{R}^M \to \mathbb{C}^N$, that satisfies a relative error guarantee such as, e.g.,

$$\min_{\theta\in[0,2\pi]} \left(\frac{\left\| \Phi_{\mathcal{P}}(\mathbf{b}) - \mathbb{e}^{\mathbb{i}\theta}\mathbf{x} \right\|_2}{\|\mathbf{x}\|_2} \right)^q \leq C_{\mathcal{P}} \cdot \frac{\|\mathbf{n}\|_2}{\sqrt{M}\|\mathbf{x}\|_2^2} \tag{7}$$

for a particular measurement matrix $\mathcal{P} \in \mathbb{C}^{M\times N}$, $q \in [1, 2]$, and approximation factor $C_{\mathcal{P}} \in \mathbb{R}^+$ (which may depend on $\mathcal{P}$).

Several recovery algorithms achieve error guarantees along the lines of (7) while using at most $M = \mathcal{O}(N \log N)$ measurements, including both *PhaseLift* [12,11] as well as a more recent graph-theoretic and frame-based approach [1]. In particular, the following robust phase retrieval result is a variant of Theorem 1.3 from [11].[4]

Theorem 2. *Let $\mathcal{P} \in \mathbb{C}^{M\times N}$ have its M rows be independently drawn either uniformly at random from the sphere of radius $\sqrt{N}$ in $\mathbb{C}^N$, or else as complex normal random vectors from $\mathcal{N}(0, \mathcal{I}_N/2) + \mathbb{i}\mathcal{N}(0, \mathcal{I}_N/2)$. Then, $\exists$ universal constants $\tilde{B}, \tilde{C}, \tilde{D} \in \mathbb{R}^+$ such that the PhaseLift procedure $\Phi_{\mathcal{P}} : \mathbb{R}^M \to \mathbb{C}^N$ satisfies*

$$\min_{\theta\in[0,2\pi]} \left\| \Phi_{\mathcal{P}}(\mathbf{b}) - \mathbb{e}^{\mathbb{i}\theta}\mathbf{x} \right\|_2 \leq \tilde{C} \cdot \frac{\|\mathbf{n}\|_1}{M\|\mathbf{x}\|_2} \tag{8}$$

for all $\mathbf{x} \in \mathbb{C}^N$ with probability $1 - \mathcal{O}(\mathbb{e}^{-\tilde{B}M})$, provided that $M \geq \tilde{D}N$. Here $\mathbf{b}, \mathbf{n} \in \mathbb{R}^M$ are as in (6).

Finally, it is important to note that the *PhaseLift* procedure from Theorem 2 can be computed via semidefinite programming techniques. Thus, it is computationally tractable for modest dimensions, N. See [12,11] for details.

3. A simple two-stage technique for sparse phase retrieval

In this section we consider using noisy magnitude measurements of the form

$$\mathbf{b} := |\mathcal{P}\mathcal{C}\mathbf{x}|^2 + \mathbf{n}, \tag{9}$$

where $\mathcal{P} \in \mathbb{C}^{\tilde{m}\times m}$ is any phase retrieval matrix with an associated recovery algorithm $\Phi_{\mathcal{P}} : \mathbb{R}^{\tilde{m}} \to \mathbb{C}^m$ that has an error guarantee along the lines of (7), and $\mathcal{C} \in \mathbb{C}^{m\times N}$ is any compressive sensing matrix with an associated approximation algorithm $\Delta_{\mathcal{C}} : \mathbb{C}^m \to \mathbb{C}^N$ that has an error guarantee like (2). In this situation the composition of the two recovery algorithms, $\Delta_{\mathcal{C}} \circ \Phi_{\mathcal{P}} : \mathbb{R}^{\tilde{m}} \to \mathbb{C}^N$, should accurately approximate $\mathbf{x} \in \mathbb{C}^N$, up to a global phase factor, from $\mathbf{b}$ whenever $\mathbf{x}$ is sufficiently sparse or compressible. This leads us to the following intuitive observation.

Proposition 1. *Let $\mathcal{A} = \mathcal{P}\mathcal{C}$ where $\mathcal{C} \in \mathbb{C}^{m\times N}$ has the robust null space property and $\mathcal{P} \in \mathbb{C}^{\tilde{m}\times m}$ is a stable phase retrieval matrix. Then, $\mathcal{A}$ has the stable compressive phase retrieval property.*

More specifically, the following compressive phase retrieval result follows easily from Theorems 1 and 2.

[4] Equation (1.8) in Theorem 1.3 is technically incorrect as stated in [11]. See [22] for a corrected and simplified proof of Theorem 2 as stated herein.

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

Theorem 3. *Let $\mathcal{P} \in \mathbb{C}^{\tilde{m}\times m}$ have its $\tilde{m}$ rows be independently drawn either uniformly at random from the sphere of radius $\sqrt{m}$ in $\mathbb{C}^m$, or else as complex normal random vectors from $\mathcal{N}(0, \mathcal{I}_m/2) + \mathbb{i}\mathcal{N}(0, \mathcal{I}_m/2)$. Furthermore, suppose that $\mathcal{C} \in \mathbb{C}^{m\times N}$ satisfies the ℓ_2-robust null space property of order s with constants $0 < \rho < 1$ and $\tau > 0$. Then, there exists a phase retrieval procedure, $\Phi_{\mathcal{P}} : \mathbb{R}^{\tilde{m}} \to \mathbb{C}^m$, and a compressive sensing recovery algorithm, $\Delta_{\mathcal{C}} : \mathbb{C}^m \to \mathbb{C}^N$, such that*

$$\min_{\theta\in[0,2\pi]} \left\| \mathbb{e}^{\mathbb{i}\theta}\mathbf{x} - \Delta_{\mathcal{C}}\left(\Phi_{\mathcal{P}}\left(\mathbf{b}\right)\right)\right\|_2 \leq \frac{C}{\sqrt{s}} \cdot \left(\inf_{\mathbf{z}\in\mathbb{C}^N, \|\mathbf{z}\|_0 \leq s} \|\mathbf{x}-\mathbf{z}\|_1 \right) + D \cdot \frac{\|\mathbf{n}\|_1}{\tilde{m}\|\mathcal{C}\mathbf{x}\|_2} \tag{10}$$

holds for all $\mathbf{x} \in \mathbb{C}^N$ with probability $1 - \mathcal{O}(\mathbb{e}^{-B\tilde{m}})$, provided that $\tilde{m} \geq E \cdot m$. Here $\mathbf{b}, \mathbf{n} \in \mathbb{R}^{\tilde{m}}$ are as in (9), and $B, E \in \mathbb{R}^+$ are universal constants, while $C, D \in \mathbb{R}^+$ are constants that only depend on ρ and τ.

Considering the number of magnitude measurements required by Theorem 3, we note that $\tilde{m} = \mathcal{O}(s\log(N/s))$ such measurements will suffice to achieve (10) for all $\mathbf{x} \in \mathbb{C}^N$ with high probability whenever $\mathcal{C} \in \mathbb{C}^{m\times N}$ is, e.g., a random matrix with i.i.d. subgaussian random entries. In this situation $\mathcal{C}$ will also likely have both (i) the ℓ_2-robust null space property of order s with constants $0 < \rho < 1$ and $\tau > 0$, and (ii) a small *restricted isometry constant of order* $2s$, $\delta_{2s} < 1$ (see, e.g., §6.2 and §9.1 of [17] for details). As a consequence, $\mathcal{C}$ will also satisfy

$$\frac{1}{\tau} \cdot \max_{S\subset\{1,\ldots,N\}, |S|=s} \left(\|\mathbf{x}_S\|_1 - \rho\|\mathbf{x}_{S^c}\|_1\right) \leq \|\mathcal{C}\mathbf{x}\|_2 \leq \sqrt{1-\delta_{2s}}\left(\|\mathbf{x}\|_2 + \frac{\|\mathbf{x}\|_1}{\sqrt{2s}}\right) \tag{11}$$

for all $\mathbf{x} \in \mathbb{C}^N$ with high probability (w.h.p.).[5] Considering Theorem 3 error guarantee (10) in light of (11), we can now see that Theorem 3 implies that all sufficiently compressible vectors with, e.g.,

$$\frac{1}{\sqrt{\tilde{m}}} \leq \frac{1}{\tau} \cdot \max_{S\subset\{1,\ldots,N\}, |S|=s} \left(\|\mathbf{x}_S\|_1 - \rho\|\mathbf{x}_{S^c}\|_1\right) \tag{12}$$

will also satisfy

$$\min_{\theta\in[0,2\pi]} \left\| \mathbb{e}^{\mathbb{i}\theta}\mathbf{x} - \Delta_{\mathcal{C}}\left(\Phi_{\mathcal{P}}\left(\mathbf{b}\right)\right)\right\|_2 \leq \frac{C}{\sqrt{s}} \cdot \left(\inf_{\mathbf{z}\in\mathbb{C}^N, \|\mathbf{z}\|_0 \leq s} \|\mathbf{x}-\mathbf{z}\|_1 \right) + D\|\mathbf{n}\|_2 \tag{13}$$

w.h.p. whenever $\mathcal{C}$ is a random matrix with i.i.d. subgaussian entries.

Finally, it is interesting to note that the two-stage approach outlined in this section also confers some computational advantages. Mainly, the phase retrieval recovery algorithm $\Phi_{\mathcal{P}} : \mathbb{R}^{\tilde{m}} \to \mathbb{C}^m$ only needs to recover a vector of length $m = \mathcal{O}(s\log(N/s))$. This allows phase retrieval approaches based on, e.g., semidefinite programming to efficiently approximate significantly larger vectors $\mathbf{x} \in \mathbb{C}^N$ than otherwise possible when $N \gg s$.

4. Empirical evaluation

We now present representative results demonstrating the numerical robustness and efficiency of the proposed two-step strategy. For the results in this section, we use *PhaseLift* [12,11] and Basis Pursuit [9] to solve the phase retrieval and compressive sensing problems in steps (i) and (ii), respectively. Moreover, we use complex Gaussian phase retrieval matrices $\mathcal{P}$ and real Gaussian compressive sensing matrices $\mathcal{C}$. Matlab code used to generate the numerical results – implemented using the optimization software packages TFOCS [6,5] and CVX [19,18] – is freely available at [23].

[5] The lower bound is a simple consequence of Definition 1. For the upper bound see, e.g., Exercise 6.6 in [17].

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

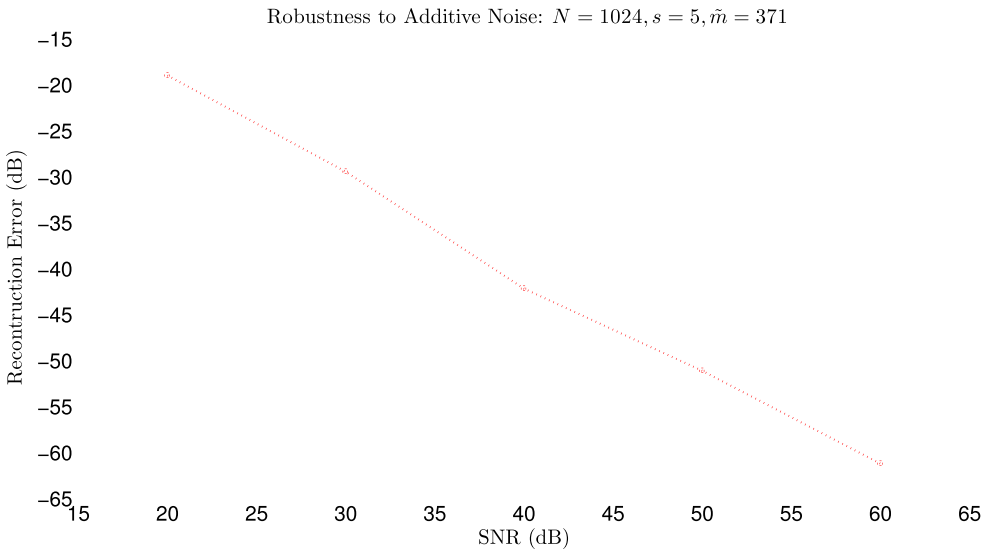


Fig. 1. Robustness to additive noise: $N = 1024, s = 5, \tilde{m} = \lceil 14s \log(N/s) \rceil$.

In each of the following results, we recover sparse, unit-norm complex vectors whose non-zero indices are independently and randomly chosen, and, whose non-zero entries are i.i.d. standard complex Gaussians.

Fig. 1 illustrates the robustness of the recovery procedure to additive noise. We add i.i.d. zero-mean Gaussian noise at several signal-to-noise ratios (SNRs) to $\tilde{m} = \lceil 14s \log(N/s) \rceil$ magnitude measurements ($N = 1024, s = 5, \tilde{m} = 371$) and record relative reconstruction errors in decibels. Each data point on the graph was obtained by averaging the results of 100 trials. We observe that the reconstruction error in every case is approximately equal to the added noise level, confirming the robust recovery properties of the proposed method.

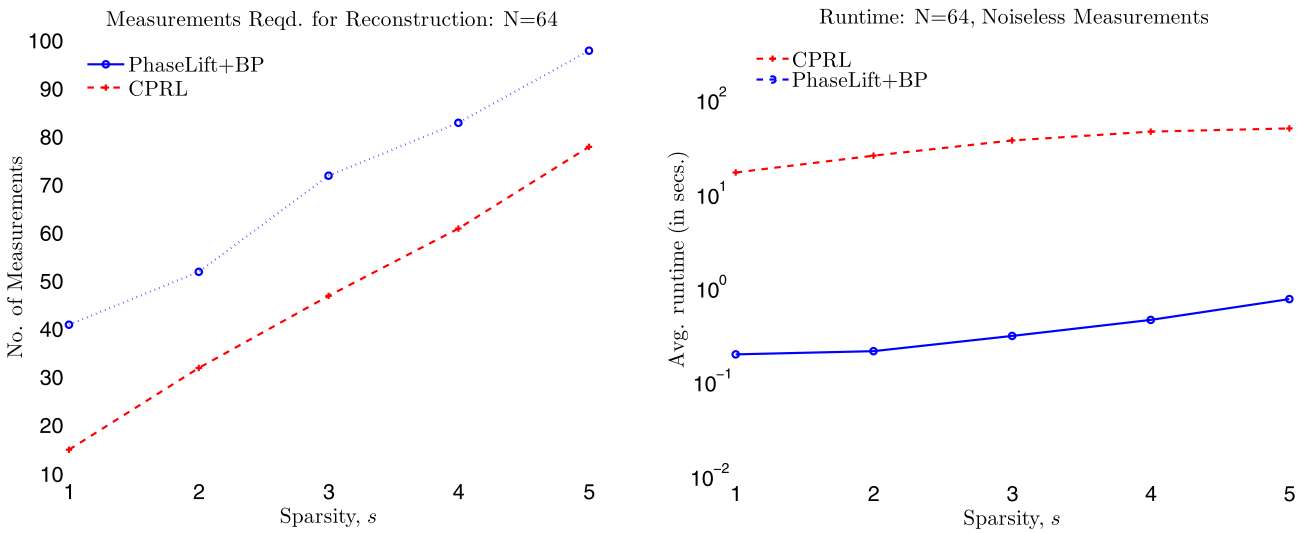
Next, we demonstrate efficiency by plotting the average runtime and minimum number of measurements necessary for successful reconstruction. For the purposes of this discussion, we classify a reconstruction as successful if the relative ℓ_2-norm error in the recovered signal is less than 10^{-5}. We also provide comparisons with Compressive Phase Retrieval via Lifting (*CPRL*) [30], an existing framework for sparse phase retrieval. Simulations were performed on a laptop computer with an Intel® Core™ i3-3120M processor, 4 GB RAM and Matlab R2014a. We first consider the reconstruction of an s-sparse signal ($N = 64$) from perfect (noiseless) measurements. The minimum number of measurements[6] required for successful reconstruction is plotted in Fig. 2a, while the corresponding runtime, averaged over 100 trials, is plotted in Fig. 2b. Fig. 2a was generated by starting with a small number of measurements, $\tilde{m}$, and incrementing this number to ensure successful reconstruction in at least 95 of the 100 trials. We notice that the *PhaseLift*+BP formulation requires a small number of additional measurements when compared to *CPRL*. This is potentially only the case for small values of s since Theorem 3 shows that $\mathcal{O}(s \log(N/s))$ measurements suffice for the *PhaseLift*+BP formulation. Moreover, since the *PhaseLift*+BP solution is obtained by solving a smaller SDP, the average runtime is significantly smaller (by several orders of magnitude) than *CPRL*, as shown in Fig. 2b.

5. Discussion

It is interesting to note that the compressive phase retrieval strategy discussed herein also immediately implies the existence of stable sublinear-time compressive phase retrieval algorithms. These can be achieved by combining the phase retrieval technique of one's choice with a $o(N)$-time compressive sensing method

[6] For the *PhaseLift*+BP implementation, we fixed the compressive sensing problem dimension to be $m = \lceil 1.75s \log(N/s) \rceil$.

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007



(A) Minimum number of measurements (B) Runtime comparison

Fig. 2. Runtime performance and minimum number of measurements required: $N = 64$, noiseless measurements.

(see, e.g., [21]) in order to create a $o(N)$-time compressive phase retrieval algorithm. In addition, we conclude by noting that random combinations of a random set of rows from a Fourier matrix will also exhibit the stable compressive phase retrieval property by Proposition 1/Theorem 3. This is of particular interest due to the special role that Fourier measurements play in many applications.

References

[1] B. Alexeev, A.S. Bandeira, M. Fickus, D.G. Mixon, Phase retrieval with polarization, SIAM J. Imaging Sci. 7 (1) (2014) 35–66.
[2] R. Balan, B.G. Bodmann, P.G. Casazza, D. Edidin, Painless reconstruction from magnitudes of frame coefficients, J. Fourier Anal. Appl. 15 (4) (2009) 488–501.
[3] R. Balan, P. Casazza, D. Edidin, On signal reconstruction without phase, Appl. Comput. Harmon. Anal. 20 (3) (2006) 345–356.
[4] R. Baraniuk, M. Davenport, R. DeVore, M. Wakin, A simple proof of the restricted isometry property for random matrices, Constr. Approx. 28 (3) (2008) 253–263.
[5] S. Becker, E.J. Candes, M. Grant, Templates for convex cone problems with applications to sparse signal recovery, Math. Program. Comput. 3 (3) (2011) 165–218.
[6] S. Becker, E.J. Candes, M. Grant, TFOCS: templates for first-order conic solvers, version 1.3.1, http://cvxr.com/tfocs, 2014.
[7] T. Blumensath, M.E. Davies, Iterative hard thresholding for compressed sensing, Appl. Comput. Harmon. Anal. 27 (3) (2009) 265–274.
[8] E. Candes, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inform. Theory 52 (2006) 489–509.
[9] E. Candes, J. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Comm. Pure Appl. Math. 59 (8) (2006) 1207–1223.
[10] E.J. Candes, T. Tao, Near optimal signal recovery from random projections: universal encoding strategies?, IEEE Trans. Inform. Theory 52 (12) (2006) 5406–5425.
[11] E.J. Candes, X. Li, Solving quadratic equations via PhaseLift when there are about as many equations as unknowns, Found. Comput. Math. 14 (5) (2014) 1017–1026.
[12] E.J. Candes, T. Strohmer, V. Voroninski, PhaseLift: exact and stable signal recovery from magnitude measurements via convex programming, Comm. Pure Appl. Math. 66 (8) (2013) 1241–1274.
[13] A. Cohen, W. Dahmen, R. DeVore, Compressed sensing and best k-term approximation, J. Amer. Math. Soc. 22 (1) (2009) 211–231.
[14] D.L. Donoho, Compressed sensing, IEEE Trans. Inform. Theory 52 (4) (2006) 1289–1306.
[15] Y.C. Eldar, S. Mendelson, Phase retrieval: stability and recovery guarantees, Appl. Comput. Harmon. Anal. 36 (3) (2014) 473–494.
[16] M. Fickus, D.G. Mixon, A.A. Nelson, Y. Wang, Phase retrieval from very few measurements, Linear Algebra Appl. 449 (2014) 475–499.
[17] S. Foucart, H. Rauhut, A Mathematical Introduction to Compressive Sensing, Springer, 2013.
[18] M. Grant, S. Boyd, Graph implementations for nonsmooth convex programs, in: V. Blondel, S. Boyd, H. Kimura (Eds.), Recent Advances in Learning and Control, in: Lecture Notes in Control and Information Sciences, Springer-Verlag Limited, 2008, pp. 95–110, http://stanford.edu/~boyd/graph_dcp.html.

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

[19] M. Grant, S. Boyd, CVX: matlab software for disciplined convex programming, version 2.1, http://cvxr.com/cvx, 2014.
[20] T. Heinosaari, L. Mazzarella, M.M. Wolf, Quantum tomography under prior information, Comm. Math. Phys. 318 (2) (2013) 355–374.
[21] M. Iwen, Compressed sensing with sparse binary matrices: instance optimal error guarantees in near-optimal time, J. Complexity 30 (1) (2014) 1–15.
[22] M. Iwen, F. Krahmer, A. Viswanathan, Technical note: a minor correction of Theorem 1.3 from [1], http://www.math.msu.edu/~markiwen/Papers/PhaseLiftproof.pdf, 2015.
[23] M. Iwen, Y. Wang, A. Viswanathan, SparsePR: matlab software for sparse phase retrieval, version 1.0, https://bitbucket.org/charms/sparsepr, 2014.
[24] K. Jaganathan, S. Oymak, B. Hassibi, Sparse phase retrieval: convex algorithms and limitations, in: Proceedings of the 2013 IEEE International Symposium on Information Theory, ISIT, IEEE, 2013, pp. 1022–1026.
[25] S. Kunis, H. Rauhut, Random sampling of sparse trigonometric polynomials II – orthogonal matching pursuit versus basis pursuit, Found. Comput. Math. 8 (6) (2008) 737–763.
[26] X. Li, V. Voroninski, Sparse signal recovery from quadratic measurements via convex programming, SIAM J. Math. Anal. 45 (5) (2013) 3019–3033.
[27] D. Needell, J.A. Tropp, CoSaMP: iterative signal recovery from incomplete and inaccurate samples, Appl. Comput. Harmon. Anal. 26 (3) (2009) 301–321.
[28] D. Needell, R. Vershynin, Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit, Found. Comput. Math. 9 (2009) 317–334.
[29] D. Needell, R. Vershynin, Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit, IEEE J. Sel. Top. Signal Process. (2010) 310–316.
[30] H. Ohlsson, A. Yang, R. Dong, S. Sastry, Cprl: an extension of compressive sensing to the phase retrieval problem, in: Proceedings of the 26th Conference on Advances in Neural Information Processing Systems, 2012, pp. 1376–1384.
[31] P. Schniter, S. Rangan, Compressive phase retrieval via generalized approximate message passing, in: Proc. Allerton Conf. on Communication, Control, and Computing, 2012.
[32] Y. Shechtman, A. Beck, Y.C. Eldar, Gespar: efficient phase retrieval of sparse signals, IEEE Trans. Signal Process. 62 (4) (2014) 928–938.
[33] J. Tropp, A. Gilbert, Signal recovery from partial information via orthogonal matching pursuit, IEEE Trans. Inform. Theory 53 (12) (Dec. 2007) 4655–4666.
[34] Y. Wang, Z. Xu, Phase retrieval for sparse signals, Appl. Comput. Harmon. Anal. 37 (3) (2014) 531–544.
[35] Ç. Yapar, V. Pohl, H. Boche, Fast compressive phase retrieval from Fourier measurements, arXiv:1410.7351, 2014.

Please cite this article in press as: M. Iwen et al., Robust sparse phase retrieval made easy, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.06.007

Appl. Comput. Harmon. Anal. • • • (• • • •) • • •–• • •

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

A multiscale sub-linear time Fourier algorithm for noisy data

Andrew Christlieb [a,1], David Lawlor [b,c,*], Yang Wang [a,2]

[a] *Department of Mathematics, Michigan State University, East Lansing, MI 48824, United States*
[b] *Statistical and Applied Mathematical Sciences Institute, 19 T. W. Alexander Drive, P.O. Box 14006, Research Triangle Park, NC 27709-4006, United States*
[c] *Department of Mathematics, Duke University, Box 90320, Durham, NC 27708-0320, United States*

ARTICLE INFO

Article history:
Received 25 March 2014
Received in revised form 10 October 2014
Accepted 12 April 2015
Available online xxxx
Communicated by Gregory Beylkin

Keywords:
Fast Fourier algorithms
Multiscale algorithms
Fourier analysis

ABSTRACT

We extend the recent sparse Fourier transform algorithm of [1] to the noisy setting, in which a signal of bandwidth N is given as a superposition of $k \ll N$ frequencies and additive random noise. We present two such extensions, the second of which exhibits a form of error-correction in its frequency estimation not unlike that of the β-encoders in analog-to-digital conversion [2]. On k-sparse signals corrupted with additive complex Gaussian noise, the algorithm runs in time $O(k \log(k) \log(N/k))$ on average, provided the noise is not overwhelming. The error-correction property allows the algorithm to outperform FFTW [3], a highly optimized software package for computing the full discrete Fourier transform, over a wide range of sparsity and noise values.

1. Introduction

The Fast Fourier Transform (FFT) [4] is a fundamental numerical algorithm whose importance in a wide variety of applications cannot be overstated. The FFT reduces the runtime complexity of calculating the discrete Fourier transform (DFT) of a length N array from the naive $O(N^2)$ to $O(N \log(N))$. At the time of its introduction in the mid-1960s, it dramatically increased the size of problems that a typical computer could handle. Over the past fifty years the typical size of data sets has grown by orders of magnitude, and in certain application areas (e.g. cognitive radio and ultra-wideband radar [5,6]) the computation of the full FFT is no longer tractable on commodity hardware. In this and other instances, however, it is known a priori that the signals of interest have small frequency support; that is, their Fourier transforms are *sparse*. This problem has received attention from a number of research communities over the past decade, who have

* Corresponding author at: Department of Mathematics, Duke University, Box 90320, Durham, NC 27708-0320, United States. Fax: +919 685 9360.
E-mail addresses: christlieb@math.msu.edu (A. Christlieb), djl@math.duke.edu (D. Lawlor), ywang@math.msu.edu (Y. Wang).
[1] Fax: +517 432 1562.
[2] Fax: +517 355 5279.

http://dx.doi.org/10.1016/j.acha.2015.04.002
1063-5203/

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

shown that it is possible to significantly outperform the FFT in both runtime and sampling requirements when the number of significant Fourier modes k is much less than the nominal bandwidth N. Early works addressing this topic from the perspective of learning boolean functions include [7,8].

The sparse Fourier transform problem was first studied explicitly in [9,10], the latter of which gave a randomized algorithm with runtime and sampling complexity $\mathrm{O}(k^2\,\mathrm{polylog}(N))$.[3] This was later improved to $\mathrm{O}(k\,\mathrm{polylog}(N))$ [11] through the use of unequally-spaced FFTs [12]. For a given failure probability δ and accuracy parameter ε, the algorithm returns a k-term approximation $\hat{y}$ to the DFT of the input $\hat{x}$ such that with probability $1-\delta$ it holds that

$$\|\hat{x}-\hat{y}\|_2^2 \leq (1+\varepsilon)\|\hat{x}-\hat{x}_k\|_2^2. \tag{1}$$

Here $\hat{x}_k$ is the best k-term approximation to $\hat{x}$ and $\|\cdot\|_2$ is the discrete ℓ_2 norm. In [13], a randomized $\mathrm{O}(k^2\mathrm{polylog}(N))$ algorithm for the sparse Fourier transform problem was given in the context of list decoding.[4] A separate group of authors [14] has developed a modified version of the algorithm of [11] with runtime $\mathrm{O}(\log(N)\sqrt{Nk\log(N)})$. While the dependence on N is sub-optimal asymptotically, in practice this algorithm is significantly faster than either [10] or [11]. The same authors presented an improved algorithm with runtime $\mathrm{O}(k\log(N)\log(N/k))$ in [15] whose frequency identification procedure is very similar to [1], upon which the present work is based. However, the performance of [15] in the presence of noise has yet to be evaluated empirically.

The algorithms described in the previous paragraph are all randomized, and so will fail on each signal with positive probability. Recognizing this as a potential detriment in failure-intolerant applications, two authors have independently given deterministic algorithms for the sparse Fourier transform problem. In [16,17] an algorithm with $\mathrm{poly}(k,\log(N))$ runtime was given where the exponent on k is at least six.[5] This high dependence on k renders the algorithm infeasible in practice, and it has not been implemented. However, we note that algorithms of [18,16,17] address a strictly wider class of signals than those with k-sparse Fourier spectrum, specifically those satisfying $\|\widehat{S}\|_1/\|\widehat{S}\|_2 \leq \mathrm{polylog}(N)$. In [19], the combinatorial properties of aliasing among frequencies were exploited to give an algorithm with runtime and sampling complexity $\mathrm{O}(k^2\,\mathrm{polylog}(N))$. While this represented a major improvement over the theoretical runtime complexity of [16], in practice it only outperformed the FFT for relatively modest values of the sparsity k.

Most recently the authors of [1] gave a deterministic algorithm whose sampling and runtime complexity are $\mathrm{O}(k\log(k))$ in the average case and $\mathrm{O}(k^2\log(k))$ in the worst case. The worst-case bounds are asymptotically of the same order in k (up to log factors) as [19], but over a representative class of random signals it was shown to significantly outperform its deterministic and randomized competitors. This was achieved by sampling the input at two sets of equispaced points slightly offset in time. This time shift appears in the Fourier domain as a frequency modulation, which allows the authors to both detect when aliasing has occurred and, for frequencies that are isolated (i.e. not aliased), to calculate the frequency value directly. While [19] also uses properties of aliasing to reconstruct frequency values, it is not able to distinguish between aliased and non-aliased terms until sufficiently many DFTs of coprime lengths have been computed, and so is unable to perform any better in the average case than in the worst case. In the empirical evaluation of [1] an improvement of over two orders of magnitude was observed over [11] and [19].

In this paper we extend the algorithm of [1] to noisy environments in two distinct ways. The first of these, which is a minor modification of the noiseless algorithm, is based on a certain rounding of the frequency estimates and was previously reported in [1]. In this work we provide an improved algorithm and more detailed analysis of that earlier work. The second extension is the main result of this paper

[3] We write $f=\mathrm{polylog}(g)$ to indicate that $f=\mathrm{O}(\log^c(g))$ for some unspecified constant c.
[4] The runtime of this algorithm was incorrectly stated as $\mathrm{O}(k^{11/2}\mathrm{polylog}(N))$ in [1].
[5] This algorithm is a de-randomization of the randomized algorithm presented in [18].

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

(summarized in Algorithm 1 and Theorem 4.5), a multiscale error-correcting algorithm that utilizes offset time samples at geometrically spaced time shifts. This extension is in essence a progressive frequency identification algorithm not unlike the β-encoders for analog-to-digital conversion [2]. While prior works have utilized multiscale frequency identification procedures [7–9,13,18,16,17], the connection to β-encoders is to the best of our knowledge novel. The new algorithm gives excellent performance in the noisy setting without significantly increasing the computational costs from the noiseless case. For both extensions we provide detailed mathematical analysis as well as empirical evaluations. While both extensions work well in the noisy environment, the multiscale algorithm achieves comparable accuracy at a significantly lower computational cost.

It should be emphasized that our algorithm assumes access to an underlying continuous signal $S(t)$, $t \in [0,1]$, rather than a discrete set of equidistant samples of S, which is the setting for the previous scholarship mentioned above. Indeed, this assumption is critical for our multiscale algorithm, as it allows the separation of nearby modes by sampling at finer scales. While this makes comparisons with other algorithms less straightforward, the assumption is valid in several application domains, including ultra-wideband radar [6]. It should also be noted that, as we discuss briefly in Section 5.4, a trivial modification of our multiscale algorithm is able to recover a single non-integral frequency in random noise. Other works addressing this problem assume a minimum separation between modes [20,21], which is the effect achieved by our continuous signal assumption.

The remainder of this paper is organized as follows. In Section 2 we review the notation introduced in [1] that will be necessary in the sequel. We also describe our noise model, discuss some of the problems noisy signals present for the algorithm of [1], and argue that in certain applications the ℓ_2 error metric is inappropriate and should be replaced with a form of Earth Mover's Distance. We also describe the random signal model used in the empirical evaluations in Section 5. In Section 3 we give our first modified algorithm and analyze the dependence of the sampling rate on the noise level. In Section 4 we describe our multiscale frequency identification procedure, and in Section 5 we provide an empirical evaluation of the accuracy and speed of both algorithms. Finally in Section 6 we provide a brief conclusion.

2. Preliminaries

2.1. Notation and brief review

In this section we introduce the notation that will be used in the remainder of this paper and briefly review the results in [1]. We denote by $\mathbb{Z}$ the set of integers, $\mathbb{C}$ the set of complex numbers, and we let N be a fixed (large) natural number. We write $\lfloor x \rfloor$ to denote the largest integer less than or equal to x. All logarithms are in base two unless explicitly specified.

We consider frequency-sparse band-limited signals $S : [0,1) \to \mathbb{C}$ of the form

$$S(t) = \sum_{\omega \in \Omega} a_\omega \mathrm{e}^{2\pi \mathrm{i} \omega t}, \tag{2}$$

where Ω is a finite set of integers bounded in $[-N/2, N/2)$ and $0 \neq a_\omega \in \mathbb{C}$ for each $\omega \in \Omega$. Denote by $a_{\min} = \min\{|a_\omega| : \omega \in \Omega\}$. For simplicity we shall extend $S(t)$ periodically to a function on the whole real line. The Fourier samples of S are given by

$$\widehat{S}(h) = \int_0^1 S(t) \mathrm{e}^{-2\pi \mathrm{i} h t} \mathrm{d}t, \ h \in \mathbb{Z}, \tag{3}$$

so that for signals of the form (2) we have $\widehat{S}(\omega) = a_\omega$ for $\omega \in \Omega$ and $\widehat{S}(h) = 0$ for all other $h \in \mathbb{Z}$.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

In practice we work with data of finite length. Given any finite sequence $\mathbf{s} = (s_0, s_1, \ldots, s_{p-1})$ of length p its DFT is given by

$$\widehat{\mathbf{s}}[h] = \sum_{j=0}^{p-1} s_j \mathrm{e}^{-2\pi \mathrm{i} jh/p} = \sum_{j=0}^{p-1} \mathbf{s}[j] W_p^{jh}, \tag{4}$$

where $h = 0, 1, \ldots, p-1$, $\mathbf{s}[j] := s_j$ and $W_p := \mathrm{e}^{-2\pi \mathrm{i}/p}$ is the primitive p-th root of unity. The FFT [4] allows the computation of $\widehat{\mathbf{s}}$ in $\mathrm{O}(p \log p)$ steps.

To obtain a fast reconstruction algorithm we apply the DFT to selected finite sample sets of $S(t)$. Let p be a positive integer and $\varepsilon > 0$. The two sample sets we use extensively are $\mathbf{S}_p$ and $\mathbf{S}_{p,\varepsilon}$, which are length p samples of $S(t)$ given by

$$\mathbf{S}_p[j] = S\Big(\frac{j}{p}\Big), \quad \mathbf{S}_{p,\varepsilon}[j] = S\Big(\frac{j}{p} + \varepsilon\Big), \quad j = 0, 1, \ldots, p-1. \tag{5}$$

For each h let $\Lambda_{p,h} = \{\omega \in \Omega : \omega \equiv h \pmod p\}$, where $\omega \equiv h \pmod p$ indicates that $\omega - h$ is divisible by p. It is a simple derivation to obtain

$$\widehat{\mathbf{S}}_p[h] = p \sum_{\omega \in \Lambda_{p,h}} a_\omega, \quad \widehat{\mathbf{S}}_{p,\varepsilon}[h] = p \sum_{\omega \in \Lambda_{p,h}} a_\omega \mathrm{e}^{2\pi \mathrm{i} \varepsilon \omega}. \tag{6}$$

Let $\omega \pmod p$ indicate the remainder after division of ω by p. In the ideal scenario where all $\{\omega \pmod p : \omega \in \Omega\}$ are distinct we have

$$\widehat{\mathbf{S}}_p[h] = \begin{cases} p a_\omega & h = \omega \pmod p \text{ for some } \omega \in \Omega, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

and similarly

$$\widehat{\mathbf{S}}_{p,\varepsilon}[h] = \begin{cases} p a_\omega \mathrm{e}^{2\pi \mathrm{i} \varepsilon \omega} & h = \omega \pmod p \text{ for some } \omega \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Thus, the nonzero elements of $\widehat{\mathbf{S}}_p[h]$ occur precisely at the locations $h = \omega \pmod p$ for some $\omega \in \Omega$, and moreover for such h we have $|\widehat{\mathbf{S}}_p[h]| = |\widehat{\mathbf{S}}_{p,\varepsilon}[h]|$. Furthermore for each $\omega \in \Omega$ and $h = \omega \pmod p$ we have $\frac{\widehat{\mathbf{S}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}_p[h]} = \mathrm{e}^{2\pi \mathrm{i} \varepsilon \omega}$. Hence

$$2\pi\varepsilon\omega \equiv \operatorname{Arg}\left(\frac{\widehat{\mathbf{S}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}_p[h]}\right) \pmod{2\pi}, \tag{9}$$

where $\operatorname{Arg}(z)$ denotes the phase angle of the complex number z in $[-\pi, \pi)$. Now assume that we have $|\varepsilon| \le \frac{1}{N}$. Then ω is completely determined by (9), as there will be no wrap-around aliasing. Hence

$$\omega = \frac{1}{2\pi\varepsilon} \operatorname{Arg}\left(\frac{\widehat{\mathbf{S}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}_p[h]}\right). \tag{10}$$

The weight a_ω can be recovered via $a_\omega = \widehat{\mathbf{S}}_p[h]/p$. In fact, more generally, if we have an estimate of $\omega \in \Omega$, say $|\omega| < \frac{L}{2}$, then by taking $\varepsilon \le \frac{1}{L}$ the same reconstruction formula (10) holds. We will use this observation in Section 4 when we develop a multiscale frequency identification procedure for noisy signals.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

Of course it is possible that not all $\{\omega \pmod p : \ \omega \in \Omega\}$ are distinct. For an $\omega \in \Omega$ we say ω *has a collision modulo* p, or simply *has a collision* when there is no ambiguity in the modulus p, if there is at least one other $\omega' \in \Omega$ such that $\omega \equiv \omega' \pmod p$. In [1] a criterion is developed to detect collisions in the noiseless case. For $\omega \in \Omega$ and $h = \omega \pmod p$, it is clear that a necessary condition for no collision to occur is

$$\left|\frac{\widehat{\mathbf{S}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}_p[h]}\right| = |\mathrm{e}^{2\pi \mathrm{i}\varepsilon\omega}| = 1. \tag{11}$$

It is shown in [1] that for a randomly chosen $\varepsilon > 0$ the converse holds with probability one, and furthermore checking the condition (11) for several ε would be sufficient to deterministically decide whether ω has a collision. In Section 4 we use this latter observation to devise a robust test for collisions even in the presence of noise.

The algorithm developed in [1] for recovering $S(t)$ is as follows: First we pick a prime $p = p_1$, which is roughly $5k$ where $k = |\Omega|$ is the number of modes in $S(t)$ (k is commonly referred to as the *sparsity* of $S(t)$). By taking $p \geq 5k$ we ensure that on average collisions do not occur for more than 90% of $\omega \in \Omega$. Let Ω' denote the subset of Ω consisting of all non-collision $\omega \in \Omega$. For each $\omega \in \Omega'$ we recover $a_\omega \mathrm{e}^{2\pi \mathrm{i}\omega t}$, and update $S(t)$ to

$$S_1(t) = S(t) - \sum_{\omega \in \Omega'} a_\omega \mathrm{e}^{2\pi \mathrm{i}\omega t}. \tag{12}$$

We now apply the above procedure again for $S_1(t)$ with a different prime $p = p_2$ approximately in the range of $5k_1$, where $k_1 = k - |\Omega'|$ is now the sparsity for $S_1(t)$. This process is repeated until all modes are found.

In the implementation of the algorithm we set a small threshold in (11) to check for collisions. This means there is a small probability that a collision is undetected by our criterion and a false value ω_0 is put into Ω' when it shouldn't be. In subsequent iterations, this will create a new mode $-c_0 \mathrm{e}^{2\pi \mathrm{i}\omega_0 t}$ for some $c_0 \in \mathbb{C}$ in $S_1(t)$. By the use of different primes p_j in each iteration this false mode will be identified and subtracted from the final reconstruction. In Section 4.3 we provide an improved aliasing test for our multiscale algorithm which makes the inclusion of spurious frequencies even less likely. However, it is still possible that incorrect modes are inserted before being deleted in the high-noise regime, as we discuss in Section 5.3.

2.2. Noise model

In a number of potential application areas for sparse Fourier algorithms, the samples collected will be corrupted by noise. One example of sparse Fourier transforms being used on real data is given in [22], where an application to faster GPS location is presented. Several previous works have considered the sparse Fourier transform problem for noisy signals, including the case of adversarial noise in [7,13].[6] Random noise models have been considered in [15,18,16,17], although the algorithms presented in those works for noisy signals have yet to be implemented and evaluated empirically.

In this paper we assume an i.i.d. noise model

$$\mathbf{S}^{\mathrm{n}}_p[j] = S\left(\frac{j}{p}\right) + \mathrm{n}_j = \mathbf{S}_p[j] + \mathrm{n}_j, \tag{13}$$

where $\mathrm{n} = (\mathrm{n}_j)$ are i.i.d. complex random variables with mean 0 and variance σ^2. A typical model is to assume $\{\mathrm{n}_j\}$ are i.i.d. complex Gaussian. With this noise model we have

[6] The algorithm of [10] implicitly addresses this challenging setting as well.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

$$\widehat{\mathbf{S}}^{\mathrm{n}}_p[h] = \widehat{\mathbf{S}}_p[h] + \widehat{\mathrm{n}}[h], \tag{14}$$

where

$$\widehat{\mathrm{n}}[h] = \sum_{j=0}^{p-1} \mathrm{n}_j \mathrm{e}^{-2\pi \mathrm{i} h j/p}. \tag{15}$$

By the i.i.d. property for $\{\mathrm{n}_j\}$ we have for each h

$$\mathbb{E}\big[\widehat{\mathrm{n}}[h]\big] = 0 \tag{16}$$

and

$$\mathrm{Var}\big[\widehat{\mathrm{n}}[h]\big] = p\sigma^2, \tag{17}$$

where the expectations are taken with respect to the randomness in the noise. This yields

$$\mathbb{E}\left[\widehat{\mathbf{S}}^{\mathrm{n}}_p[h]\right] = \widehat{\mathbf{S}}_p[h] \tag{18}$$

and

$$\mathbb{E}\left[|\widehat{\mathbf{S}}^{\mathrm{n}}_p[h] - \widehat{\mathbf{S}}_p[h]|^2\right] = p\sigma^2. \tag{19}$$

Thus, a typical noisy DFT coefficient $\widehat{\mathbf{S}}^{\mathrm{n}}_p[h]$ will deviate from the true value $\widehat{\mathbf{S}}_p[h]$ by an amount proportional to $\sigma\sqrt{p}$. Similarly, for $\mathbf{S}^{\mathrm{n}}_{p,\varepsilon} = \mathbf{S}_{p,\varepsilon} + \mathrm{n}_\varepsilon$ we will have

$$\mathbb{E}\left[\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]\right] = \widehat{\mathbf{S}}_{p,\varepsilon}[h] \tag{20}$$

and

$$\mathbb{E}\left[|\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h] - \widehat{\mathbf{S}}_{p,\varepsilon}[h]|^2\right] = p\sigma^2. \tag{21}$$

We now pick a non-collision $\omega \in \Omega$. Then for $h = \omega \pmod p$ we will have

$$\begin{aligned} \widehat{\mathbf{S}}^{\mathrm{n}}_p[h] &= pa_\omega + \mathrm{O}(\sigma\sqrt{p}), \\ \widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h] &= pa_\omega \mathrm{e}^{2\pi \mathrm{i}\omega\varepsilon} + \mathrm{O}(\sigma\sqrt{p}). \end{aligned} \tag{22}$$

As a result a_ω can now be estimated easily via

$$a_\omega = \frac{1}{p}\widehat{\mathbf{S}}^{\mathrm{n}}_p[h] + \mathrm{O}\Big(\frac{\sigma}{\sqrt{p}}\Big). \tag{23}$$

The real challenge lies in the recovery of the frequencies in Ω. Assume that $|\widehat{\mathbf{S}}_{p,\varepsilon}|$ has a peak at h. Then $h = \omega \pmod p$ for some $\omega \in \Omega$. If there is no collision for ω, in the noiseless environment ω is recovered via (10) as long as $\varepsilon \leq \frac{1}{N}$. In the noisy setting $\widehat{\mathbf{S}}_{p,\varepsilon}[h]/\widehat{\mathbf{S}}_p[h]$ must be replaced by $\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]/\widehat{\mathbf{S}}^{\mathrm{n}}_p[h]$. Interestingly, the mean of $\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]/\widehat{\mathbf{S}}^{\mathrm{n}}_p[h]$ is in general *not* $\widehat{\mathbf{S}}_{p,\varepsilon}[h]/\widehat{\mathbf{S}}_p[h]$ as a result of the division. Nevertheless we have

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

$$\begin{aligned}\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]} &= \frac{\widehat{\mathbf{S}}_p[h]\mathrm{e}^{2\pi \mathrm{i}\omega\varepsilon} + \widehat{\mathrm{n}}_\varepsilon[h]}{\widehat{\mathbf{S}}_p[h] + \widehat{\mathrm{n}}[h]} \\ &= \frac{\widehat{\mathbf{S}}_p[h]\mathrm{e}^{2\pi \mathrm{i}\omega\varepsilon} + \mathrm{O}\left(\sigma\sqrt{p}\right)}{\widehat{\mathbf{S}}_p[h] + \mathrm{O}\left(\sigma\sqrt{p}\right)} \\ &= \frac{\mathrm{e}^{2\pi \mathrm{i}\omega\varepsilon} + \mathrm{O}\left(\frac{\sigma}{a_\omega\sqrt{p}}\right)}{1 + \mathrm{O}\left(\frac{\sigma}{a_\omega\sqrt{p}}\right)} \\ &= \mathrm{e}^{2\pi \mathrm{i}\omega\varepsilon} + \mathrm{O}\left(\frac{\sigma}{a_\omega\sqrt{p}}\right). \end{aligned} \tag{24}$$

Thus the ratio of noisy DFT coefficients agrees with the noiseless ratio up to an error term on the order of $\frac{\sigma}{|a_\omega|\sqrt{p}}$.

Given this estimate for the ratio of noisy DFT coefficients, we can derive bounds for the error in the Lee norm for the phase angle computed via $\mathrm{Arg}(z)$. Let $\mathcal{L}$ be a lattice in $\mathbb{R}$. For any $\theta \in \mathbb{R}$ the *Lee norm associated with the lattice* $\mathcal{L}$ for θ is given by the distance of θ to the lattice $\mathcal{L}$, i.e. $\|\theta\|_{\mathcal{L}} := \min_{k\in\mathcal{L}} |\theta - k|$. Under the Lee norm associated with the lattice $2\pi\mathbb{Z}$ it is well known that for $z, \eta \in \mathbb{C}$ with $|\eta| < |z|$,

$$\begin{aligned}\| \mathrm{Arg}\,(z+\eta) - \mathrm{Arg}(z)\|_{2\pi\mathbb{Z}} &= \| \mathrm{Arg}\left(1 + z^{-1}\eta\right) \|_{2\pi\mathbb{Z}} \\ &\leq |z^{-1}\eta|. \end{aligned} \tag{25}$$

Thus for a non-collision $\omega \in \Omega$ and $h = \omega \pmod p$, the estimates (25) and (24) combined yield

$$\left\| \mathrm{Arg}\left(\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]}\right) - 2\pi\omega\varepsilon \right\|_{2\pi\mathbb{Z}} \leq \mathrm{O}\left(\frac{\sigma}{|a_\omega|\sqrt{p}}\right). \tag{26}$$

When we apply the estimate (10) for ω under the noise model we will end up with an approximation

$$\omega^{\mathrm{n}} := \frac{1}{2\pi\varepsilon}\,\mathrm{Arg}\left(\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]}\right) \tag{27}$$

such that

$$\|\omega^{\mathrm{n}} - \omega\|_{\mathbb{Z}} \leq \mathrm{O}\left(\frac{\sigma}{2\pi\varepsilon|a_\omega|\sqrt{p}}\right). \tag{28}$$

Now if we apply the algorithm developed in [1] the ratio $\frac{\sigma}{\varepsilon a_{\min}\sqrt{p}}$ is critical in determining the sensitivity of our phase estimation (as well as the weight estimation) to noise. Without any modifications to the algorithm it is thus important that we choose the lengths p so that $\frac{\sigma}{\varepsilon a_{\min}\sqrt{p}}$ is within the tolerance.

2.3. Earth mover distance

In the existing literature on sparse Fourier transforms, the ℓ_2 norm is most often used to assess the quality of approximation. There are many reasons for this choice, with the two most convincing perhaps being the completeness of the complex exponentials with respect to the ℓ_2 norm and Parseval's theorem. For certain applications, however, this choice of norm is inappropriate. For example, in wide-band spectral estimation and radar applications, one is interested in identifying a set of frequency intervals containing active Fourier modes. In this case, an estimate $\widetilde{\omega}$ of the true frequency ω with $|\widetilde{\omega} - \omega| \ll N$ is useful, but unless $\widetilde{\omega} = \omega$ the ℓ_2 metric will report an error of size $\mathrm{O}(a_{\min})$. 111

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

For these reasons, we propose measuring the approximation error of sparse Fourier transform problems with the Earth Mover Distance (EMD) [23]. Originally developed in the context of content-based image retrieval, EMD measures the minimum cost that must be paid (with a user-specified cost function) to transform one distribution of points into another. EMD can be calculated efficiently as the solution of a linear program corresponding to a certain flow minimization problem. In addition to allowing for misidentified frequencies, this choice of error metric also has the flexibility to measure the quality of approximation for signals with non-integer frequencies. This important problem has recently been considered in [20], and while not the primary focus of this manuscript, in Section 5.4 we consider a modification of our proposed algorithm which allows for the identification of a single non-integer frequency in the presence of noise.

For our problem, we consider the cost to move a set of estimated Fourier modes and coefficients $\{(\widetilde{\omega}_j, a_{\widetilde{\omega}_j})\}_{j=1}^{\widetilde{k}}$ to the true values $\{(\omega_j, a_{\omega_j})\}_{j=1}^{k}$ under the cost function

$$d_1\big((\omega, a_\omega), (\widetilde{\omega}, a_{\widetilde{\omega}}); N\big) := \frac{|\omega - \widetilde{\omega}|}{N} + |a_\omega - a_{\widetilde{\omega}}|. \tag{29}$$

This choice of cost function strikes a balance between the fidelity of the frequency estimate (as a fraction of the bandwidth) and that of the coefficient estimate. We also consider the "phase-only" cost function

$$d_\omega(\omega_1, \omega_2; N) := \frac{|\omega_1 - \omega_2|}{N}, \tag{30}$$

which provides a measure of how close our frequency estimates are to the true values. We denote the EMD using d_1 by EMD(1) and using d_ω by EMD(ω) in our empirical studies in Section 5 below.

Since these error metrics may be unfamiliar to the reader, we note here that the theoretical best possible EMD(1) error is easy to compute in the special case when the EMD(ω) error is zero (i.e., all frequencies are estimated correctly). In this case, we can combine (23) with (29) above to yield

$$\text{EMD}(1) = \text{O}\left(\frac{k\sigma}{a_{\min}\sqrt{p}}\right). \tag{31}$$

Note in particular that since we measure distances in ℓ_1 the error scales with k, rather than $\sqrt{k}$ as would be the case in ℓ_2. The case when EMD(ω) is non-zero is much more difficult to analyze and is an important question that merits considerable attention. We plan to conduct such a study in future work.

2.4. Random signal model

For the average-case analysis in Section 4.3.3 and the empirical evaluations in Section 5 we consider signals with uniformly random phase over the bandwidth and coefficients chosen uniformly from the complex unit circle. In other words, given k and N, we choose k frequencies ω_j uniformly at random (without replacement) from $[-N/2, N/2) \cap \mathbb{Z}$. The corresponding Fourier coefficients a_j are of the form $\mathrm{e}^{2\pi \mathrm{i}\theta_j}$, where θ_j is drawn uniformly from $[0, 1)$. The signal is then given by

$$S(t) = \sum_{j=1}^{k} a_j \mathrm{e}^{2\pi \mathrm{i}\omega_j t}. \tag{32}$$

This is the standard signal model considered in previous empirical evaluations of sub-linear Fourier algorithms [24,19,14,1]. We note here that we also conducted the empirical evaluations of Section 5 on signals whose Fourier coefficients have varying magnitudes. These results did not differ substantively from those on signals of the form (32), so we omit a detailed discussion.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

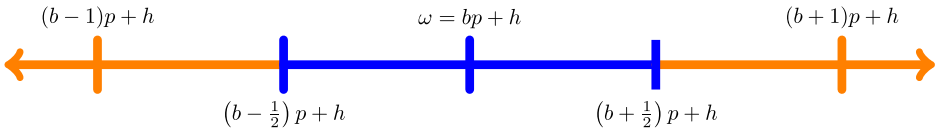


Fig. 1. The rounding procedure is exact as long as the phase estimate $\widetilde{\omega}$ is within $p/2$ of correct multiple of p (blue region in figure). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. Rounding: a minor modification of noiseless algorithm

A simple modification to the noiseless algorithm of [1] for the noisy case is to increase the sample lengths p. By choosing p large enough, the error from noise can be mitigated to be within a given tolerance. The modification can be viewed simply as rounding, and we include it here both as a more direct and simple-to-implement extension as well as for comparison purposes. When the noise level is low, this modification yields reasonably good results.

As in the noiseless case we choose the shift $\varepsilon > 0$ so that $\varepsilon \leq \frac{1}{N}$. In the noiseless case $\varepsilon = \frac{1}{N}$ would be sufficient to avoid wrap-around aliasing in the phase reconstruction. Due to the presence of noise we will need to make ε slightly smaller because of (28). Let us analyze the recovery of a candidate frequency $\omega \in \Omega$ if we simply carry out the same process as in the noiseless environment.

First we choose a length p. Assume that $\omega \in \Omega$ does not collide with any other $\omega' \in \Omega$ modulo p. Let $h = \omega \pmod{p}$. The reconstruction of ω utilizes two factors. First, the location of peaks in the DFT are robust to noise: even with a relatively high noise level we may take $h = \omega \pmod{p}$ to be exact. Second, by (28) the frequency reconstruction from noisy measurements is correct up to an error term of size $\mathrm{O}\left(\frac{\sigma}{\varepsilon a_{\min}\sqrt{p}}\right)$. By combining these two measures we can more reliably estimate ω.

Our proposed modification is to simply round the noisy frequency estimate

$$\widetilde{\omega} = \frac{1}{2\pi\varepsilon} \operatorname{Arg}\left(\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]}\right) \tag{33}$$

to the nearest integer of the form $np + h$. This improved estimate is therefore given by

$$\widetilde{\omega}' = p \cdot \operatorname{round}\left(\frac{\widetilde{\omega} - h}{p}\right) + h, \tag{34}$$

where $\operatorname{round}(x)$ returns the nearest integer to x. For low noise levels this modification will return the true value ω, while for larger noise levels it is possible that $\widetilde{\omega}$ deviates by more than $p/2$ from the true frequency ω. In this case the estimate $\widetilde{\omega}'$ will be wrong by a multiple of p. Larger values of p will reduce the likelihood of an error in frequency estimation. See Fig. 1 for an illustration of this rounding procedure.

To ensure that the estimated frequencies are sufficiently far from the branch cut of $\operatorname{Arg}(z)$ along the negative real axis, we take the shift $\varepsilon \leq \frac{1}{2N}$. The estimated frequencies then satisfy $-N \leq \widetilde{\omega} < N$, while the true frequencies lie in the smaller interval $[-N/2, N/2)$. It is thus extremely unlikely that the deviations due to the noise will push the estimates across the discontinuity.

We saw in the previous section that the error in the phase estimation is on the order of $\sigma(a_{\min}p)^{-1/2}$ when using the reconstruction formula (10). When using the rounding procedure (34), however, we should expect accurate results for a wider range of sample lengths p and noise levels σ. Indeed, note that the rounded frequency estimate $\widetilde{\omega}'$ is *exact* as long as

$$|\widetilde{\omega} - \omega| < \frac{p}{2}. \tag{35}$$

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

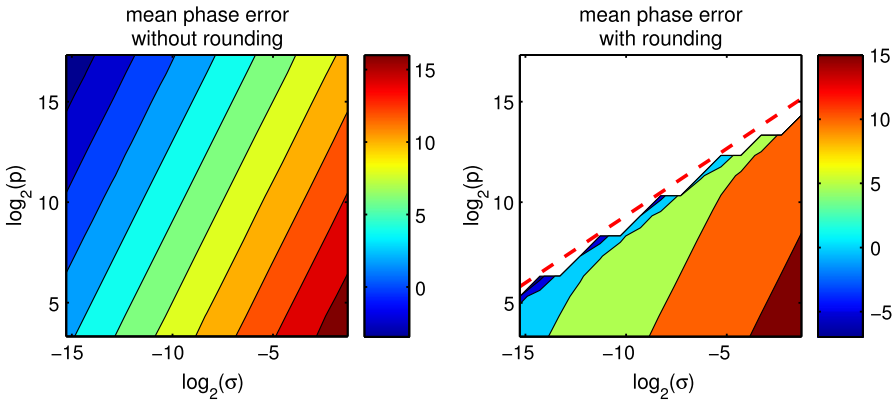


Fig. 2. (Left) Mean phase error (in log scale) for frequency estimation via (10). (Right) Mean phase error (in log scale) for frequency estimation with rounding via (34). The red dashed line marks the transition to exact recovery when $p > (2\sigma/\varepsilon)^{2/3}$.

Recall from Section 2.2 that the error of the frequency estimate $\widetilde{\omega}$ is on the order of $\mathrm{O}(\frac{\sigma}{\varepsilon a_{\min}\sqrt{p}})$. Let us assume that it is bounded by $C\frac{\sigma}{\varepsilon a_{\min}\sqrt{p}}$ for some constant C. Combining this with the requirement (35) we see that the rounded frequency estimate $\widetilde{\omega}'$ will be exact provided

$$C\frac{\sigma}{\varepsilon a_{\min} p^{3/2}} < \frac{1}{2}. \tag{36}$$

It follows that we get exact reconstruction if $p \geq \left(\frac{2C\sigma}{\varepsilon a_{\min}}\right)^{2/3}$.

To illustrate this relationship, we generated 1000 test signals with frequencies chosen uniformly at random from $[-N/2, N/2)$ and set the corresponding coefficient to unity. Thus our test signals for this empirical trial were one-term trigonometric polynomials. For this test we took $N = 2^{22}$, $\varepsilon = \frac{1}{2N}$ and investigated a range of parameters (σ, p). We reconstructed the frequencies in two ways: first, simply using the formula (10), and second by combining this estimate with the rounding procedure (34). In Fig. 2 we plot the average phase error in logarithmic scale as a function of both σ and p, which were varied from 2.5×10^{-5} to 0.4096 and from 10 to 163 840, respectively, by powers of two.

In the plot on the left, which corresponds to reconstruction using only (10), we can clearly see the contours of constant phase error obeying the relationship $\log_2(p) = 2\log_2(\sigma) + \alpha$ for various α. This confirms our analytic estimate from Section 2.2 that the phase error is proportional to $\sigma/(a_{\min}\sqrt{p})$. In the plot on the right, which corresponds to the improved reconstruction using (34), we can see that for large values of σ and small values of p the same relationship holds. However, for smaller σ and larger p we see an abrupt transition to exact reconstruction (the white area in the upper-left). The boundary of this region (red dashed line) follows the relationship $\log_2(p) = \frac{2}{3}\log_2(\sigma) + 16$, corresponding to $C = 1$ in (36) above. This confirms that for small enough values of the ratio $\frac{\sigma}{\varepsilon a_{\min} p^{3/2}}$ the rounding procedure is exact.

3.1. Algorithm

Our first algorithm for noisy signals is only a slight modification of the noiseless algorithm presented in [1, Algorithm 1]. Considering (36), we change the lower bound

$$p > c_1 k \tag{37}$$

to

$$p > \max\left\{c_1 k, c_2 \left(\frac{\sigma}{\varepsilon a_{\min}}\right)^{2/3}\right\}, \tag{38}$$

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

where c_1, c_2 are constants. In this way we ensure that the choice of p is always large enough to isolate most of the k frequencies on average as well as being large enough to ensure that the rounding procedure (34) is exact. In all of our experiments in Section 5 below we took $c_2 = 4$.

4. A multiscale algorithm

In Section 3 we saw that increasing p sufficed to ensure that the rounding procedure was exact. While this gives good results in terms of accuracy, the increased runtime associated with larger noise levels is undesirable. The main contribution of this paper is a multiscale algorithm for recovering the frequency set Ω of the signal $S(t)$. This algorithm achieves similar accuracy while providing an improvement of several orders of magnitude in computational efficiency.

The key feature of this multiscale algorithm is the employment of multiple shifts ε_j, which enable us to improve the accuracy of the phase estimations progressively without the need to significantly increase the sample length p. As we will see, taking successively larger shifts enables a form of error-correction in our frequency estimates at finer and finer scales, in essence "zooming in" on the true frequencies in a multiscale fashion. The idea of progressively learning finer scale approximations to significant frequencies has appeared in prior works addressing the sparse Fourier transform problem, including [7–9,13,18,16,17], but the connection to β-encoders is to the best of our knowledge novel.

In Section 4.1 we give some background on our multiscale method and introduce the main idea of our algorithm. In Section 4.2 we prove that our multiscale approximations are accurate estimates of the true frequencies, and in Section 4.3 we describe the basic multiscale algorithm, discuss several implementation details, and present our main Theorem (4.5).

4.1. Multiscale frequency estimation

The main idea for the multiscale algorithm is that a value can be estimated with high precision with an inaccurate (coarse) estimator applied progressively at different scales, much like in analog-to-digital conversion where a signal value can be estimated with very high precision by the very coarse binary quantization. In our sparse Fourier recovery algorithm, the coarse estimator is the approximation formula given by (26)

$$\varepsilon\omega =_{\mathbb{Z}} \frac{1}{2\pi} \operatorname{Arg}\left(\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]}\right), \tag{39}$$

where $=_{\mathbb{Z}}$ is measured by the Lee norm $\|\cdot\|_{\mathbb{Z}}$.

For simplicity let us assume for the moment that our signal contains a single frequency ω with non-zero Fourier coefficient. For a fixed p, let $\widetilde{\omega}$ be our estimate for ω using the rounding procedure from Section 3 with shift $\varepsilon_0 \leq \frac{1}{N}$. Then we have

$$\widetilde{\omega} = \omega \pmod{p}, \tag{40}$$

although in general $\widetilde{\omega}$ may differ from ω by a multiple of p.

Suppose now that we repeat the computation of $\widetilde{\omega}$ using a larger shift $\varepsilon_1 > \varepsilon_0$; that is, we sample our signal at time points $\frac{j}{p} + \varepsilon_1$, take the FFT, and compute

$$b_1 = \frac{1}{2\pi} \operatorname{Arg}\left(\frac{\widehat{\mathbf{S}}^{\mathrm{n}}_{p,\varepsilon_1}[h]}{\widehat{\mathbf{S}}^{\mathrm{n}}_{p}[h]}\right) \tag{41}$$

(note that we do not divide by ε_1 here). Since in general $\varepsilon_1 > \frac{1}{N}$, we cannot take b_1/ε_1 as an estimate for ω, although it still holds that

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

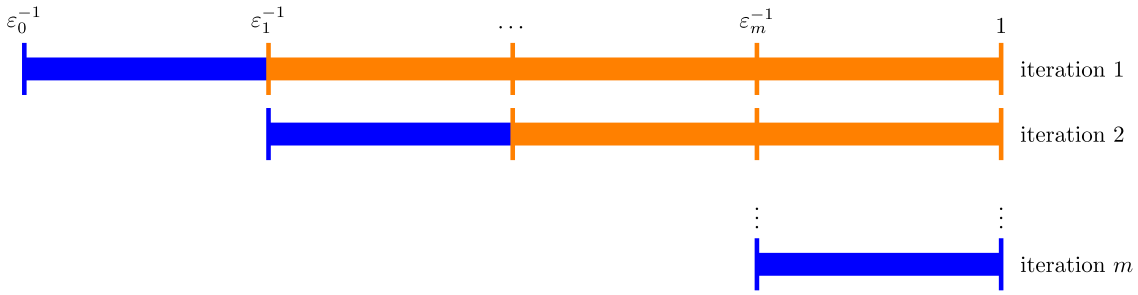


Fig. 3. Diagram of the multiscale frequency estimation procedure, with a candidate frequency pictured as a string of digits, from most significant on the left to least significant on the right. In this figure, blue regions represent correct digits learned by the algorithm, and orange regions represent digits where errors are likely. In the first iteration, the most significant bits are learned using shift ε_0^{-1}. Subsequent iterations give corrections at finer scales $\varepsilon_1^{-1}, \ldots, \varepsilon_m^{-1}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$b_1 \approx \varepsilon_1 \omega \pmod{[-\tfrac{1}{2}, \tfrac{1}{2})}, \tag{42}$$

where $x \pmod{[-\frac{1}{2}, \frac{1}{2})}$ is the unique value y in $[-\frac{1}{2}, \frac{1}{2})$ such that $x \equiv y \pmod 1$. We can use this fact to estimate the error $\omega - \widetilde{\omega}$ as follows. Note that

$$\begin{aligned} \varepsilon_1(\omega - \widetilde{\omega}) &= \varepsilon_1 \omega - \varepsilon_1 \widetilde{\omega} \\ &\approx (b_1 - \varepsilon_1 \widetilde{\omega}) \pmod{[-\tfrac{1}{2}, \tfrac{1}{2})}, \end{aligned} \tag{43}$$

so that

$$\omega - \widetilde{\omega} \approx \frac{(b_1 - \varepsilon_1 \widetilde{\omega}) \pmod{[-\frac{1}{2}, \frac{1}{2})}}{\varepsilon_1}. \tag{44}$$

This estimate of the error is not exact, since there is still noise that can perturb the calculated value b_1 from the true value $\varepsilon_1 \omega \pmod{[-\frac{1}{2}, \frac{1}{2})}$. However, analogously to (28) we have

$$(\omega - \widetilde{\omega}) - \frac{(b_1 - \varepsilon_1 \widetilde{\omega}) \pmod{[-\frac{1}{2}, \frac{1}{2})}}{\varepsilon_1} = \mathrm{O}\left(\frac{\sigma}{\varepsilon_1 a_{\min} \sqrt{p}}\right), \tag{45}$$

which immediately implies that the updated estimate satisfies

$$\omega - \left(\widetilde{\omega} + \frac{(b_1 - \varepsilon_1 \widetilde{\omega}) \pmod{[-\frac{1}{2}, \frac{1}{2})}}{\varepsilon_1}\right) = \mathrm{O}\left(\frac{\sigma}{\varepsilon_1 a_{\min} \sqrt{p}}\right). \tag{46}$$

Since $\varepsilon_1 > \varepsilon_0$, adding the correction term (44) to our previous estimate $\widetilde{\omega}$ will give a finer approximation to the true frequency ω. By iterating this error correction process with progressively larger shifts ε_j, we obtain an algorithm which adaptively corrects for the error in a multiscale fashion. See Fig. 3 for a diagram of the multiscale estimation procedure. In the next section we provide a detailed analysis of this multiscale approximation scheme, and prove that the frequency estimates it produces are accurate.

4.2. Analysis of multiscale approximations

We begin with a technical lemma relating arithmetic in the Lee norm $\|\cdot\|_{\mathbb{Z}}$ to that on the interval $[-\frac{1}{2}, \frac{1}{2})$. It will be used repeatedly in the sequel.

Lemma 4.1. *Let $\delta > 0$ and $x \in [-\frac{1}{2} + \delta, \frac{1}{2} - \delta]$. Assume that $\|x - b\|_{\mathbb{Z}} < \delta$ and $b \in [-\frac{1}{2}, \frac{1}{2})$. Then $|x - b| < \delta$.*

Proof. Let $r = \|x - b\|_{\mathbb{Z}}$. Then $x - b = \pm r + k$ for some $k \in \mathbb{Z}$. If $k = 0$ we have

$$|x - b| = \|x - b\|_{\mathbb{Z}} < \delta \tag{47}$$

by hypothesis, so the claim holds. Now assume $k \neq 0$. Note that

$$|x - b| \leq |x| + |b| \leq 1 - \delta \tag{48}$$

by the triangle inequality and the assumptions on x and b. At the same time, we have

$$|\pm r + k| \geq 1 - r > 1 - \delta. \tag{49}$$

This is a contradiction, since (48) and (49) cannot hold simultaneously. Thus we must in fact have $k = 0$, and the claim holds. □

The following theorem formalizes the multiscale frequency estimation procedure which was introduced in the previous section.

Theorem 4.2. *Let $\omega \in [-\frac{N}{2}, \frac{N}{2})$. Let $0 < \varepsilon_0 < \varepsilon_1 < \cdots < \varepsilon_m$ and $b_0, b_1, \ldots, b_m \in \mathbb{R}$ such that*

$$\|\varepsilon_j \omega - b_j\|_{\mathbb{Z}} < \delta, \qquad 0 \leq j \leq m \tag{50}$$

where $0 < \delta \leq \frac{1}{4}$. Assume that $\varepsilon_0 \leq \frac{1-2\delta}{N}$ and $\beta_j := \varepsilon_j/\varepsilon_{j-1} \leq (1 - 2\delta)/(2\delta)$. Then there exist $c_0, c_1, \ldots, c_m \in \mathbb{R}$, each computable from $\{\varepsilon_j\}$ and $\{b_j\}$, such that

$$|\widetilde{\omega} - \omega| \leq \frac{\delta}{\varepsilon_0} \prod_{j=1}^{m} \beta_j^{-1}, \qquad \text{where} \qquad \widetilde{\omega} := \sum_{j=0}^{m} \frac{c_j}{\varepsilon_j}. \tag{51}$$

Proof. Denote $\omega_0 := \omega$. We first note that

$$|\varepsilon_0 \omega_0| \leq \varepsilon_0 \frac{N}{2} \leq \frac{1}{2} - \delta, \tag{52}$$

where the second inequality follows from the assumptions of the theorem. Let $c_0 = b_0 \pmod{[-\frac{1}{2}, \frac{1}{2})}$, so that $|\varepsilon_0 \omega_0 - c_0| < \delta$ by Lemma 4.1. Let $\lambda_0 = c_0/\varepsilon_0$, which represents a coarse estimate of ω_0 with the error bound

$$|\lambda_0 - \omega_0| < \delta/\varepsilon_0. \tag{53}$$

Next, let $\omega_1 = \omega_0 - \lambda_0$. By the above $|\omega_1| < \delta/\varepsilon_0$ and

$$|\varepsilon_1 \omega_1| < \frac{\varepsilon_1 \delta}{\varepsilon_0} = \beta_1 \delta \leq \frac{1}{2} - \delta. \tag{54}$$

We then have

$$\|\varepsilon_1 \omega - b_1\|_{\mathbb{Z}} = \|\varepsilon_1 \omega_1 - (b_1 - \varepsilon_1 \lambda_0)\|_{\mathbb{Z}} < \delta. \tag{55}$$

Set $c_1 = b_1 - \varepsilon_1 \lambda_0 \pmod{[-\frac{1}{2}, \frac{1}{2})}$. It follows from Lemma 4.1 again that $|\varepsilon_1 \omega_1 - c_1| < \delta$. We set $\lambda_1 = c_1/\varepsilon_1$.

We can recursively define c_j, λ_j and ω_j for all $1 \leq j \leq m$. In general we define $\omega_j := \omega_{j-1} - \lambda_{j-1}$. This leads to

$$|\varepsilon_j \omega_j| < \frac{\varepsilon_j \delta}{\varepsilon_{j-1}} = \beta_j \delta \leq \frac{1}{2} - \delta. \tag{56}$$

Set

$$c_j = (b_j - \varepsilon_j \lambda_{j-1}) \pmod{[-\tfrac{1}{2}, \tfrac{1}{2})}, \tag{57}$$

which yields

$$\|\varepsilon_j \omega_j - c_j\|_{\mathbb{Z}} < \delta. \tag{58}$$

Lemma 4.1 now gives $|\varepsilon_j \omega_j - c_j| < \delta$. Set $\lambda_j = c_j/\varepsilon_j$.

Finally denote $\omega_{m+1} = \omega_m - \lambda_m$. It is straightforward now to verify that

$$\begin{aligned} \omega = \omega_0 &= \sum_{j=0}^{m} \lambda_j + \omega_{m+1} \\ &= \sum_{j=0}^{m} \frac{c_j}{\varepsilon_j} + \omega_{m+1}. \end{aligned} \tag{59}$$

Furthermore, by construction $\omega_{m+1} = \omega_m - \lambda_m$, which has $|\omega_{m+1}| \leq \delta/\varepsilon_m$. By hypothesis $\varepsilon_m = \varepsilon_0 \prod_{j=1}^{m} \beta_j$, yielding

$$|\omega_{m+1}| \leq \frac{\delta}{\varepsilon_0} \prod_{j=1}^{m} \beta_j^{-1} \tag{60}$$

and completing the proof. □

Remark 4.1. From the proof of Theorem 4.2 the values c_j and $\widetilde{\omega}$ are explicitly computable through the recursive formula $\omega_0 = \omega$, $c_0 = b_0 \pmod{[-\frac{1}{2}, \frac{1}{2})}$, $\lambda_0 = c_0/\varepsilon_0$ and

$$\begin{cases} \omega_j &= \omega_{j-1} - \lambda_{j-1} \\ c_j &= (b_j - \varepsilon_j \lambda_{j-1}) \pmod{[-\frac{1}{2}, \frac{1}{2})} \\ \lambda_j &= c_j/\varepsilon_j \end{cases} \tag{61}$$

for $1 \leq j \leq m$. Equivalently, we can write the updated frequency estimates along the lines of (46) as

$$\begin{aligned} \widetilde{\omega}_0 &= b_0/\varepsilon_0 \\ \widetilde{\omega}_{n+1} &= \widetilde{\omega}_n + \frac{(b_n - \varepsilon_n \widetilde{\omega}_n) \pmod{[-\frac{1}{2}, \frac{1}{2})}}{\varepsilon_n}. \end{aligned} \tag{62}$$

Corollary 4.3. *Assume that in the above theorem we have $\beta_j = \beta$ where $\beta \leq (1 - 2\delta)/(2\delta)$, i.e. $\varepsilon_j = \beta^j \varepsilon_0$ for all j. Let $p > 0$ and $m \geq \left\lfloor \log_\beta \frac{2\delta}{p\varepsilon_0} \right\rfloor + 1$. Then*

$$|\widetilde{\omega} - \omega| \leq \frac{\delta}{\varepsilon_0} \beta^{-m} < \frac{p}{2}. \tag{63}$$

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

Proof. This is a straightforward corollary. By Theorem 4.2 we have

$$|\widetilde{\omega} - \omega| \leq \frac{\delta}{\varepsilon_0} \prod_{j=1}^{m} \beta_j^{-1} = \frac{\delta}{\varepsilon_0} \beta^{-m}. \tag{64}$$

It is easy to check that $m = \left\lfloor \log_\beta \frac{2\delta}{p\varepsilon_0} \right\rfloor + 1$ is the smallest integer such that $\frac{\delta}{\varepsilon_0}\beta^{-m} < \frac{p}{2}$. □

Note that as mentioned in Section 3, even with noise the value $\omega \pmod p$ can be accurately computed very reliably. Thus if the difference $|\omega - \widetilde{\omega}|$ is smaller than $\frac{p}{2}$ then ω can be recovered exactly by taking the closest integer to $\widetilde{\omega}$ with the same residue modulo p.

In numerical tests we choose uniform $\beta_j = \beta$. While making β as large as it can be for a given error estimate δ will undoubtedly reduce the computational cost, there is nevertheless a good reason that we should not be too "greedy" and be more conservative by choosing a smaller $\beta > 1$. The reason is that given the random nature of the noise the error bound δ is only in the average sense. To minimize reconstruction errors we should try to provide as much latitude as possible for the uncertainties associated with the error estimate δ. Hence it is useful to ask how much latitude does one get for given choices of ε_0 and β.

Theorem 4.4. *Let $\omega \in [-\frac{N}{2}, \frac{N}{2})$, $\varepsilon_0 > 0$ and $\beta > 1$. Set $\varepsilon_j = \beta^j \varepsilon_0$ for $1 \leq j \leq m$. Assume that we have $b_0, b_1, \ldots, b_m \in \mathbb{R}$ such that*

$$\|\varepsilon_j \omega - b_j\|_{\mathbb{Z}} < \delta, \qquad 1 \leq j \leq m \tag{65}$$

where

$$\delta = \min\Big(\frac{1 - \varepsilon_0 N}{2}, \frac{1}{2\beta + 2}\Big). \tag{66}$$

Then the estimate $\widetilde{\omega}$ of ω given by $\widetilde{\omega} := \sum_{j=0}^{m} \frac{c_j}{\varepsilon_j}$ satisfies

$$|\widetilde{\omega} - \omega| \leq \frac{\delta}{\varepsilon_0} \beta^{-m}, \tag{67}$$

where c_j are given in (61).

Proof. The proof is straightforward. Note that Theorem 4.2 holds under the conditions $\varepsilon_0 \leq \frac{1-2\delta}{N}$ and $\beta_j \leq \frac{1-2\delta}{2\delta}$. These two conditions are equivalent to the condition $\delta \leq \min\big(\frac{1-\varepsilon_0 N}{2}, \frac{1}{2\beta+2}\big)$. Clearly, $\delta = \min\big(\frac{1-\varepsilon_0 N}{2}, \frac{1}{2\beta+2}\big)$ is the largest admissible value for δ. □

4.3. Algorithm

In this section we provide some details of our implementation of the multiscale frequency estimation procedure described in Section 4.1. In particular, we discuss the choice of various parameters necessary for reconstruction according to Theorem 4.2 as well as changes made to the aliasing detection test from [1] to improve robustness in the presence of noise. We give pseudocode for the iterative frequency estimation procedure in Algorithm 1; the full algorithm is given by replacing lines 6–22 in [1, Algorithm 1] with this procedure. We also present our main Theorem (4.5) stating the correctness and average and worst case runtime and sampling complexity of the multiscale algorithm.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

Algorithm 1 MULTISCALEFREQEST.

Input: $S(t), N, k, \beta, \sigma, a_{\min}, c_\sigma, \eta$
Output: $\{\widetilde{\omega}_\ell\}_{\ell=1}^{\widetilde{k}}$

$p \leftarrow \max\left\{c_1 k, \left(\frac{\beta(\beta+1)a_{\min}c_\sigma\sigma}{\pi}\right)^2\right\}$
$\tau \leftarrow \frac{c_\sigma\sigma}{a_{\min}\sqrt{p}}$, $m \leftarrow 1 + \left\lfloor \log_\beta \frac{N}{p} \right\rfloor$
$\text{vote}_\ell \leftarrow 0$, $\ell = 1, \ldots, k$
$\widehat{\mathbf{S}}_p \leftarrow$ FFT of $\frac{1}{p}$-samples of $S(t)$
for $j = 0$ to m **do**
  $\varepsilon_j \leftarrow \frac{\beta^j}{2N}$
  $\widehat{\mathbf{S}}_{p,\varepsilon_j} \leftarrow$ FFT of ε_j-shifted $\frac{1}{p}$-samples of $S(t)$
  for $\ell = 1$ to k **do**
    $h \leftarrow$ index of ℓth largest peak in $\widehat{\mathbf{S}}_p$
    $r \leftarrow \left| \frac{\left|\widehat{\mathbf{S}}_{p,\varepsilon_j}[h]\right|}{\left|\widehat{\mathbf{S}}_p[h]\right|} - 1 \right|$
    if $r > \tau$ **then**
      $\text{vote}_\ell \leftarrow \text{vote}_\ell + 1$
    end if
    $b_j \leftarrow \frac{1}{2\pi}\,\text{Arg}\left(\frac{\widehat{\mathbf{S}}^{\text{n}}_{p,\varepsilon_j}[h]}{\widehat{\mathbf{S}}^{\text{n}}_p[h]}\right)$
    if $j = 0$ **then**
      $\widetilde{\omega}_\ell \leftarrow b_j/\varepsilon_j$
    else
      $\widetilde{\omega}_\ell \leftarrow \widetilde{\omega}_\ell + (b_j - \varepsilon_j\widetilde{\omega}_\ell) \pmod{[-\frac{1}{2}, \frac{1}{2})}/\varepsilon_j$
    end if
    if $j = m$ **then**
      $\widetilde{\omega}_\ell \leftarrow p \cdot \text{round}\left(\frac{\widetilde{\omega}_\ell - h}{p}\right) + h$
    end if
  end for
end for
return $\widetilde{\omega}_\ell$ with $\text{vote}_\ell \le \eta(m+1)$

4.3.1. Choice of p

It remains to determine the choice of sampling length p, given the parameter β and the noise level σ. Recall from the proof of Theorem 4.2 that the estimated frequency $\widetilde{\omega}$ is given by the sum $\sum_{j=1}^m \lambda_j$, where $\lambda_j = c_j/\varepsilon_j$. Moreover, the difference between successive frequency approximations is given in terms of λ_j as

$$\omega_j := \omega_{j-1} - \lambda_{j-1} \quad \Longrightarrow \quad \lambda_j = \omega_j - \omega_{j+1}. \tag{68}$$

Thus we can decompose the error of approximation at stage $j+1$ as

$$\begin{aligned} |\omega - \omega_{j+1}| &= |(\omega_j - \omega_{j+1}) - (\omega_j - \omega)| \\ &= |\lambda_j - (\omega_j - \omega)|. \end{aligned} \tag{69}$$

By Theorem 4.2 the left-hand side of (69) satisfies

$$|\omega - \omega_{j+1}| < \frac{\delta}{\varepsilon_{j+1}}, \tag{70}$$

while analogously to (28) the right-hand side of (69) satisfies

$$|\lambda_j - (\omega_j - \omega)| \le \mathrm{O}\left(\frac{\sigma}{2\pi\varepsilon_j a_{\min}\sqrt{p}}\right). \tag{71}$$

Denoting by c_σ the constant in the right-hand side above and equating the two upper bounds gives

$$\frac{2\pi\delta\sqrt{p}}{a_{\min}c_\sigma\sigma} = \frac{\varepsilon_{j+1}}{\varepsilon_j} =: \beta. \tag{72}$$

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

Under the assumptions of Theorem 4.4, we have

$$\delta = \min\left(\frac{1-\varepsilon_0 N}{2}, \frac{1}{2\beta+2}\right). \tag{73}$$

Since we take $\varepsilon_0 = \frac{1}{2N}$ and fix $\beta > 1$, the latter term is necessarily the smaller. Plugging this into (72) above and rearranging to solve for p gives

$$p = \left(\frac{\beta(\beta+1)a_{\min}c_\sigma\sigma}{\pi}\right)^2. \tag{74}$$

As in the rounding algorithm, we require in addition that $p > c_1 k$, so the sample lengths for the multiscale algorithm are chosen to satisfy

$$p > \max\left\{c_1 k, \left(\frac{\beta(\beta+1)a_{\min}c_\sigma\sigma}{\pi}\right)^2\right\}. \tag{75}$$

4.3.2. Robust aliasing test

As noted in Section 2.1, our frequency estimation procedure works only for non-collision ω. In [1] two tests were given to determine whether a collision had occurred at a candidate frequency. In the implementation of that algorithm in the noiseless setting, requiring the ratio (11) to be within some threshold of unity sufficed to detect collisions. In the setting of the current paper, where the samples are corrupted with noise, we resort to the second of the tests given in [1], which examines the ratios (11) for several values of ε. For $0 \le j \le m$ we compute the ratio (11) and compare it with a threshold τ. We count the number of times the ratio exceeds τ and reject those frequencies which fail more than an η fraction of the tests. Since we expect fluctuations in this ratio due to noise of order $\frac{\sigma}{a_{\min}\sqrt{p}}$ we set τ to be a small constant multiple of this quantity.

4.3.3. Number of iterations

Recall from Corollary 4.3 that, for constant $\beta_j = \beta$, $m = \left\lfloor \log_\beta \frac{2\delta}{p\varepsilon_0} \right\rfloor + 1$ shifts suffices to ensure that the estimated frequency satisfies $|\widetilde{\omega} - \omega| < \frac{p}{2}$. As in Section 3 we take $\varepsilon_0 = \frac{1}{2N}$ to avoid the branch cut of $\mathrm{Arg}(z)$. Assume that the first term in (75) is the larger of the two, so that $p = \mathrm{O}(k)$. Then after $\mathrm{O}(\log(N/k))$ iterations, by rounding the approximate frequency $\widetilde{\omega}$ to the closest integer of the form $np + h$, where $h = \omega \pmod p$ is known from the location of the peak in $\widehat{\mathbf{S}}^{\mathrm{n}}_p$, we will recover the true frequency ω. With the results of [1, Theorems 3–4] this immediately implies the following

Theorem 4.5. *Let $S^{\mathrm{n}}(t) = S(t) + \mathrm{n}(t)$, where $\widehat{S}(\omega)$ is k-sparse with integral frequencies satisfying $\omega \in \Omega \subset [-N/2, N/2)$ and n is complex i.i.d. Gaussian noise of variance σ^2. Moreover, suppose that $k > C(\beta(\beta+1)a_{\min}\sigma)^2$ for a constant C. There is a deterministic algorithm that, given N, k, β and access to $S^{\mathrm{n}}(t)$ returns a list of k pairs $(\widehat{\omega}, \widehat{a}_{\widehat{\omega}})$ such that (i) each $\widehat{\omega} \in \Omega$ and (ii) for each $\widehat{\omega}$, there is an $\omega \in \Omega$ such that $|a_\omega - \widehat{a}_{\widehat{\omega}}| \le C'\sigma/\sqrt{k}$.*

The average-case runtime and sampling complexity are

$$\mathrm{O}(k\log(k)\log(N/k)) \quad \textit{and} \quad \mathrm{O}(k\log(N/k)),$$

respectively, over the class of signals in Section 2.4. The worst-case runtime and sampling complexities are

$$\mathrm{O}(k^2\log(k)\log(N/k)) \quad \textit{and} \quad \mathrm{O}(k^2\log(N/k)),$$

respectively.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

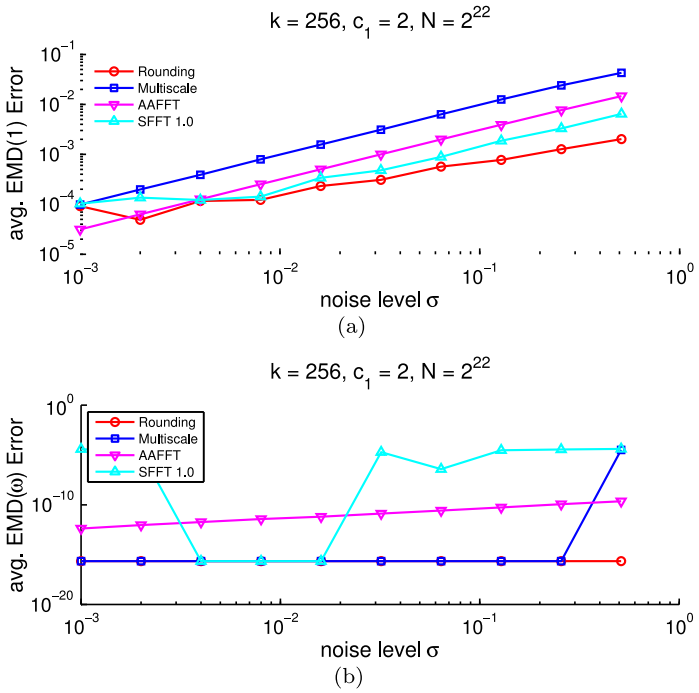


Fig. 4. (a) Average EMD(1) error of the algorithms as a function of noise level σ. (b) Average EMD(ω) error as a function of noise level σ. Due to the log scale on the y axis, all EMD(ω) values have been shifted up by 10^{-16} for clarity.

Proof. Replacing lines 6–22 of [1, Algorithm 1] with the multiscale frequency estimation procedure of Algorithm 1 yields an algorithm with the stated runtime and sampling complexities. The termination criteria of [1, Algorithm 1] ensures that k frequencies are returned, and the previous analysis in this section ensures that the returned frequencies are correct. The coefficient estimates $\widehat{a}_{\widehat{\omega}}$ satisfy (23), which together with the assumption on k yields the claim. □

5. Empirical evaluation

In this section we describe the results of an empirical evaluation of the algorithms of Sections 3 and 4. We focus on two aspects of the algorithms' performance: accuracy as measured in the EMD(1) and EMD(ω) metrics (cf. Section 2.3), and runtime as a function of both the sparsity k and the noise level σ. In all of the experiments reported below, we report averages over 100 random test signals generated according to the prescription in Section 2.4. The bandwidth for these tests was fixed at $N = 2^{22}$.

All experiments were conducted in C++ on a Linux machine with four Intel Xeon X5355 dual-core processors at 2.66 GHz and 64 Gb of RAM. The GNU compiler was used with optimization flag -O3. For the multiscale algorithm, it was determined after extensive testing that the choice of parameters $c_1 = 2$, $c_\sigma = 6$, $\eta = \frac{1}{4}$, $\beta = 2.5$ gave a satisfactory balance between runtime and accuracy. All FFTs were computed using FFTW3 [3]. For comparison, we also present the results of the same trials for two alternative sparse Fourier algorithms: sFFT 1.0 [14] and AAFFT [24].

5.1. Accuracy

In Fig. 4 (a) we plot the average EMD(1) error of the algorithms as a function of the noise level σ. For the rounding algorithm, the EMD(1) error increases as $\sigma^{2/3}$, while for the other three it increases linearly. In all cases the EMD(1) error is dominated by the coefficient error. The coefficient estimates in all four algorithms are given by an empirical average of the samples, and so the accuracy is determined by the number of samples taken. This explains both the scaling of the error of our rounding algorithm (recall from Section 3

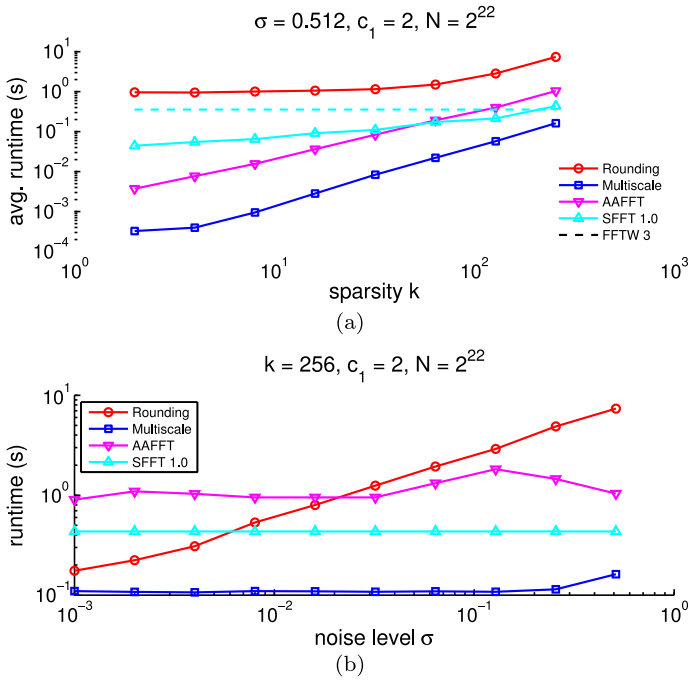


Fig. 5. (a) Average runtime vs. sparsity k for the algorithms tested. (b) Average runtime vs. noise level σ.

that $p > \left(\frac{\sigma}{\varepsilon a_{\min}}\right)^{2/3}$), as well as the larger EMD(1) error of our multiscale algorithm, which performs well even with c_1 as small as two. The multiscale error correction allows us to take much coarser sampling rates to achieve a tolerable error. As we show in the next subsection, these coarser sampling rates lead to much improved runtime.

In order to assess the accuracy of the frequency lists returned by each of the four algorithms, in Fig. 4 (b) we plot the average EMD(ω) error as a function of the noise level. The EMD(ω) error was zero for all trials of the rounding algorithm, as expected due to the choice of p. Moreover, for all but the highest noise level, the EMD(ω) error of the multiscale algorithm was zero in all trials. For most values of σ, the EMD(ω) error of sFFT 1.0 was non-zero, indicating that even at low to moderate noise levels, erroneous frequencies are returned. The EMD(ω) error of AAFFT was always less than $1/N$, indicating that true frequencies were recovered in all cases; the non-zero values are numerical artifacts.

5.2. Runtime

In Fig. 5 (a) we plot the average runtime of the algorithms as a function of the sparsity k for a fixed value of the noise level $\sigma = 0.512$ and the parameter $c_1 = 2$. As a reference for runtime comparisons, we also plot the time taken by FFTW3 on the same machine. For the rounding algorithm, we see that there is no dependence on k until $k = 64$; this is a consequence of the requirement (38) on the choice of sampling rate. Thus at this noise level our modified algorithm is slightly slower than a highly optimized FFT implementation. The average runtime of our multiscale algorithm scales slightly superlinearly with k, which is expected given the runtime bound $O(k \log(k) \log(N/k))$ of Section 4.3.3. Moreover, we note that for all levels of sparsity tested, the multiscale algorithm outperforms AAFFT, sFFT 1.0, and FFTW3.

In Fig. 5 (b) we plot the average runtime of the algorithms as a function of the noise level σ for a fixed value of the sparsity $k = 256$. For the rounding algorithm we can see the approximate dependence of the runtime on $\sigma^{2/3}$, as dictated by the choice of p in (38). For the multiscale algorithm, there is no dependence on σ until the very noisy case $\sigma = 0.512$.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

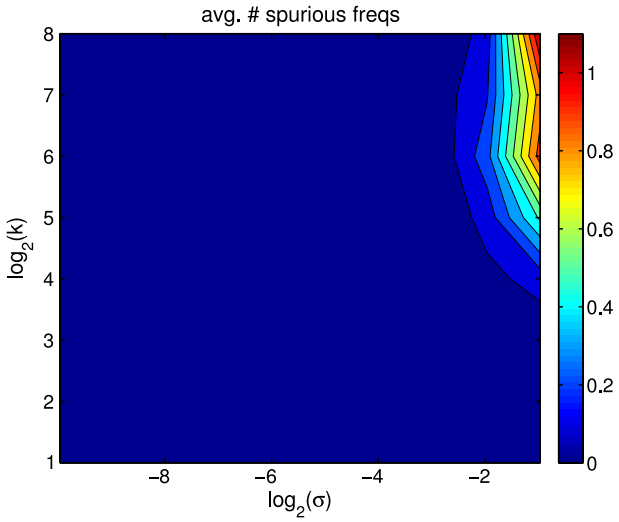


Fig. 6. Average number of spurious frequencies inserted and deleted by the multiscale algorithm as a function of k and σ.

5.3. Spurious frequencies

As noted in Section 2, due to noise it is possible that one or more spurious frequencies are introduced into our signal representation. In subsequent iterations, any such spurious frequency will be identified and subtracted from the updated representation. Since this happens with non-zero probability, it is of interest to examine how often such an insertion and deletion occurs. In Fig. 6, we plot the average number of spurious frequencies inserted and deleted by the multiscale algorithm as a function of k and σ. It is clear that the inclusion of spurious frequencies only occurs in the high-noise, high-sparsity regime. Moreover, on average only one such wrong frequency appears in an intermediate representation even in this challenging environment. This indicates that our robust aliasing test of Section 4.3.2 does a very good job at detecting collisions in all but the most extreme circumstances.

5.4. Non-integer frequency estimation

We report here on an experiment to investigate the utility of our multiscale algorithm for the estimation of a single non-integer frequency. While an exhaustive study is beyond the scope of this paper, it is interesting to note that a minor modification of our multiscale frequency estimation algorithm performs quite well in practice. In addition, this setting provides another justification for use of the EMD metric for assessing the quality of the output of our algorithm, since there is no way to compare non-integer frequencies using a discrete ℓ_2 norm. See also [20] for a brief discussion of the output evaluation metric for this problem. The question of estimating multiple non-integer (or off-grid) frequencies in noise is difficult, requiring more robust methods than those described here. Recent work addressing this question from the algorithms and optimization perspectives include [20] and [21], respectively.

In the non-integer frequency case, we modify Algorithm 1 to omit lines 20–22, i.e. we do not round our frequency estimates to the nearest integer with the same remainder as h. Our frequency estimates are thus not necessarily integers, and an empirical evaluation shows that they approximate the true non-integer frequencies quite well, even in the presence of noise. In our empirical evaluation, we set the single non-integer frequency to be $\sqrt{2}u$, where u is uniform on $\left(-\lceil\frac{N}{2\sqrt{2}}\rceil, \lfloor\frac{N}{2\sqrt{2}}\rfloor\right)$, and set the corresponding coefficient to be unity. In Fig. 7 we plot the average EMD(ω) error as a function of the noise level σ, averaged over 100 trials. It is clear from the figure that the EMD(ω) error scales linearly with the noise level, indicating the robustness of our estimation procedure for the single-frequency case. We do not attempt to explain this phenomenon here, and leave a detailed study of this important question as a topic for future work.

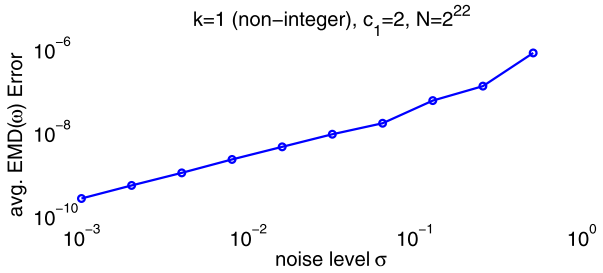


Fig. 7. Average EMD(ω) error vs. noise level σ for a single non-integer frequency.

6. Conclusion

In this paper we gave two extensions of the sparse Fourier algorithm of [1] to handle noisy signals. The first of these was a minor modification of the original algorithm that involved rounding frequency estimates to the nearest integer with the correct residue modulo the inverse sampling rate. We showed that in order for this modification to correctly identify the true frequencies in Gaussian noise of standard deviation σ the sampling rate needed to satisfy $p \geq (a_{\min}\sigma)^{2/3}$. While this resulted in accurate approximations of the Fourier transform in the EMD(1) and EMD(ω) metrics, the sampling rate requirement forced the algorithms to be slow in practice.

The second extension overcame this pitfall by introducing a multiscale approach to frequency estimation inspired by the literature on β-encoders in analog-to-digital conversion. By using samples of the input at multiple geometrically spaced time shifts, our algorithm exhibits a form of error correction in its frequency estimation. This allows the use of much coarser sampling rates than the first modification, which in turn leads to greatly reduced runtimes in our empirical evaluation. The error correction of our multiscale algorithm is similar to that of the β-encoders, and this connection is to the best of our knowledge novel in the sparse Fourier transform context.

Acknowledgments

During the preparation of this manuscript we became aware of related work by Laurent Demanet and his collaborators. We kindly acknowledge their generosity in sharing their work with us. We also acknowledge the helpful comments of the anonymous reviewers. AC was supported in part by AFOSR FA9550-11-1-0281, NSF DMS 1115709, AFOSR FA9550-12-1-0343, USAF FA9550-12-1-0455, SPG-2355-SPG-12, and ORAU HPC LDRD. DL was supported in part by AFOSR FA9550-11-1-0281, NSF DMS 1115709, and NSF DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. YW was supported in part by NSF DMS-08135022 and NSF DMS-1043032.

References

[1] D. Lawlor, Y. Wang, A. Christlieb, Adaptive sub-linear time Fourier algorithms, Adv. Adapt. Data Anal. 5 (1) (2013) 1350003.
[2] I. Daubechies, R. DeVore, C.S. Gunturk, V. Vaishampayan, A/D conversion with imperfect quantizers, IEEE Trans. Inform. Theory 52 (3) (2006) 874–885.
[3] M. Frigo, S.G. Johnson, The design and implementation of FFTW3, Proc. I.E.E.E. 93 (2) (2005) 216–231, special issue on “Program Generation, Optimization, and Platform Adaptation”.
[4] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex Fourier series, Math. Comp. 19 (1965) 297–301.
[5] M. Lin, A.P. Vinod, C.M.S. See, A new flexible filter bank for low complexity spectrum sensing in cognitive radios, J. Signal Process. Syst. 62 (2) (2011) 205–215.
[6] G. Shi, J. Lin, X. Chen, F. Qi, D. Liu, L. Zhang, UWB echo signal detection with ultra-low rate sampling based on compressed sensing, IEEE Trans. Circuits Syst. II., Express Briefs 55 (4) (2008) 379–383.
[7] O. Goldreich, L. Levin, A hard-core predicate for all one-way functions, in: Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing, ACM, 1989, pp. 25–32.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

[8] E. Kushilevitz, Y. Mansour, Learning decision trees using the Fourier spectrum, SIAM J. Comput. 22 (6) (1993) 1331–1348.
[9] Y. Mansour, Randomized interpolation and approximation of sparse polynomials, SIAM J. Comput. 24 (2) (1995) 357–368.
[10] A. Gilbert, S. Guha, P. Indyk, S. Muthukrishnan, M. Strauss, Near-optimal sparse Fourier representations via sampling, in: Symposium on Theory of Computing, ACM, 2002, pp. 152–161.
[11] A. Gilbert, S. Muthukrishnan, M. Strauss, Improved time bounds for near-optimal sparse Fourier representations, in: SPIE Wavelets XI, 2005.
[12] C. Anderson, M.D. Dahleh, Rapid computation of the discrete Fourier transform, SIAM J. Sci. Comput. 17 (4) (1996) 913–919.
[13] A. Akavia, S. Goldwasser, S. Safra, Proving hard-core predicates using list decoding, in: Foundations of Computer Science, FOCS, vol. 44, 2003, pp. 146–159.
[14] H. Hassanieh, P. Indyk, D. Katabi, E. Price, Simple and practical algorithm for sparse Fourier transform, in: Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2012, pp. 1183–1194.
[15] H. Hassanieh, P. Indyk, D. Katabi, E. Price, Nearly optimal sparse Fourier transform, in: Proceedings of the 44th Symposium on Theory of Computing, ACM, 2012, pp. 563–578.
[16] A. Akavia, Deterministic sparse Fourier approximation via fooling arithmetic progressions, in: Conference on Learning Theory, CoLT, 2010, pp. 381–393.
[17] A. Akavia, Deterministic sparse Fourier approximation via approximating arithmetic progressions, IEEE Trans. Inform. Theory 60 (2) (2014) 1733–1741.
[18] A. Akavia, Solving hidden number problem with one bit oracle and advice, in: Advances in Cryptology, CRYPTO 2009, Springer, 2009, pp. 337–354.
[19] M. Iwen, Combinatorial sublinear-time Fourier algorithms, Found. Comput. Math. 10 (3) (2010) 303–338.
[20] P. Boufounos, V. Cevher, A. Gilbert, Y. Li, M. Strauss, Whats the frequency, Kenneth?: Sublinear Fourier sampling off the grid, in: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, Springer, 2012, pp. 61–72.
[21] G. Tang, B.N. Bhaskar, P. Shah, B. Recht, Compressive sensing off the grid, in: 2012 50th Annual Allerton Conference on Communication, Control, and Computing, Allerton, IEEE, 2012, pp. 778–785.
[22] H. Hassanieh, F. Adib, D. Katabi, P. Indyk, Faster GPS via the sparse Fourier transform, in: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking, ACM, 2012, pp. 353–364.
[23] Y. Rubner, C. Tomasi, L. Guibas, The earth mover's distance as a metric for image retrieval, Int. J. Comput. Vis. 40 (2) (2000) 99–121.
[24] M. Iwen, A. Gilbert, M. Strauss, Empirical evaluation of a sub-linear time sparse DFT algorithm, Commun. Math. Sci. 5 (4) (2007) 981–998.

Please cite this article in press as: A. Christlieb et al., A multiscale sub-linear time Fourier algorithm for noisy data, Appl. Comput. Harmon. Anal. (2015), http://dx.doi.org/10.1016/j.acha.2015.04.002

Pipeline Schwarz Waveform Relaxation

Benjamin Ong[1], Scott High[2], and Felix Kwok[3]

Abstract To leverage the computational capability of modern supercomputers, existing algorithms need to be reformulated in a manner that allows for many concurrent operations. In this paper, we outline a framework that reformulates classical Schwarz waveform relaxation so that successive waveform iterates can be computed in a parallel pipeline fashion after an initial start-up cost. The communication costs for various implementations are discussed, and numerical scaling results are presented.

Key words: Schwarz waveform relaxation, pipeline parallelism, domain decomposition, distributed computing

1 Introduction

Schwarz Waveform Relaxation (SWR) introduced in [2] has been analyzed for a wide range of time-dependent problems, including the parabolic heat equation [7], wave equation and advection-diffusion equations [6, 8], Maxwell's equations [4], and the porous medium equation [9]. In contrast to classical Schwarz iterations, where the time-dependent PDE is discretized in time and domain-decomposition is applied to the sequence of steady-state problems, SWR solves *time-dependent* subproblems; this relaxes synchronization of the sub-problems and provides a means to couple disparate solvers applied to individual sub-problems, for example [10]. SWR has also been shown in [8, 1] to have superlinear convergence for small time windows. This paper outlines a framework that reformulates SWR so that successive waveform iterates can be computed in a pipeline fashion, allowing for increased concurrency and hence, increased scalability for SWR-type algorithms. In §2, we review the SWR algorithm before introducing and comparing several Pipeline Schwarz Waveform Relaxation algorithms (PSWR) in §3. Numerical scaling results for the linear heat equation are presented in §4.

[1] Michigan State University, Institute for Cyber-Enabled Research, e-mail: ongbw@msu.edu ·
[2] Michigan State University, Department of Mathematics, e-mail: highscot@msu.edu ·
[3] Université de Genève, Switzerland, Section de Mathématiques e-mail: felix.kwok@unige.ch

2 Schwarz Waveform Relaxation

Denote the PDE of interest as

$$\begin{aligned} u_t &= \mathscr{L}(t,u), \quad (x,t) \in \Omega \times [0,T] \\ u(x,0) &= f(x), \quad x \in \Omega \\ u(z,t) &= g(z,t), \quad z \in \partial\Omega. \end{aligned} \tag{1}$$

Consider a partitioning of the domain, $\Omega = \cup_i \Omega_i$. The domains in the partition may be overlapping or non-overlapping. Let u_i denote the solution on sub-domain Ω_i. Then, equation (1) can be decomposed into a coupled system of equations,

$$\begin{aligned} (u_i)_t &= \mathscr{L}(t,u_i), \quad (x,t) \in \Omega_i \times [0,T] \\ u_i(x,0) &= f(x), \quad x \in \Omega_i \\ u_i(z,t) &= g(z,t), \quad z \in \partial\Omega_i \cap \partial\Omega, \\ \mathscr{T}_{ij}(u_i(z,t)) &= \mathscr{T}_{ij}(u_j(z,t)), \quad z \in \partial\Omega_i \cap \partial\Omega_j. \end{aligned} \tag{2}$$

where T are transmission operators appropriate to the equation (1). SWR decouples the system of PDEs in equation (2). Let $u_i^{[k]}$ denote the k-th waveform iterate on sub-domain Ω_i. After specifying an initial estimate for the sub-domain solution on the interfaces, $u_i^{[0]}(z,t), z \in \partial\Omega_i \setminus \partial\Omega$, the SWR algorithm iteratively solves PDEs (3) for waveform iterates $k = 1,2,\ldots$ until convergence,

$$\begin{aligned} (u_i^{[k]})_t &= \mathscr{L}(t,u_i^{[k]}), \quad (x,t) \in \Omega_i \times [0,T] \\ u_i^{[k]}(x,0) &= f(x), \quad x \in \Omega_i \\ u_i^{[k]}(z,t) &= g(z,t), \quad z \in \partial\Omega_i \cap \partial\Omega, \\ \mathscr{T}_{ij}(u_i^{[k]}(z,t)) &= \mathscr{T}_{ij}(u_j^{[k-1]}(z,t)), \quad z \in \partial\Omega_i \cap \partial\Omega_j. \end{aligned} \tag{3}$$

A pseudo-code for the algorithm is presented on the next page. Observe that SWR allows for each sub-domain to independently compute time-dependent solutions on their respective sub-domains (lines 9-11) During each waveform iteration, transmission data on each sub-domain is aggregated for the entire computational time interval before boundary data is exchanged between neighboring sub-domains (lines 12-14).

3 Pipeline Schwarz Waveform Relaxation

Using a similar approach described in [3, 12], the relaxation framework can be rewritten so that after initial start-up costs, multiple waveform iterations can be computed in a pipeline-parallel fashion. A graphical example of the PSWR algo-

Schwarz Waveform Relaxation Algorithm

1. `MPI Initialization`
2. `parallel for` $i = 1 \ldots N$ *(Sub-domain)*
3. `for` $t = \Delta t \ldots T$
4. `Guess` $u_i^{[0]}(z,t), \quad z \in \partial\Omega_i \cap \partial\Omega_j$
5. `end`
6. `end`
7. `for` $k = 1 \ldots K$ *(Waveform iteration)*
8. `parallel for` $i = 1 \ldots N$ *(Sub-domain)*
9. `for` $t = \Delta t \ldots T$
10. `Solve for` $u_i^{[k]}(t,x)$
11. `end`
12. `for` $t = \Delta t \ldots T$
13. `Exchange transmission data` $\mathcal{T}(u_i^{[k]}(t,z))$
14. `end`
15. `Check convergence`
16. `end`
17. `end`

rithm for two subdomains is shown in Figure 1. To simplify the presentation, we

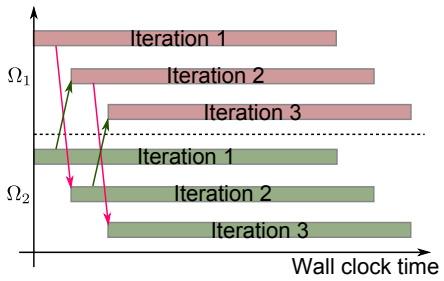


Fig. 1 The proposed PSWR algorithm allows for multiple Schwarz waveform iterations to be simultaneously computed. After an initial start-up cost, multiple iterates are computed in a pipeline fashion.

first present the algorithm for the simplified case where the *same* time discretization is used for all sub-problems (Pipeline Schwarz Waveform Relaxation Algorithm 1).

Several observations should be made about the proposed PSWR algorithm. First, a Schwarz iteration can only proceed if boundary data (i.e. transmission conditions) from the previous iterate are available; this condition (part of the start-up cost before the PSWR algorithm can be run in a pipeline fashion) is checked by the `if` statement in line 12. Secondly, transmission data is exchanged after every time step to facilitate the pipeline parallellism. This added synchronization can be relaxed at the expense of increasing the start-up cost needed to run this algorithm in a pipeline fashion. This

Pipeline Schwarz Waveform Relaxation Algorithm 1

1. `MPI Initialization`
2. `parallel for` $i = 1 \ldots N$ *(Sub-domain)*
3. `for` $t = \Delta t \ldots T$
4. `Guess` $u_i^{[0]}(z,t), \quad z \in \partial\Omega_i \cap \partial\Omega_j$
5. `end`
6. `Set` $t^{[0]} = T$
7. `end`
8. `parallel for` $k = 1 \ldots K$ *(Waveform iteration)*
9. `parallel for` $i = 1 \ldots N$ *(Sub-domain)*
10. `set` $t^{[k]} = \Delta t$
11. `While` $t^{[k]} \leq T$
12. `If` $t^{[k]} < t^{[k-1]}$
13. `Solve for` $u_i^{[k]}(t^{[k]}, x)$
14. `Exchange transmission data` $\mathscr{T}(u_i^{[k]}(t^{[k]}, z))$
15. $t^{[k]} \leftarrow t^{[k]} + \Delta t$
16. `end`
17. `end`
18. `Check convergence`
19. `end`
20. `end`

pipeline parallelism allows for $N \cdot K$ concurrent processes in the PSWR algorithm with efficiency $\frac{N_t}{K+N_t}$ (accounting for start-up costs), where N_t is the number of time steps used to discretize the time domain $[0,T]$, N is the number of sub-domains, K is the number of waveform iterates. This contrasts with the SWR algorithm, which can only utilize N concurrent processes corresponding to the N sub-domains. This increased concurrency in PSWR comes with the overhead of an increased number of messages and synchronization.

For the SWR algorithm, one needs to send $O(K-1)$ message of size $O(N_t)$. If $N \cdot K$ processors are used in a pipeline parallel fashion as described in Pipeline Schwarz Waveform Relaxation Algorithm 1, $O((K-1) \cdot N_t)$ messages of size $O(1)$ are needed. More generally, if $N \cdot p$ processors are used in the PSWR algorithm, where $p < K$ is a multiple of K, then $O((p-1)/p \cdot K \cdot N_t)$ messages of size $O(1)$, and $O(K/p-1)$ messages of size $O(N_t)$, are needed. We note that the PSWR algorithm can also be implemented using a framework the naturally reduces the number of messages in a system. Assuming a heterogeneous computing platform (where each socket has multiple cores), one can use the MPI-3 framework [11] or the OpenMP protocol in the outer "parallel for" statement in line 8, to aggregate transmission data from line 14 naturally before exchanging transmission data with neighboring nodes. Alternatively, because nodes working on waveform iterate k only need to communicate with waveform iterates $k-1$, the PSWR algorithm allows for a natural grouping of nodes so that one can (in principle) use multiple overlapping communicators to leverage data/network-topology and software defined networking advances [5] to add scalability.

Generalizations to allow for disparate time discretizations in each sub-problem are possible. We list the algorithm without implementation. Unlike PSWR Algorithm 1, it is not possible to keep the "pipe" full, i.e. domain i might necessarily need to wait for it's neighbouring domains to provide boundary data. Additionally, solving for $u_i^{[k]}(t_i^{[k]},x)$ in line 14 requires an interpolation algorithm to correctly obtain the correct transmission condition to be used in the solution of (3). Lastly, an implementation decision has to be made on how to collect and store the data from neighboring domains before the interpolation is used to obtain the transmission conditions for an update in line 14.

Pipeline Schwarz Waveform Relaxation Algorithm 2

```
1. MPI Initialization
2. parallel for  i = ...1..N (Sub-domain)
3.     for t_i = Δt_i ... T
4.         Guess u_i^[0](z,t),   z ∈ ∂Ω_i ∩ ∂Ω_j
5.     end
6.     Set t_i^[0] = T
7. end
8. parallel for  k = 1...K (Waveform iteration)
9.     parallel for  i = 1...N (Sub-domain)
10.        initialize Δt_i^[k]
11.        set t_i^[k] = Δt_i^[k]
12.        While t_i^[k] ≤ T
13.            If  t_i^[k] < t_j^[k-1] for all neighbors j
14.                Solve for  u_i^[k](t_i^[k],x)
15.                Send transmission data 𝒯(u_i^[k](t_i^[k],z)) to neighbor nodes
16.                t_i^[k] ← t_i^[k] + Δt_i^[k]
17.            end
18.        end
19.        Check convergence
20.    end
21. end
```

4 Numerical Experiments

We present results from scaling studies, which vary the number of computational cores used to compute the PSWR algorithm while keeping total discretized problem size constant. The diffusion equation $u_t = k(u_{xx} + u_{yy})$ is solved in $\mathbb{R}^2$ using a centered five point finite-difference approximation in space, and a backward Euler time integrator. In our first scaling study, 400x400 grid points are decomposed into 4x4 non-overlapping domains for 400 total time steps. Optimized robin transmission

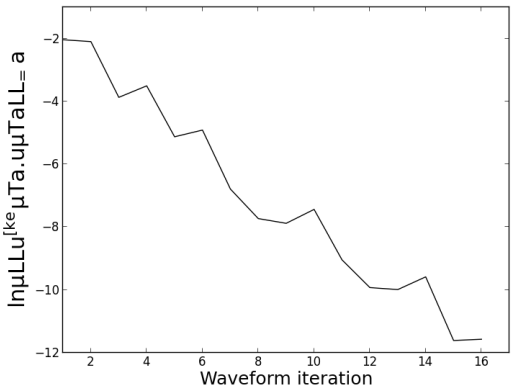


Fig. 2 The error of the waveform iterates at time T is computed relative to monodomain solution for a 4×4 decomposition of the problem using optimized transmission conditions. The convergence behavior of the PSWR algorithm is identical to the convergence behavior of the SWR algorithm.

conditions of the form

$$\mathscr{T}_{ij}[\cdot] = \left(\frac{d}{d\hat{n}} + p\right)[\cdot], \quad \mathscr{T}_{ji}[\cdot] = \left(\frac{d}{d\hat{n}} - p\right)[\cdot],$$

are used, where $\frac{d}{d\hat{n}}$ is the derivative in the normal direction, and $p = 1$. (A recursive formula is used to compute the transmission condition in lieu of discretizing the derivative in the normal direction). In each experiment a total of 16 full waveform iterations are completed. Timing results are obtained using the stampede supercomputer at the Texas Advanced Computing Center. Good parallel efficiency and speedup is observed in spite of the increase in the number of messages required by the PSWR algorithm. Note that the $4 \times 4 \times 1$ case is identically the SWR algorithm.

$N_x \times N_y \times N_k$	# cores	walltime	speedup	efficiency
$4 \times 4 \times 1$	16	293.02 seconds	1.00 $\times$	1.00
$4 \times 4 \times 2$	32	149.92 seconds	1.95 $\times$	0.98
$4 \times 4 \times 4$	64	75.48 seconds	3.89 $\times$	0.97
$4 \times 4 \times 8$	128	38.71 seconds	7.57 $\times$	0.95
$4 \times 4 \times 16$	256	23.90 seconds	12.26 $\times$	0.77

In our second scaling study, 1600x1600 grid points are decomposed into 16x16 non-overlapping domains domains for 400 total time steps. Again, a centered five point finite difference stencil, a backward Euler time integrator, and optimized transmission conditions are used. Good parallel efficiency and speedup is observed even with the increased synchronization/number of messages in the system.

$N_x \times N_y \times N_k$	# cores	walltime	speedup	efficiency
$16 \times 16 \times 1$	256	295.86 seconds	1.00 $\times$	1.00
$16 \times 16 \times 2$	512	155.98 seconds	1.90 $\times$	0.95
$16 \times 16 \times 4$	1024	77.10 seconds	3.84 $\times$	0.96
$16 \times 16 \times 8$	2048	43.20 seconds	6.85 $\times$	0.86
$16 \times 16 \times 16$	4096	26.65 seconds	11.10 $\times$	0.69

In the above computations, a linear solve on a sub-domain takes $O(10^{-2})$ seconds. This relatively small problems size was chosen (100×100 on each subdomain) so that communications would play a substantial role in the timing studies. The presented efficiencies can be improved by partitioning the problem to be more computationally expensive (i.e. more time is spent in the linear solve).

5 Conclusions

In this paper, we have reformulated classical Schwarz waveform relaxation to allow for pipeline-parallel computation of the waveform iterates, after an initial startup cost. Theoretical estimates for the parallel speedup and communication overhead are presented, along with scaling studies to show the effectiveness of the pipeline Schwarz waveform relaxation algorithm.

Acknowledgements This work was supported in part by Michigan State University through computational resources provided by the Institute for Cyber-Enabled Research and AFOSR Grant FA9550-12-1-0455. This work also used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number OCI-1053575.

References

1. Bennequin, D., Gander, M.J., Halpern, L.: A homographic best approximation problem with application to optimized Schwarz waveform relaxation. Math. Comp. **78**(265), 185–223 (2009). DOI 10.1090/S0025-5718-08-02145-5. URL http://dx.doi.org/10.1090/S0025-5718-08-02145-5
2. Bjørhus, M.: On domain decomposition, subdomain iteration and waveform relaxation. Ph.D. thesis, PhD thesis, University of Trondheim, Norway (1995)
3. Christlieb, A., Macdonald, C., Ong, B.: Parallel high-order integrators. SIAM J. Sci. Comput. **32**(2), 818–835 (2010)
4. Courvoisier, Y., Gander, M.J.: Time domain maxwell equations solved with schwarz waveform relaxation methods. In: Domain Decomposition Methods in Science and Engineering XX, pp. 263–270. Springer (2013)
5. Feamster, N., Balakrishnan, H., Rexford, J., Shaikh, A., van der Merwe, J.: The case for separating routing from routers. In: Proceedings of the ACM SIGCOMM Workshop on Future Directions in Network Architecture, FDNA '04, pp. 5–12. ACM, New York, NY, USA (2004). DOI 10.1145/1016707.1016709. URL http://doi.acm.org/10.1145/1016707.1016709

6. Gander, M.J., Halpern, L., Nataf, F., et al.: Optimal convergence for overlapping and non-overlapping schwarz waveform relaxation. In: the Eleventh International Conference on Domain Decomposition Methods, CH. Lai, P. Bjørstad, M. Cross, and O. Widlund, eds, pp. 27–36. Citeseer (1999)
7. Gander, M.J., Stuart, A.M.: Space-time continuous analysis of waveform relaxation for the heat equation. SIAM J. Sci. Comput. **19**(6), 2014–2031 (1998). DOI 10.1137/S1064827596305337
8. Giladi, E., Keller, H.B.: Space-time domain decomposition for parabolic problems. Numer. Math. **93**(2), 279–313 (2002). DOI 10.1007/s002110100345
9. Japhet, C., Omnes, P.: Optimized schwarz waveform relaxation for porous media applications. In: Domain Decomposition Methods in Science and Engineering XX, pp. 585–592. Springer (2013)
10. Lemarié, F., Patrick, M., Debreu, L., Blayo, E.: Sensitivity of an Ocean-Atmosphere Coupled Model to the Coupling Method : Study of Tropical Cyclone Erica. URL `http://hal.inria.fr/hal-00872496`
11. Tipparaju, V., Gropp, W., Ritzdorf, H., Thakur, R., Traff, J.: Investigating high performance rma interfaces for the mpi-3 standard. In: Parallel Processing, 2009. ICPP '09. International Conference on, pp. 293–300 (2009). DOI 10.1109/ICPP.2009.54
12. Vandewalle, S.G., Van de Velde, E.F.: Space-time concurrent multigrid waveform relaxation. Ann. Numer. Math. **1**(1-4), 347–360 (1994). Scientific computation and differential equations (Auckland, 1993)

Algorithm xxx: RIDC Methods – A Family of Parallel Time-Integrators

Benjamin W. Ong, Michigan Technological University
Ronald D. Haynes, Memorial University of Newfoundland
Kyle Ladd, Barracuda Networks

Revisionist integral deferred correction (RIDC) methods are a family of parallel–in–time methods to solve systems of initial values problems. The approach is able to bootstrap lower order time integrators to provide high order approximations in approximately the same wall clock time, hence providing a multiplicative increase in the number of compute cores utilized. Here we provide a library which automatically produces a parallel–in–time solution of a system of initial value problems given user supplied code for the right hand side of the system and a sequential code for a first-order time step. The user supplied time step routine may be explicit or implicit and may make use of any auxiliary libraries which take care of the solution of any nonlinear algebraic systems which may arise or the numerical linear algebra required.

Categories and Subject Descriptors: G.1.0 [**Numerical Analysis**]: Parallel Algorithms; G.1.7 [**Numerical Analysis**]: Ordinary Differential Equations – Initial Value Problems

General Terms: Algorithms, Performance

Additional Key Words and Phrases: Parallel–in–time, deferred correction

ACM Reference Format:
Benjamin W. Ong, Ronald D. Haynes and Kyle Ladd, 2014. RIDC Algorithms: A Family of Parallel Time-Integrators *ACM Trans. Embedd. Comput. Syst.* 0, 0, Article 0 (2015?), 13 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

The fast, accurate solution of an initial-value problem (IVP) of the form

$$\mathbf{y}'(t) = f(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0, \quad t \in [0, T], \tag{1}$$

where $\mathbf{y}(t) \in \mathbb{C}^N$, $f : [\mathbb{R} \times C^N] \to \mathbb{C}^N$, is of practical interest in scientific computing. IVP (1) often arises from the spatial discretization of partial differential equations, and may require either an explicit or implicit time-integrator. The purpose of this software is to "wrap" a user-implemented first-order explicit or implicit solver for IVP (1) into a high-order parallel solver; that is, given (t_n, y_n, f_n), a user specifies a function that returns $(t_{n+1}, y_{n+1}, f_{n+1})$ using either a forward Euler or backward Euler integrator. This work differs from existing ODE integration software or libraries, where a user typically only needs to specify the system of ODEs and relevant problem parameters. The upside is that our software provides a parallel–in–time solution while giving the user complete control of the first-order time step routine. For example, the user may chose their own quality libraries for the solution of systems of nonlinear algebraic

This work is supported by AFOSR Grant FA9550-12-1-0455, and NSERC Discovery grant (Canada).
Author's addresses: R. D. Haynes, Department of Mathematics and Statistics, Memorial University of Newfoundland; B. W. Ong, Mathematical Sciences, Michigan Technological University; K. Ladd, Barracuda Networks

DOI:http://dx.doi.org/10.1145/0000000.0000000

equations or efficient linear system solvers particularly tuned to the structure of their problems.

There are three general approaches for a time-parallel solution of IVPs [Burrage 1993]. One approach is "*parallelism-across-the-problem*", where a problem is decomposed into sub-problems that can be computed in parallel, and an iterative procedure is used to couple the sub-problems. Examples of this class of methods include parallel wave-form relaxation methods [Vandewalle and Roose 1989]. The second approach is "*parallel-across-the-step*" methods, where the time domain is partitioned into smaller temporal subdomains which are solved simultaneously. Examples of this class of methods include parareal methods [Maday and Turinici 2002; Gander and Vandewalle 2007], where the method alternates between applying a coarse sequential solver and a fine parallel solver. The third approach is "*parallelism-across-the-method*", where one exploits concurrent function evaluations within a step to generate a parallel time integrator. This approach typically allows for *small-scale* parallelism, constrained by the number of function evaluations that can evaluated in parallel. This is often related to the order of the approximation. Examples of Runge–Kutta methods where stages can be evaluated in parallel include [Miranker and Liniger 1967; Enenkel and Jackson 1997; Ketcheson and bin Waheed 2014]. Alternatively, one can use a predictor–corrector framework to generate parallel-across-the-method time integrators. This includes parallel extrapolation methods [Kappeller et al. 1996], and RIDC integrators [Christlieb et al. 2010; Christlieb and Ong 2011], which are the focus of this paper. A survey of parallel time integration methods has recently appeared [Gander 2015].

1.1. Related Software

There are several well established software packages for solving differential algebraic equations, however not many of them are able to solve IVPs (1) in parallel. For sequential integrators, probably the most well known are MATLAB routines `ode45`, `ode23`, `ode15s` [Shampine et al. 1999] to solve their systems of differential equations. These schemes use embedded RK pairs or numerical differentiation formulas (of the specified order) to approximate solutions to the differential equations using adaptive time-stepping. Readers might also be familiar with DASSL [Petzold 1983], which implements backward differentiation formulas of order one through five. The nonlinear system at each time-step is solved by Newton's method, and the resulting linear systems are solved using routines from LINPACK. DASSL leverages the SLATEC Common Mathematical Library [Vandevender and Haskell 1982] for step-size adaptivity. Also popular are ODEPACK [Hindmarsh 1983] and VODE [Brown et al. 1989], a collection of fortran solvers for IVPs, SUNDIALS, a suite of robust time integrators and nonlinear solvers [Hindmarsh et al. 2005], and there are a variety of ODE and DAE time steppers implemented in PETSc [Balay et al. 2014] and GSL [Gough 2009].

The selection of parallel solvers for IVPs is fairly sparse. EPPEER [Schmitt 2013] is a Fortran95/OpenMP implementation of explicit parallel two-step peer methods [Weiner et al. 2008] for the solution of ODEs on multicore architectures. PyPFASST [Emmett 2013] is a python implementation of a modified parareal solver for ODEs and PDEs [Emmett and Minion 2012]. XBRAID [Schroder et al. 2015] is a C library that implements a multigrid-reduction-in-time algorithm [Falgout et al. 2014], where multiple time-grids of different granularity are distributed across processors using MPI. PFASST++ [Emmett et al. 2015] is a C++ implementation of the " parallel full approximation scheme in space and time (PFASST) algorithm [Emmett and Minion 2014]. There are other implementations (such as the dependency-driven parareal framework developed at Oakridge National Laboratory [Elwasif et al. 2011]) that do not appear to be available for download at present.

2. REVIEW OF RIDC METHODS

Spectral deferred correction (SDC) [Dutt et al. 2000] provides an iterative correction of an approximate solution by solving an integral formulation of an error equation. This integral form stabilizes the classical differential deferred correction approach. RIDC is a re–formulation of SDC, pipelining successive calculations so that corrections can be obtained in parallel with an appropriate time lag. SDC, in contrast, is a sequential algorithm. Unlike the spectral deferred correction, which uses Gauss–Lobatto nodes, RIDC uses uniformly spaced nodes to minimize the memory footprint and to allow one to embed high order integrators [Christlieb et al. 2009; 2010].

The basic idea of the IDC and RIDC approaches is to formulate associated error IVPs which *correct* numerical errors from the solutions to IVP (1); the parallelism arises from the ability to simultaneously compute solutions to both IVP (1) and solutions to the associated error IVPs. In this section, we review the formulation of the error equations, discretizations, and parallel properties of the RIDC algorithm. Please refer to [Christlieb et al. 2010; Christlieb and Ong 2011] for accuracy and stability properties of the RIDC approach.

2.1. Error IVPs

Denote the (unknown) exact solution of IVP (1) as $y(t)$, and the approximate solution as $u(t)$, with $u(0) = y(0)$. The error in the approximate solution is $e(t) = y(t) - u(t)$. Define the residual (sometimes known as the defect) as $r(t) = u'(t) - f(t, u)$. Then, the time derivative of the error satisfies

$$e'(t) = y'(t) - u'(t) = f(t, u + e) - f(t, u) - r(t).$$

Since $e(0) = u(0) - y(0) = 0$, we have just derived the associated error IVP. For stability, the integral form of the error IVP is preferred [Dutt et al. 2000],

$$\left(e + \int_0^t r(\tau)\, d\tau \right)' = f(t, u + e) - f(t, u). \tag{2}$$

Observing that the corrected approximation $u + e$ is still an approximation if the error equation (2) is solved numerically, we adopt a more general notation which will allow us to iteratively correct the solution until a desired accuracy is reached. Denote the initial approximation as $u^{[0]}$, the pth approximation as $u^{[p]}$, and the error to $u^{[p]}$ as $e^{[p]}$. Then, the error equation can be rewritten as

$$\left(e^{[p]} + \int_0^t r^{[p]}(\tau)\, d\tau \right)' = f(t, u^{[p]} + e^{[p]}) - f(t, u^{[p]}), \tag{3}$$

where $r^{[p]} = u^{[p]}(t)' - f(t, u^{[p]})$.

2.2. Discretization

With some algebra, a first-order explicit discretization of (3), written in terms of the solution, gives

$$u_{n+1}^{[p+1]} = u_n^{[p+1]} + \Delta t f(t_n, u_n^{[p+1]}) - \Delta t f(t_n, u_n^{[p]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau. \tag{4}$$

Likewise a first-order implicit discretization of (3) gives

$$u_{n+1}^{[p+1]} = u_n^{[p+1]} + \Delta t f(t_{n+1}, u_{n+1}^{[p+1]}) - \Delta t f(t_{n+1}, u_{n+1}^{[p]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau. \tag{5}$$

In both semi-descretizations (4) and (5), a sufficiently accurate quadrature is needed to approximate the integrals present [Dutt et al. 2000]. If a first order predictor was applied to obtain an approximate solution to (1), and first order correctors such as (4) and (5) are used, approximating the quadrature using

$$\int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau \approx \begin{cases} \sum_{\nu=0}^{p+1} \alpha_{p\nu} f(t_{n+1-\nu}, u^{[p]}_{n+1-\nu}), & \text{if } n \geq p, \\ \sum_{\nu=0}^{p+1} \alpha_{p\nu} f(t_\nu, u^{[p]}_\nu), & \text{if } n < p, \end{cases},$$

where $\alpha_{p\nu}$ are quadrature weights,

$$\alpha_{p\nu} = \begin{cases} \int_{t_n}^{t_{n+1}} \prod_{i=0, i\neq\nu}^{p+1} \frac{(t - t_{n+1-i})}{(t_{n+1-\nu} - t_{n+1-i})}\, dt, & \text{if } n \geq p, \\ \int_{t_n}^{t_{n+1}} \prod_{i=0, i\neq\nu}^{p+1} \frac{(t - t_i)}{(t_\nu - t_i)}\, dt, & \text{if } n < p \end{cases}$$

results in a Pth order method, if $(P-1)$ such corrections are applied.

2.3. Stability

A study of the (linear) stability of explicit RIDC methods is provided in [Christlieb et al. 2010] and for implicit RIDC methods in [Christlieb and Ong 2011]. The results indicate that the region of absolute stability of RIDC methods approach the region of absolute stability of the underlying predictor as the number of time steps increases. Moreoever, for the implicit RIDC4-BE method preserves the A–stability property of backward Euler.

2.4. Parallelization

As mentioned earlier, the parallelism arises from the ability to simultaneously compute solutions to both IVP (1) and solutions to the associated error IVPs (3). This is possible if there is some staggering to decouple solutions of IVP (1) and the error equations. As shown in Figure 1, staggering of one timestep is required to compute solutions in a pipeline parallel fashion. For example, while the predictor computes a solution at time t_{10}, the first corrector computes the correction at time t_9, the second corrector the second correction at time t_8, and so on. We discuss the "memory footprint" and the startup routine required by the RIDC method in Section 2.5 before presenting a pseudo algorithm for the RIDC methods on page 6.

2.5. Memory Footprint, Efficiency, Start-up and Shut-down

Figure 1 also shows the "memory footprint" required to execute the RIDC method in a pipeline-parallel fashion. The memory footprint are copies of the solution vector evaluated at earlier correction/prediction levels and time steps; one can also think of the memory footprint as the discretization stencil across the different correction and prediction levels. For a Pth order RIDC method, the $(P-1)$st correction update (i.e. solving error IVP #(P-1)) requires a stencil of size $(P+1)$, the $(P-2)$nd correction requires an additional $(P-2)$ size stencil, the $(P-2)$nd correction requires an additional $(P-3)$ size stencil, and so on. The total memory footprint required for a Pth order RIDC method

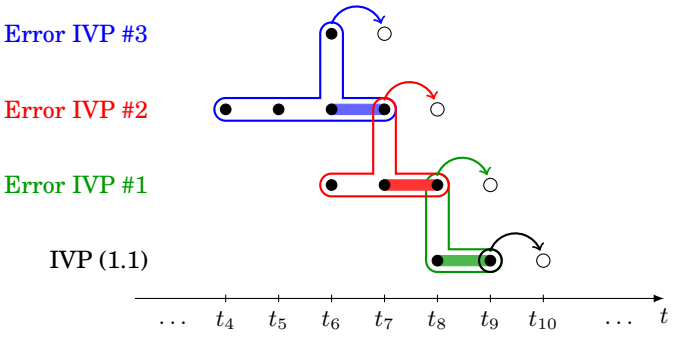


Fig. 1. In a RIDC method, solution values and correction terms are computed in a pipeline fashion. For example, while a processor is computing a solution to IVP (1) at t_{10}, a second processor computes corrections to the numerical error at time t_9, a third processor computes additional corrections at time t_8, and so forth. (i.e., the open circles denote solutions that are simultaneously computed). The solid circles denote stored solution values that are needed for the quadrature approximation.

is

$$\left(\sum_{i=1}^{P-1}(i+1)\right)+1=\frac{(P-1)(P)}{2}+(P-1)+1=\frac{P(P+1)}{2}.$$

In [Christlieb et al. 2010] it is shown that the ratio of time steps taken by Pth-order RIDC–Euler method, using K steps before a restart, to the number of steps taken by the forward Euler method is

$$\gamma=1+\frac{(P-1)^2}{K}.$$

This shows that the method becomes more efficient (in terms of wall-clock time) as K increases. One does have to balance a large value of K with the possible increase in error this may cause. A study of this balance is provided in [Christlieb et al. 2010].

Because of the staggering, start-up steps are needed to fill the memory footprint. As discussed in [Christlieb et al. 2010], one should control the start-up steps to minimize the size of the memory footprint; that is, it is more desirable to stall the predictors and lower-level correctors initially (as appropriate) until all predictors and correctors can be marched in a pipeline fashion with the minimal memory footprint. For example, Figure 2 shows the start-up routine for a fourth-order RIDC method. Initially, only the predictor advances the solution from t_0 to t_1 in step one. In steps two and three, both the predictor and first corrector are advanced to populate the memory stencil in preparation for the second corrector. In step four, only the second corrector is advanced; the predictor and first corrector are stalled because the memory stencil needed to advance the second corrector from t_1 to t_2 is *the same* memory stencil needed to advance the corrector from t_0 to t_1.

Although this concept is easy to grasp, the startup algorithm looks non-intuitive at first glance. Algorithm 1 specifies the nuts-and-bolts of the start-up routine. The RIDC method can be run in a pipe-line fashion (with the minimal memory footprint) after $startnum-1$ initialization steps, where $startnum=min(1,\frac{p(p+1)}{2}-1)$. For example, no initialization is required if $p=1$. If $p=4$, eight initialization steps are required – the RIDC method starts marching in a pipeline fashion at step nine. In the RIDC software, this startup routine is implemented using the $filter$ variable.

The shut-down routine for the RIDC method is straightforward; each predictor and corrector only advances the solution until the final time, t_F, is reached. The parallel RIDC pseudo-code is summarized in Algorithm 2.

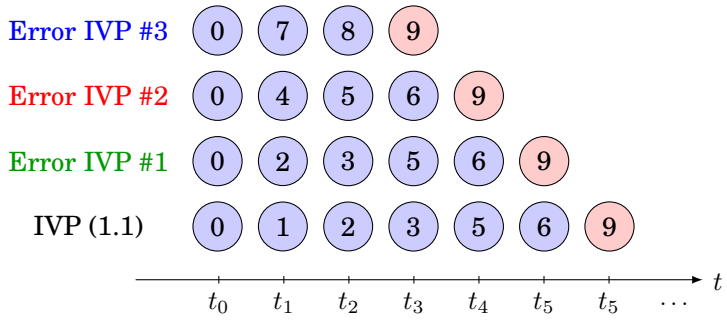


Fig. 2. Start-up routine for a fourth-order RIDC method. Observe that the predictor and lower order correctors are occasionally stalled to ensure that a minimal memory footprint is used for the RIDC method. The fourth-order RIDC startup takes eight steps; from step nine on, the RIDC method can be marched in a pipeline fashion.

ALGORITHM 1: RIDC Startup-Routine

$startnum = min(1, \frac{p(p+1)}{2} - 1)$;
for $p = 1$ **to** $(P-1)$ **do**
  march previous levels (i.e. $0, \ldots, (p-1)$) in a pipe for one step;
  march current level $(p-1)$ steps;
  march levels $0, \ldots, p$, in a pipe for one step;
end

ALGORITHM 2: RIDC Pseudo Code

fill memory stencil, compute $startnum$;
for $nt = startnum$ **to** NT **do**
  for $p = 0$ **to** $(P-1)$ **do in parallel**
    if $p = 0$ **then**
      use `step` to advance solution on prediction level (if t_F not reached on prediction level);
    else
      use `corr_fe` or `corr_be` to advance solution on correction level p (if t_F not reached on correction level p);
    end
  end
  update memory stencil ;
end

3. RIDC SOFTWARE

To utilize popular sequential integrators as described in Section 1.1, a user often specifies $f(t, y)$, the range of integration $[0, T]$, the initial condition $y_0 = y(0)$ (and for DASSL, the derivative $y'_0 = y'(0)$), and integrator parameters (such as parameters for controlling step-size adaptivity). While these general purpose time integration routines are convenient and easy to use, this "black-box" approach (for example, a user does not have to deal with the nonlinear solves arising from the backward differentiation formulas) sometimes precludes the use of additional information, such as the use of a problem-specific preconditioner, sparsity of the matrices, or multigrid iterative solvers.

The RIDC software presented here differs from the type of time-integration software mentioned above in that a first-order, user-specified, advance for $t \to t + \Delta t$ is bootstrapped to generate a high-order, parallel integrator using the integral deferred correction framework described in Section 2.

3.1. Under the hood

The RIDC software and examples are coded in C++; task parallelism is achieved using OpenMP threads to solve the predictors and the correctors in parallel. This mode of parallelism was chosen to accommodate the data movement/communication required by the RIDC algorithm when solving equations (4) and (5). We assume that the user-defined step routine to advance the solution is a first-order sequential integrator, although with some minor modifications to the RIDC software provided, bootstrapping higher order integrators is possible. The RIDC software can also be modified to leverage a thread-safe user-defined step routine, for example a CUDA-accelerated step routine [Ong et al. 2012] or an MPI-parallelized step routine [Christlieb et al. 2012] can be utilized, see Section 3.3. If the step routine uses an explicit Euler integrator, the RIDC software assumes that u_{n+1} satisfies

$$u_{n+1} - u_n = \Delta t f(t_n, u_n).$$

If the step routine uses an implicit Euler integrator, the RIDC software assumes that u_{n+1} satisfies

$$u_{n+1} - u_n = \Delta t f(t_{n+1}, u_{n+1}).$$

The RIDC software treats this step routine as a black box, as depicted in Figure 3.

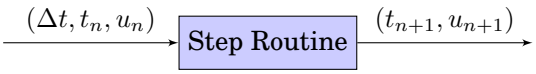


Fig. 3. User-defined step routine that advances a solution from t_n to t_{n+1}.

The RIDC functions solve equations (4) and (5) by creating the necessary data structures to store copies of the solution vector described in Section 2.5, and then performing the appropriate algebraic computations on these stored solution values. First, consider the explicit Euler discretization of the error equation (4). Observe that $u_{n+1}^{[p+1]}$ can be constructed by applying the user-defined step routine to $u_n^{[p+1]}$ to obtain $\tilde{v}_{n+1}^{[p+1]}$, and then adding $-\Delta t f(t_n, u_n^{[p]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau$ to $\tilde{v}_{n+1}^{[p+1]}$ to finally obtain $u_{n+1}^{[p+1]}$. The explicit RIDC wrapper is displayed in Figure 4.

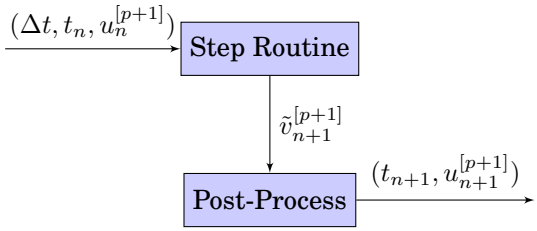


Fig. 4. A visualization of the RIDC wrapper to obtain a solution to equation (4). The post process takes an input $\tilde{v}_{n+1}^{[p+1]}$ and returns $\tilde{v}_{n+1}^{[p+1]} - \Delta t f(t_n, u_n^{[p]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau$.

A similar observation can be made about the implicit Euler discretization of the error equation (5), however, one first constructs the intermediate value $\tilde{v}_n^{[p+1]} = u_n^{[p+1]} - \Delta t f(t_n, u_n^{[p]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau$, and then applies the user-defined step function to $\tilde{v}_n^{[p+1]}$. The implicit RIDC wrapper is displayed in Figure 5.

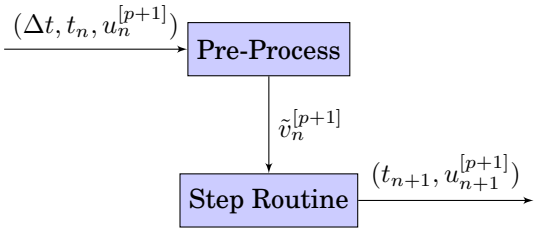


Fig. 5. A visualization of the RIDC wrapper to obtain a solution to equation (5). The pre-process takes an input $u_n^{[p+1]}$ and returns $\tilde{u}_n^{[p+1]} - \Delta t f(t_n, u_n^{[p+1]}) + \int_{t_n}^{t_{n+1}} f(\tau, u^{[p]})\, d\tau$.

3.2. Discussion

The computational overhead of RIDC methods resides mainly in the quadrature approximation, and the subsequent linear combinations used to compute the corrected solutions. Provided this computational overhead is small compared to an evaluation of the step routine, good parallel speedup is achieved. In practice, this is almost always the case for implicit RIDC methods where solutions to linear equations, and/or Newton iterations are required. For explicit RIDC methods, good parallel speedup is only observed when the step routine is sufficiently expensive, such as in the computation of self-consistent forces for an n-body problem [Christlieb et al. 2010].

As mentioned in Section 2.5, the RIDC method has to store copies of the solution vector evaluated at ealier correction/prediction levels. Although this memory requirement might appear restrictive, the memory footprint for high order single-step, multi-step or general linear methods are similar. Implicit RIDC methods also benefit from the loose coupling between the prediciton and corection equations; whereas a general implicit s-stage implicit RK method neccessitates the solution of a system of (potentially nonlinear) sN equations, where N are the number of differential algebraic equations. A pth-order RIDC method constructed using backward Euler integrators requires the solution of p decoupled systems of N (potentially nonlinear) algebraic equations.

3.3. Possible Generalizations

For clarity, only the simplest variant of the RIDC method (constructed using first order Euler integrators, uniform time-stepping, serial computation of the step routine) has been presented, and released as part of the base software version. Here, we make some remarks on how the base version of the software can be modified by the user to accommodate several generalizations discussed in this secton; indeed, the authors will release (when possible) modified versions of the software within the source repository that illustrate how to generate generalized RIDC integrators.

Step-size adaptivity for error control: In [Christlieb et al. 2015], various variants of adaptive RIDC methods were presented. In the simplest variant, one uses standard error control stratagies to adaptively select step-sizes while solving IVP (1). These adaptively selected step-sizes are used for solving the error equations (2). To build step-size adaptivity into the provided RIDC software, the following modifications will be needed: (i) modify the time-loop appropriately to allow for non-uniform steps, (ii) modify the driver file appropriately to take a user-defined tolerance (as opposed to the number of time steps), (iii) recompute the integration matrix containing the quadrature weights at every time step. The user will presumably provide an additional `adapt_step` function, which takes as inputs the solution at time t, the previous time step used, Δt_{old}, a tolerance tol, and returns the time step selected, Δt, and the solution at the new time step, $t + \Delta t$.

Restarts: As discussed in [Christlieb et al. 2010], the RIDC method accumulates error while running in a pipeline fashion – the most accurate solution does not propagate to the earlier prediction/correction levels. In some cases, it might be advantageous to

stop the RIDC method, and use the most accurate solution to "restart" the computation. This requires only a simple modification to the main RIDC loop in `ridc.cpp`.

Constructing RIDC methods using higher-order integrators: With a few modifications, it is possible to use higher-order single step integrators within the RIDC software. The memory stencil, integration matrix and quadrature approximations will need to be modified in `ridc.cpp`.

Semi-implicit RIDC methods: Although semi-implicit RIDC methods have been constructed and studied in [Ong et al. 2012], it is in general not possible to wrap a user-defined semi-implicit step function to solve the error equation (2). Consider the IVP

$$\mathbf{y}'(t) = f_N(t, \mathbf{y}) + f_S(t, \mathbf{y}),$$

where f_S contains stiff terms and f_N contains the nonstiff terms. A first-order user-defined step function to solve this IVP would look like

$$u_{n+1} - u_n = \Delta t f_N(t_n, u_n) + \Delta t f_S(t_{n+1}, u_{n+1}),$$

whereas the first-order IMEX discretization of the error equation (2) is

$$u_{n+1}^{[p+1]} = u_n^{[p+1]} + \Delta t \left[f_N(t_n, u_n^{[p+1]}) + f_S(t_{n+1}, u_{n+1}^{[p+1]}) \right] - \Delta t \left[f_N(t_n, u_n^{[p]}) + f_S(t_{n+1}, u_{n+1}^{[p]}) \right] + \int_{t_n}^{t_{n+1}} \left[f_N(\tau, u^{[p]}) + f_S(\tau, u^{[p]}) \right] d\tau.$$

Althought it is not obvious how to automaticaly bootstrap a semi–implicit step function, a user can leverage the data structures and quadrature approximations in `ridc.cpp` to construct a new `corr_fbe` function, which should look similar in structure to the users' `step` function.

Using accelerators for the step routine: Many computing clusters feature nodes with multiple accelerators, e.g. Nvidia GPGPUs or Intel Xeon Phis. If the user wishes to provide a step routine that is accelerated using these emerging architectures, the RIDC code can be modified to leverage *multiple* accelerators in a computational node. Modifications that are required include: adding an input variable "level" (an integer from 0 to $p-1$, where p is the desired order / number of accelerators in the system) into the step routine, a function call within the step function to specify the appropriate accelerator, e.g. `cudaSetDevice` for the NVIDIA GPGPUs, and a modification of `ridc.cpp` so that the prediction/correction level is fed into the step function, ensuring that the linear algebra is performed on the appropriate accelerator.

Using distributed MPI for the step routine: Although the RIDC software can be modified to allow for an MPI-distributed step routine (provided this step-routine is thread safe), we showed in [Haynes and Ong 2014] that a tighter coupling of the hybrid MPI-OpenMP formulation to reduce the number of messages is necessary for performance.

4. NUMERICAL EXPERIMENT

The software includes several examples verifying that the RIDC methods attain their designed orders of accuracy. As mentioned, these examples also serve as templates for the user to bootstrap their own first order time integration methods to give a parallel–in–time approximation. Good parallel speedup is observed when the computational overhead for the RIDC methods (namely, the quadrature approximation and the linear combinations to compute the corrected solutions) is small compared to an evaluation of the step routine. Here, we present the numerical results for the Brusselator in $\mathbb{R}^1$.

$$\begin{aligned} u_t &= A + u^2 v - (B+1)u + \alpha u_{xx}, \\ v_t &= Bu - u^2 v + \alpha v_{xx}, \end{aligned} \tag{6}$$

with $A = 1$, $B = 3$ and $\alpha = 0.02$, initial conditions

$$u(0, x) = 1 + \sin(2\pi x), \qquad v(0, x) = 3,$$

and boundary conditions

$$u(t, 0) = u(t, 1) = 1, \qquad v(t, 0) = v(t, 1) = 3.$$

A central finite difference approximation is used to discretize equation (6). The resulting nonlinear system of equations is solved using a Newton iteration. In the timing results, the Intel Math Kernel Library (MKL) is used to solve the linear system arising in each Newton iteration. The code for this example can be found in the `examples/brusselator_mkl` directory. Figure 6 shows a standard convergence study of error versus number of timesteps to demonstrate that the RIDC software bootstraps the first order integrator to generate a high-order method of the desired accuracy.

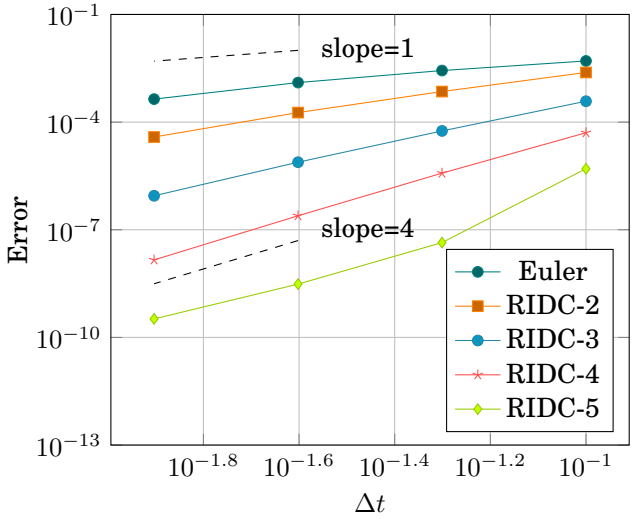


Fig. 6. Standard convergence study of error versus time step, Δt, showing that RIDC methods achieve their designed orders of accuracy.

In Figure 7, the walltime used to compute each ridc method is plotted to show the "weak scaling" capability of RIDC methods. For example, the fourth-order RIDC method computes a solution using four computing cores that is 3-5 orders of magnitude more accurate than the first order Euler solution in approximately the same wallclock time. Tming results using a serial three-stage, fifth-order RADAU IIA integrator is also presented. A fifth order RIDC method (with five computing cores) provides a solution with comparable accuracy in 10% of the walltime. The scaling studies were performed on a single computational node consisting of a dual socket Intel E5-2670v2 chipset.

5. CONCLUSIONS

In this paper, we presented the revisionist integral deferred correction (RIDC) software for solving systems of initial values problems. The approach bootstraps lower order time integrators to provide high order approximations in approximately the same wall clock time, providing a multiplicative increase in the number of compute cores utilized. The C++ framework produces a parallel–in–time solution of a system of initial value problems given user supplied code for the right hand side of the system and the sequential code for a first-order time step. The user supplied time step routine may be explicit or implicit and may make use of any auxiliary libraries which take care of the

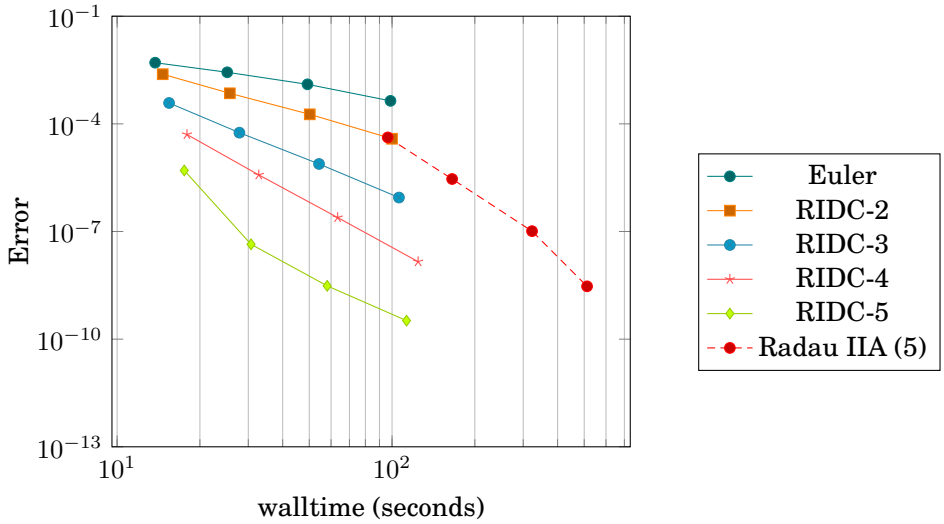


Fig. 7. The error as a function of walltime is plotted for various RIDC methods. Here, two computing cores (set via `OMP_NUM_THREADS=2`) is used to compute the second order RIDC method (RIDC-2), three computing cores are used to compute RIDC-3, four computing cores are used to compute RIDC-4, and give compute cores are used to compute RIDC-5. A single computing core was used to compute Radau IIA. The RIDC software computes a pth order solution in approximately the same wall clock time as an Euler solution, provided p computing cores are available. The parallel RIDC methods also provide good speedup over a serial Radau IIA integrator.

solution of the nonlinear algebraic systems which arise or the numerical linear algebra required.

REFERENCES

S Balay, S Abhyankar, M Adams, J Brown, P Brune, K Buschelman, V Eijkhout, W Gropp, D Kaushik, M Knepley, and others. 2014. PETSc Users Manual Revision 3.5. http://www.mcs.anl.gov/petsc/petsc-current/docs/manual.pdf. (2014).

Peter N. Brown, George D. Byrne, and Alan C. Hindmarsh. 1989. VODE: a variable-coefficient ODE solver. *SIAM J. Sci. Statist. Comput.* 10, 5 (1989), 1038–1051. DOI:http://dx.doi.org/10.1137/0910062

Kevin Burrage. 1993. Parallel methods for initial value problems. *Appl. Numer. Math.* 11, 1-3 (1993), 5–25. DOI:http://dx.doi.org/10.1016/0168-9274(93)90037-R Parallel methods for ordinary differential equations (Grado, 1991).

Andrew Christlieb, Ronald Haynes, and Benjamin Ong. 2012. A Parallel Space-Time Algorithm. *SIAM J. Sci. Comput.* 34, 5 (2012), 233–248.

Andrew Christlieb, Colin Macdonald, Benjamin Ong, and Raymond Spiteri. 2015. Revisionist integral deferred correction with adaptive step-size control. *Commun. Appl. Math. Comput. Sci.* 10, 1 (2015), 1–25. DOI:http://dx.doi.org/10.2140/camcos.2015.10.1

Andrew Christlieb and Benjamin Ong. 2011. Implicit parallel time integrators. *J. Sci. Comput.* 49, 2 (2011), 167–179. DOI:http://dx.doi.org/10.1007/s10915-010-9452-4

Andrew Christlieb, Benjamin Ong, and Jing-Mei Qiu. 2009. Comments on high-order integrators embedded within integral deferred correction methods. *Commun. Appl. Math. Comput. Sci.* 4 (2009), 27–56. DOI:http://dx.doi.org/10.2140/camcos.2009.4.27

Andrew Christlieb, Benjamin Ong, and Jing-Mei Qiu. 2010. Integral deferred correction methods constructed with high order Runge-Kutta integrators. *Math. Comp.* 79, 270 (2010), 761–783. DOI:http://dx.doi.org/10.1090/S0025-5718-09-02276-5

Andrew J. Christlieb, Colin B. Macdonald, and Benjamin W. Ong. 2010. Parallel high-order integrators. *SIAM J. Sci. Comput.* 32, 2 (2010), 818–835. DOI:http://dx.doi.org/10.1137/09075740X

Alok Dutt, Leslie Greengard, and Vladimir Rokhlin. 2000. Spectral deferred correction methods for ordinary differential equations. *BIT* 40, 2 (2000), 241–266.

Wael Elwasif, Samantha Foley, David Bernholdt, Lee Berry, Debasmita Samaddar, David Newman, and Raul Sanchez. 2011. A dependency-driven formulation of parareal: parallel-in-time solution of PDEs as a many-task application. In *Proceedings of the 2011 ACM international workshop on Many task computing on grids and supercomputers (MTAGS '11)*. ACM, New York, NY, USA, 15–24. DOI:http://dx.doi.org/10.1145/2132876.2132883

Matthew Emmett. 2013. PyPFASST: Parallel Full Approximation Scheme in Space and Time. (April 2013). http://pypfasst.readthedocs.org/en/latest

Matthew Emmett, Torbjorn Klatt, Robert Speck, and Daniel Ruprecht. 2015. parallel full approximation scheme in space and time. https://github.com/Parallel-in-Time/PFASST. (2015).

Matthew Emmett and Michael L. Minion. 2012. Toward an efficient parallel in time method for partial differential equations. *Commun. Appl. Math. Comput. Sci.* 7, 1 (2012), 105–132.

Matthew Emmett and Michael L. Minion. 2014. Efficient Implementation of a Multi-Level Parallel in Time Algorithm. In *Domain Decomposition Methods in Science and Engineering XXI*, Jocelyne Erhel, Martin J. Gander, Laurence Halpern, Géraldine Pichot, Taoufik Sassi, and Olof Widlund (Eds.). Lecture Notes in Computational Science and Engineering, Vol. 98. Springer International Publishing, 359–366. DOI:http://dx.doi.org/10.1007/978-3-319-05789-7_33

Robert F. Enenkel and Kenneth R. Jackson. 1997. DIMSEMs-diagonally implicit single-eigenvalue methods for the numerical solution of stiff ODEs on parallel computers. *Adv. Comput. Math.* 7, 1-2 (1997), 97–133. DOI:http://dx.doi.org/10.1023/A:1018986500842 Parallel methods for ODEs.

R. D. Falgout, S. Friedhoff, Tz. V. Kolev, S. P. MacLachlan, and J. B. Schroder. 2014. Parallel Time Integration with Multigrid. *SIAM Journal on Scientific Computing* 36, 6 (2014), C635–C661. DOI:http://dx.doi.org/10.1137/130944230

MartinJ. Gander. 2015. 50 Years of Time Parallel Time Integration. In *Multiple Shooting and Time Domain Decomposition Methods*, Thomas Carraro, Michael Geiger, Stefan Krkel, and Rolf Rannacher (Eds.). Contributions in Mathematical and Computational Sciences, Vol. 9. Springer International Publishing, 69–113. DOI:http://dx.doi.org/10.1007/978-3-319-23321-5_3

Martin J. Gander and Stefan Vandewalle. 2007. On the superlinear and linear convergence of the parareal algorithm. In *Domain decomposition methods in science and engineering XVI*. Lect. Notes Comput. Sci. Eng., Vol. 55. Springer, Berlin, 291–298. DOI:http://dx.doi.org/10.1007/978-3-540-34469-8_34

Brian Gough. 2009. *GNU scientific library reference manual*. Network Theory Ltd., sales@network-theory.co.uk.

Ronald D. Haynes and Benjamin W. Ong. 2014. MPI–OpenMP Algorithms for the Parallel Space–Time Solution of Time Dependent PDEs. In *Domain Decomposition Methods in Science and Engineering XXI*, Jocelyne Erhel, Martin J. Gander, Laurence Halpern, Géraldine Pichot, Taoufik Sassi, and Olof Widlund (Eds.). Lecture Notes in Computational Science and Engineering, Vol. 98. Springer International Publishing, 179–187. DOI:http://dx.doi.org/10.1007/978-3-319-05789-7_14

Alan C. Hindmarsh. 1983. ODEPACK, a systematized collection of ODE solvers. (1983), 55–64.

Alan C. Hindmarsh, Peter N. Brown, Keith E. Grant, Steven L. Lee, Radu Serban, Dan E. Shumaker, and Carol S. Woodward. 2005. SUNDIALS: suite of nonlinear and differential/algebraic equation solvers. *ACM Trans. Math. Software* 31, 3 (2005), 363–396. DOI:http://dx.doi.org/10.1145/1089014.1089020

M. Kappeller, M. Kiehl, M. Perzl, and M. Lenke. 1996. Optimized extrapolation methods for parallel solution of IVPs on different computer architectures. *Appl. Math. Comput.* 77, 23 (1996), 301 – 315. DOI:http://dx.doi.org/10.1016/S0096-3003(95)00219-7

David I. Ketcheson and Umair bin Waheed. 2014. A comparison of high-order explicit Runge-Kutta, extrapolation, and deferred correction methods in serial and parallel. *Commun. Appl. Math. Comput. Sci.* 9, 2 (2014), 175–200. DOI:http://dx.doi.org/10.2140/camcos.2014.9.175

Yvon Maday and Gabriel Turinici. 2002. A parareal in time procedure for the control of partial differential equations. *C. R. Math. Acad. Sci. Paris* 335, 4 (2002), 387–392. DOI:http://dx.doi.org/10.1016/S1631-073X(02)02467-6

Willard Miranker and Werner Liniger. 1967. Parallel methods for the numerical integration of ordinary differential equations. *Math. Comp.* 21 (1967), 303–320.

Benjamin Ong, Andrew Christlieb, and Andrew Melfi. 2012. *Parallel Semi-Implicit Time Integrators*. Technical Report. Michigan State University, East Lansing, MI. http://arxiv.org/pdf/1209.4297.pdf.

Linda Petzold. 1983. A description of DASSL: a differential/algebraic system solver. In *Scientific computing (Montreal, Que., 1982)*. IMACS, New Brunswick, NJ, 65–68.

Bernhard Schmitt. 2013. Peer methods for ordinary differential equations. (April 2013). http://www.mathematik.uni-marburg.de/~schmitt/peer/

Jacob Schroder, Robert Falgout, Tzanio Kolev, Ulrike Yang, Anders Petersson, Veselin Dobrev, Scott MacLachlan, Stephanie Friedhoff, and Ben O'Neil. 2015. XBraid: Parallel multigrid in time. http://llnl.gov/casc/xbraid. (2015).

Lawrence Shampine, Mark Reichelt, and Jacek Kierzenka. 1999. Solving index-1 DAEs in MATLAB and Simulink. *SIAM Rev.* 41, 3 (1999), 538–552 (electronic). DOI:http://dx.doi.org/10.1137/S003614459933425X

Walter Vandevender and Karen Haskell. 1982. The SLATEC mathematical subroutine library. *ACM SIGNUM Newsletter* 17, 3 (1982), 16–21.

Stefan Vandewalle and Dirk Roose. 1989. The parallel waveform relaxation multigrid method. In *Parallel Processing for Scientific Computing, Proceedings of the Third SIAM Conference on Parallel Processing for Scientific Computing,*. Soc. Indust. Appl. Math., Soc. Indust. Appl. Math., Philadelphia, PA, 152–156.

Rüdiger Weiner, Katja Biermann, Bernhard A. Schmitt, and Helmut Podhaisky. 2008. Explicit two-step peer methods. *Comput. Math. Appl.* 55, 4 (2008), 609–619. DOI:http://dx.doi.org/10.1016/j.camwa.2007.04.026

J Fourier Anal Appl
DOI 10.1007/s00041-015-9434-x

Stable Signal Recovery from Phaseless Measurements

Bing Gao[1] · **Yang Wang**[2] · **Zhiqiang Xu**[1]

Received: 29 April 2015

Abstract The aim of this paper is to study the stability of the ℓ_1 minimization for the compressive phase retrieval and to extend the instance-optimality in compressed sensing to the real phase retrieval setting. We first show that $m = \mathcal{O}(k \log(N/k))$ measurements are enough to guarantee the ℓ_1 minimization to recover k-sparse signals stably provided the measurement matrix A satisfies the strong RIP property. We second investigate the phaseless instance-optimality presenting a null space property of the measurement matrix A under which there exists a decoder Δ so that the phaseless instance-optimality holds. We use the result to study the phaseless instance-optimality for the ℓ_1 norm. This builds a parallel for compressive phase retrieval with the classical compressive sensing.

Keywords Phase retrieval · Sparse signals · Compressed sensing

Mathematics Subject Classification 94A12

Communicated by Peter G. Casazza.

✉ Zhiqiang Xu
xuzq@lsec.cc.ac.cn

Bing Gao
gaobing@lsec.cc.ac.cn

Yang Wang
yangwang@ust.hk

[1] LSEC, Institute of Computational Mathematics, Academy of Mathematics and System Science, Chinese Academy of Sciences, Beijing 100091, China

[2] Department of Mathematics, The Hong Kong University of Science and Technology, Clear Watre Bay, Kowloon, Hong Kong

1 Introduction

In this paper we consider the phase retrieval for sparse signals with noisy measurements, which arises in many different applications. Assume that

$$b_j := |\langle a_j, x_0\rangle| + e_j, \quad j = 1, \ldots, m$$

where $x_0 \in \mathbb{R}^N$, $a_j \in \mathbb{R}^N$ and $e_j \in \mathbb{R}$ is the noise. Our goal is to recover x_0 up to a unimodular scaling constant from $b := (b_1, \ldots, b_m)^\top$ with the assumption of x_0 being approximately k-sparse. This problem is referred to as the *compressive phase retrieval problem* [9].

The paper attempts to address two problems. Firstly we consider the stability of ℓ_1 minimization for the compressive phase retrieval problem where the signal x_0 is approximately k-sparse, which is the ℓ_1 minimization problem defined as follows:

$$\min \|x\|_1 \quad \text{subject to} \quad \big\| |Ax| - |Ax_0| \big\|_2 \le \epsilon, \tag{1.1}$$

where $A := [a_1, \ldots, a_m]^\top$ and $|Ax_0| := [|\langle a_1, x_0\rangle|, \ldots, |\langle a_m, x_0\rangle|]^\top$. Secondly we investigate instance-optimality in the phase retrieval setting.

Note that in the classical compressive sensing setting the stable recovery of a k-sparse signal $x_0 \in \mathbb{C}^N$ can be done using $m = \mathcal{O}(k \log(N/k))$ measurements for several classes of measurement matrices A. A natural question is whether stable compressive phase retrieval can also be attained with $m = \mathcal{O}(k \log(N/k))$ measurements. This has indeed proved to be the case in [6] if $x_0 \in \mathbb{R}^N$ and A is a random real Gaussian matrix. In [8] a two-stage algorithm for compressive phase retrieval is proposed, which allows for very fast recovery of a sparse signal if the matrix A can be written as a product of a random matrix and another matrix (such as a random matrix) that allows for efficient phase retrieval. The authors proved that stable compressive phase retrieval can be achieved with $m = \mathcal{O}(k \log(N/k))$ measurements for complex signals x_0 as well. In [10], the strong RIP (S-RIP) property is introduced and the authors show that one can use the ℓ_1 minimization to recover sparse signals up to a global sign from the *noiseless* measurements $|Ax_0|$ provided A satisfies S-RIP. Naturally, one is interested in the performance of ℓ_1 minimization for the compressive phase retrieval with noisy measurements. In this paper, we shall show that the ℓ_1 minimization scheme given in (1.1) will recover a k-sparse signal stably from $m = \mathcal{O}(k \log(N/k))$ measurements, provided that the measurement matrix A satisfies the strong RIP (S-RIP) property. This establishes an important parallel for compressive phase retrieval with the classical compressive sensing. Note that in [11] such a parallel in terms of the null space property was already established.

The notion of *instance optimality* was first introduced in [5]. We use $\|x\|_0$ to denote the number of non-zero elements in x. Given a norm $\|\cdot\|_X$ such as the ℓ_1-norm and $x \in \mathbb{R}^N$, the best k-term approximation error is defined as

$$\sigma_k(x)_X := \min_{z \in \Sigma_k} \|x - z\|_X,$$

where

$$\Sigma_k := \{x \in \mathbb{R}^N : \|x\|_0 \le k\}.$$

We use $\Delta : \mathbb{R}^m \mapsto \mathbb{R}^N$ to denote a decoder for reconstructing x. We say the pair (A, Δ) is *instance optimal of order k with constant* C_0 if

$$\|x - \Delta(Ax)\|_X \le C_0 \sigma_k(x)_X \tag{1.2}$$

holds for all $x \in \mathbb{R}^N$. In extending it to phase retrieval, our decoder will have the input $b = |Ax|$. A pair (A, Δ) is said to be *phaseless instance optimal of order k with constant* C_0 if

$$\min\Big\{\|x - \Delta(|Ax|)\|_X, \|x + \Delta(|Ax|)\|_X\Big\} \le C_0 \sigma_k(x)_X \tag{1.3}$$

holds for all $x \in \mathbb{R}^N$. We are interested in the following problem : *Given* $\|\cdot\|_X$ *and* $k < N$, *what is the minimal value of m for which there exists* (A, Δ) *so that* (1.3) *holds?*

The null space $\mathcal{N}(A) := \{x \in \mathbb{R}^N : Ax = 0\}$ of A plays an important role in the analysis of the original instance optimality (1.2) (see [5]). Here we present a null space property for $\mathcal{N}(A)$, which is necessary and sufficient, for which there exists a decoder Δ so that (1.3) holds. We apply the result to investigate the instance optimality where X is the ℓ_1 norm. Set

$$\Delta_1(|Ax|) := \operatorname*{argmin}_{z \in \mathbb{R}^N}\Big\{\|z\|_1 : |Ax| = |Az|\Big\}.$$

We show that the pair (A, Δ_1) satisfies (1.3) with X being the ℓ_1-norm provided A satisfies the strong RIP property (see Definition 2.1). As shown in [10], the Gaussian random matrix $A \in \mathbb{R}^{m\times N}$ satisfies the strong RIP of order k for $m = \mathcal{O}(k \log(N/k)$. Hence $m = \mathcal{O}(k \log(N/k))$ measurements suffice to ensure the phaseless instance optimality (1.3) for the ℓ_1-norm exactly as with the traditional instance optimality (1.2).

2 Auxiliary Results

In this section we provide some auxiliary results that will be used in later sections. For $x \in \mathbb{R}^N$ we use $\|x\|_p := \|x\|_{\ell_p}$ to denote the p-norm of x for $0 < p \le \infty$. The measurement matrix is given by $A := [a_1, \ldots, a_m]^T \in \mathbb{R}^{m\times N}$ as before. Given an index set $I \subset \{1, \ldots, m\}$ we shall use A_I to denote the sub-matrix of A where only rows with indices in I are kept, i.e.,

$$A_I := [a_j : j \in I]^\top.$$

The matrix A satisfies the *Restricted Isometry Property* (*RIP*) *of order* k if there exists a constant $\delta_k \in [0, 1)$ such that for all k-sparse vectors $z \in \Sigma_k$ we have

$$(1 - \delta_k)\|z\|_2^2 \leq \|Az\|_2^2 \leq (1 + \delta_k)\|z\|_2^2.$$

It was shown in [2] that one can use ℓ_1-minimization to recover k-sparse signals provided that A satisfies the RIP of order tk and $\delta_{tk} < \sqrt{1 - \frac{1}{t}}$ where $t > 1$.

To investigate compressive phase retrieval, a stronger notion of RIP is given in [10]:

Definition 2.1 (*S-RIP*) We say the matrix $A = [a_1, \ldots, a_m]^\top \in \mathbb{R}^{m\times N}$ has the *Strong Restricted Isometry Property* of order k with bounds $\theta_-,\ \theta_+ \in (0, 2)$ if

$$\theta_-\|x\|_2^2 \leq \min_{I\subseteq[m],|I|\geq m/2} \|A_I x\|_2^2 \leq \max_{I\subseteq[m],|I|\geq m/2} \|A_I x\|_2^2 \leq \theta_+\|x\|_2^2 \tag{2.1}$$

holds for all k-sparse signals $x \in \mathbb{R}^N$, where $[m] := \{1, \ldots, m\}$. We say A has the *Strong Lower Restricted Isometry Property* of order k with bound θ_- if the lower bound in (2.1) holds. Similarly we say A has the *Strong Upper Restricted Isometry Property* of order k with bound θ_+ if the upper bound in (2.1) holds.

The authors of [10] proved that Gaussian matrices with $m = \mathcal{O}(tk\log(N/k))$ satisfy S-RIP of order tk with high probability.

Theorem 2.1 ([10]) *Suppose that* $t > 1$ *and* $A = (a_{ij}) \in \mathbb{R}^{m\times N}$ *is a random Gaussian matrix with* $m = \mathcal{O}(tk\log(N/k))$ *and* $a_{ij} \sim \mathcal{N}(0, \frac{1}{\sqrt{m}})$. *Then there exist* $\theta_-, \theta_+ \in (0, 2)$ *such that with probability* $1 - \exp(-cm/2)$ *the matrix A satisfies the S-RIP of order tk with constants* θ_- *and* θ_+, *where* $c > 0$ *is an absolute constant and* θ_-, θ_+ *are independent of t*.

The following is a very useful lemma for this study.

Lemma 2.1 *Let* $x_0 \in \mathbb{R}^N$ *and* $\rho \geq 0$. *Suppose that* $A \in \mathbb{R}^{m\times N}$ *is a measurement matrix satisfying the restricted isometry property with* $\delta_{tk} \leq \sqrt{\frac{t-1}{t}}$ *for some* $t > 1$. *Then for any*

$$\hat{x} \in \left\{x \in \mathbb{R}^N : \|x\|_1 \leq \|x_0\|_1 + \rho,\ \|Ax - Ax_0\|_2 \leq \epsilon\right\}$$

we have

$$\|\hat{x} - x_0\|_2 \leq c_1\epsilon + c_2\frac{2\sigma_k(x_0)_1}{\sqrt{k}} + c_2 \cdot \frac{\rho}{\sqrt{k}},$$

where $c_1 = \frac{\sqrt{2(1+\delta)}}{1-\sqrt{t/(t-1)}\delta}$, $c_2 = \frac{\sqrt{2}\delta+\sqrt{(\sqrt{t(t-1)}-\delta t)\delta}}{\sqrt{t(t-1)}-\delta t} + 1$.

Remark 2.1 We build the proof of Lemma 2.1 following the ideas of Cai and Zhang [2]. The full proof is given in Appendix for completeness. It is well-known that an effective method to recover approximately-sparse signals x_0 in the traditional compressive sensing is to solve

$$x^{\#} := \underset{x}{\operatorname{argmin}}\{\|x\|_1 : \|Ax - Ax_0\|_2 \leq \epsilon\}. \tag{2.2}$$

The definition of $x^{\#}$ shows that

$$\|x^{\#}\|_1 \leq \|x_0\|_1, \quad \|Ax^{\#} - Ax_0\|_2 \leq \epsilon,$$

which implies that

$$\|x^{\#} - x_0\|_2 \leq C_1\epsilon + C_2\frac{\sigma_k(x_0)_1}{\sqrt{k}},$$

provided that A satisfies the RIP condition with $\delta_{tk} \leq \sqrt{1 - 1/t}$ for $t > 1$ (see [2]). However, in practice one prefers to design fast algorithms to find an approximation solution of (2.2), say $\hat{x}$. Thus it is possible to have $\|\hat{x}\|_1 > \|x_0\|_1$. Lemma 2.1 gives an estimate of $\|\hat{x} - x_0\|_2$ for the case where $\|\hat{x}\|_1 \leq \|x_0\|_1 + \rho$.

Remark 2.2 In [7], Han and Xu extend the definition of S-RIP by replacing the $m/2$ in (2.1) by βm where $0 < \beta < 1$. They also prove that, for any fixed $\beta \in (0, 1)$, the $m \times N$ random Gaussian matrix satisfies S-RIP of order k with high probability provided $m = \mathcal{O}(k \log(N/k))$.

3 Stable Recovery of Real Phase Retrieval Problem

3.1 Stability Results

The following lemma shows that the map $\phi_A(x) := |Ax|$ is stable on Σ_k modulo a unimodular constant provided A satisfies strong lower RIP of order $2k$. Define the equivalent relation $\sim$ on $\mathbb{R}^N$ and $\mathbb{C}^N$ by the following: for any x, y, $x \sim y$ iff $x = cy$ for some unimodular scalar c, where x, y are in $\mathbb{R}^N$ or $\mathbb{C}^N$. For any subset Y of $\mathbb{R}^N$ or $\mathbb{C}^N$ the notation $Y/\sim$ denotes the equivalent classes of elements in Y under the equivalence. Note that there is a natural metric $D_\sim$ on $\mathbb{C}^N/\sim$ given by

$$D_\sim(x, y) = \min_{|c|=1} \|x - cy\|.$$

Our primary focus in this paper will be on $\mathbb{R}^N$, and in this case $D_\sim(x, y) = \min\{\|x - y\|_2, \|x + y\|_2\}$.

Lemma 3.1 *Let $A \in \mathbb{R}^{m\times N}$ satisfy the strong lower RIP of order $2k$ with constant θ_-. Then for any $x, y \in \Sigma_k$ we have*

$$\||Ax| - |Ay|\|_2^2 \geq \theta_- \min(\|x - y\|_2^2, \|x + y\|_2^2).$$

Proof For any $x, y \in \Sigma_k$ we divide $\{1, \ldots, m\}$ into two subsets:

$$T = \{j : \ \text{sign}(\langle a_j, x\rangle) = \text{sign}(\langle a_j, y\rangle)\}$$

and

$$T^c = \{j : \ \text{sign}(\langle a_j, x\rangle) = -\text{sign}(\langle a_j, y\rangle)\}.$$

Clearly one of T and T^c will have cardinality at least $m/2$. Without loss of generality we assume that T has cardinality no less than $m/2$. Then

$$\begin{aligned} \||Ax| - |Ay|\|_2^2 &= \|A_T x - A_T y\|_2^2 + \|A_{T^c} x + A_{T^c} y\|_2^2 \\ &\geq \|A_T x - A_T y\|_2^2 \\ &\geq \theta_- \|x - y\|_2^2 \\ &\geq \theta_- \min(\|x - y\|_2^2, \|x + y\|_2^2). \end{aligned}$$

□

Remark 3.1 Note that the combination of Lemma 3.1 and Theorem 2.1 shows that for an $m \times N$ Gaussian matrix A with $m = O(k \log(N/k))$ one can guarantee the stability of the map $\phi_A(x) := |Ax|$ on $\Sigma_k / \sim$.

3.2 The Main Theorem

In this part, we will consider how many measurements are needed for the stable sparse phase retrieval by ℓ_1-minimization via solving the following model:

$$\min \|x\|_1 \ \text{ subject to } \ \||Ax| - |Ax_0|\|_2^2 \leq \epsilon^2, \tag{3.1}$$

where A is our measurement matrix and $x_0 \in \mathbb{R}^N$ is a signal we wish to recover. The next theorem tells under what conditions the solution to (3.1) is stable.

Theorem 3.1 *Assume that $A \in \mathbb{R}^{m \times N}$ satisfies the S-RIP of order tk with bounds $\theta_-, \theta_+ \in (0, 2)$ such that*

$$t \geq \max\left\{\frac{1}{2\theta_- - \theta_-^2}, \frac{1}{2\theta_+ - \theta_+^2}\right\}.$$

Then any solution $\hat{x}$ for (3.1) satisfies

$$\min\{\|\hat{x} - x_0\|_2, \|\hat{x} + x_0\|_2\} \leq c_1 \epsilon + c_2 \frac{2\sigma_k(x_0)_1}{\sqrt{k}},$$

where c_1 and c_2 are constants defined in Lemma 2.1.

Proof Clearly any $\hat{x} \in \mathbb{R}^N$ satisfying (3.1) must have

$$\|\hat{x}\|_1 \leq \|x_0\|_1 \tag{3.2}$$

and

$$\||A\hat{x}| - |Ax_0|\|_2^2 \leq \epsilon^2. \tag{3.3}$$

Now the index set $\{1, 2, \ldots, m\}$ is divisible into two subsets

$$T = \{j : \ \text{sign}(\langle a_j, \hat{x}\rangle) = \text{sign}(\langle a_j, x_0\rangle)\},$$
$$T^c = \{j : \ \text{sign}(\langle a_j, \hat{x}\rangle) = -\text{sign}(\langle a_j, x_0\rangle)\}.$$

Then (3.3) implies that

$$\|A_T\hat{x} - A_T x_0\|_2^2 + \|A_{T^c}\hat{x} + A_{T^c}x_0\|_2^2 \leq \epsilon^2. \tag{3.4}$$

Here either $|T| \geq m/2$ or $|T^c| \geq m/2$. Without loss of generality we assume that $|T| \geq m/2$. We use the fact

$$\|A_T\hat{x} - A_T x_0\|_2^2 \leq \epsilon^2. \tag{3.5}$$

From (3.2) and (3.5) we obtain

$$\hat{x} \in \left\{x \in \mathbb{R}^N : \|x\|_1 \leq \|x_0\|_1, \|A_T x - A_T x_0\|_2 \leq \epsilon\right\}. \tag{3.6}$$

Recall that A satisfies S-RIP of order tk and constants $\theta_-, \ \theta_+$. Here

$$t \geq \max\{\frac{1}{2\theta_- - \theta_-^2}, \frac{1}{2\theta_+ - \theta_+^2}\} > 1. \tag{3.7}$$

The definition of S-RIP implies that A_T satisfies the RIP of order tk in which

$$\delta_{tk} \leq \max\{1 - \theta_-, \ \theta_+ - 1\} \leq \sqrt{\frac{t-1}{t}} \tag{3.8}$$

where the second inequality follows from (3.7). The combination of (3.6), (3.8) and Lemma 2.1 now implies

$$\|\hat{x} - x_0\|_2 \leq c_1\epsilon + c_2\frac{2\sigma_k(x_0)_1}{\sqrt{k}},$$

where c_1 and c_2 are defined in Lemma 2.1. If $|T^c| \geq \frac{m}{2}$ we get the corresponding result

$$\|\hat{x} + x_0\|_2 \leq c_1\epsilon + c_2\frac{2\sigma_k(x_0)_1}{\sqrt{k}}.$$

The theorem is now proved. □

This theorem demonstrates that, if the measurement matrix has the S-RIP, the real compressive phase retrieval problem can be solved stably by ℓ_1-minimization.

4 Phase Retrieval and Best k-term Approximation

4.1 Instance Optimality from the Linear Measurements

We introduce some definitions and results in [5]. Recall that for a given encoder matrix $A \in \mathbb{R}^{m\times N}$ and a decoder $\Delta : \mathbb{R}^m \mapsto \mathbb{R}^N$, the pair (A, Δ) is said to have instance optimality of order k with constant C_0 with respect to the norm X if

$$\|x - \Delta(Ax)\|_X \leq C_0\sigma_k(x)_X \tag{4.1}$$

holds for all $x \in \mathbb{R}^N$. Set $\mathcal{N}(A) := \{\eta \in \mathbb{R}^N : A\eta = 0\}$ to be the null space of A. The following theorem gives conditions under which the (4.1) holds.

Theorem 4.1 ([5]) *Let $A \in \mathbb{R}^{m\times N}$, $1 \leq k \leq N$ and $\|\cdot\|_X$ be a norm on $\mathbb{R}^N$. Then a sufficient condition for the existence of a decoder Δ satisfying (4.1) is*

$$\|\eta\|_X \leq \frac{C_0}{2}\sigma_{2k}(\eta)_X, \quad \forall \eta \in \mathcal{N}(A). \tag{4.2}$$

A necessary condition for the existence of a decoder Δ satisfying (4.1) is

$$\|\eta\|_X \leq C_0\sigma_{2k}(\eta)_X, \quad \forall \eta \in \mathcal{N}(A). \tag{4.3}$$

For the norm $X = \ell_1$ it was established in [5] that instance optimality of order k can indeed be achieved, e.g. for a Gaussian matrix A, with $m = O(k\log(N/k))$. The authors also considered more generally taking different norms on both sides of (4.1). Following [5], we say the pair (A, Δ) has (p, q)-*instance optimality of order k with constant C_0* if

$$\|x - \Delta(Ax)\|_p \leq C_0 k^{\frac{1}{q}-\frac{1}{p}}\sigma_k(x)_q, \quad \forall x \in \mathbb{R}^N, \tag{4.4}$$

with $1 \leq q \leq p \leq 2$. It was shown in [5] that the (p, q)-instance optimality of order k can be achieved at the cost of having $m = \mathcal{O}(k(N/k)^{2-2/q})\log(N/k)$ measurements.

4.2 Phaseless Instance Optimality

A natural question here is whether an analogous result to Theorem 4.1 exists for phaseless instance optimality defined in (1.3). We answer the question by presenting such a result in the case of real phase retrieval.

Recall that a pair (A, Δ) is said to be have the phaseless instance optimality of order k with constant C_0 for the norm $\|.\|_X$ if

$$\min\Big\{\|x - \Delta(|Ax|)\|_X, \|x + \Delta(|Ax|)\|_X\Big\} \leq C_0\sigma_k(x)_X \tag{4.5}$$

holds for all $x \in \mathbb{R}^N$.

Theorem 4.2 *Let $A \in \mathbb{R}^{m\times N}$, $1 \le k \le N$ and $\|\cdot\|_X$ be a norm. Then a sufficient condition for the existence of a decoder Δ satisfying the phaseless instance optimality (4.5) is: For any $I \subseteq \{1,\dots,m\}$ and $\eta_1 \in \mathcal{N}(A_I)$, $\eta_2 \in \mathcal{N}(A_{I^c})$ we have*

$$\min\{\|\eta_1\|_X, \|\eta_2\|_X\} \le \frac{C_0}{4}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{4}\sigma_k(\eta_1+\eta_2)_X. \tag{4.6}$$

A necessary condition for the existence of a decoder Δ satisfying (4.5) is: For any $I \subseteq \{1,\dots,m\}$ and $\eta_1 \in \mathcal{N}(A_I)$, $\eta_2 \in \mathcal{N}(A_{I^c})$ we have

$$\min\{\|\eta_1\|_X, \|\eta_2\|_X\} \le \frac{C_0}{2}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{2}\sigma_k(\eta_1+\eta_2)_X. \tag{4.7}$$

Proof We first assume (4.6) holds, and show that there exists a decoder Δ satisfying the phaseless instance optimality (4.5). To this end, we define a decoder Δ as follows:

$$\Delta(|Ax_0|) = \operatorname*{argmin}_{|Ax|=|Ax_0|} \sigma_k(x)_X.$$

Suppose $\hat{x} := \Delta(|Ax_0|)$. We have $|A\hat{x}| = |Ax_0|$ and $\sigma_k(\hat{x})_X \le \sigma_k(x_0)_X$. Note that $\langle a_j, \hat{x}\rangle = \pm\langle a_j, x_0\rangle$. Let $I \subseteq \{1,\dots,m\}$ be defined by

$$I = \Big\{ j : \ \langle a_j, \hat{x}\rangle = \langle a_j, x_0\rangle \Big\}.$$

Then

$$A_I(x_0-\hat{x}) = 0, \quad A_{I^c}(x_0+\hat{x}) = 0.$$

Set

$$\begin{aligned} \eta_1 &:= x_0 - \hat{x} \in \mathcal{N}(A_I),\\ \eta_2 &:= x_0 + \hat{x} \in \mathcal{N}(A_{I^c}). \end{aligned}$$

A simple observation yields

$$\sigma_k(\eta_1-\eta_2)_X = 2\sigma_k(\hat{x})_X \le 2\sigma_k(x_0)_X, \quad \sigma_k(\eta_1+\eta_2)_X = 2\sigma_k(x_0)_X. \tag{4.8}$$

Then (4.6) implies that

$$\begin{aligned} \min\{\|\hat{x}-x_0\|_X, \|\hat{x}+x_0\|_X\} &= \min\{\|\eta_1\|_X, \|\eta_2\|_X\}\\ &\le \frac{C_0}{4}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{4}\sigma_k(\eta_1+\eta_2)_X\\ &\le C_0\sigma_k(x_0)_X. \end{aligned}$$

Here the last equality is obtained by (4.8). This proves the sufficient condition.

We next turn to the necessary condition. Let Δ be a decoder for which the phaseless instance optimality (4.5) holds. Let $I \subseteq \{1, \dots, m\}$. For any $\eta_1 \in \mathcal{N}(A_I)$ and $\eta_2 \in \mathcal{N}(A_{I^c})$ we have

$$|A(\eta_1 + \eta_2)| = |A(\eta_1 - \eta_2)| = |A(\eta_2 - \eta_1)|. \tag{4.9}$$

The instance optimality implies

$$\begin{aligned} &\min\Big\{\|\Delta(|A(\eta_1+\eta_2)|) + \eta_1 + \eta_2\|_X, \|\Delta(|A(\eta_1+\eta_2)|) - (\eta_1+\eta_2)\|_X\Big\} \\ &\quad \le C_0 \sigma_k(\eta_1+\eta_2)_X. \end{aligned} \tag{4.10}$$

Without loss of generality we may assume that

$$\|\Delta(|A(\eta_1+\eta_2)|) + \eta_1 + \eta_2\|_X \;\le\; \|\Delta(|A(\eta_1+\eta_2)|) - (\eta_1+\eta_2)\|_X.$$

Then (4.10) implies that

$$\|\Delta(|A(\eta_1+\eta_2)|) + \eta_1 + \eta_2\|_X \le C_0 \sigma_k(\eta_1+\eta_2)_X. \tag{4.11}$$

By (4.9), we have

$$\begin{aligned} \|\Delta(|A(\eta_1+\eta_2)|) + \eta_1 + \eta_2\|_X &= \|\Delta(|A(\eta_2-\eta_1)|) - (\eta_2-\eta_1) + 2\eta_2\|_X \\ &\ge 2\|\eta_2\|_X - \|\Delta(|A(\eta_2-\eta_1)|) - (\eta_2-\eta_1)\|_X. \end{aligned} \tag{4.12}$$

Combining (4.11) and (4.12) yields

$$2\|\eta_2\|_X \le C_0 \sigma_k(\eta_1+\eta_2)_X + \|\Delta(|A(\eta_2-\eta_1)|) - (\eta_2-\eta_1)\|_X. \tag{4.13}$$

At the same time, (4.9) also implies

$$\begin{aligned} \|\Delta(|A(\eta_1+\eta_2)|) + \eta_1 + \eta_2\|_X &= \|\Delta(|A(\eta_2-\eta_1)|) + (\eta_2-\eta_1) + 2\eta_1\|_X \\ &\ge 2\|\eta_1\|_X - \|\Delta(|A(\eta_2-\eta_1)|) + (\eta_2-\eta_1)\|_X. \end{aligned} \tag{4.14}$$

Putting (4.11) and (4.14) together, we obtain

$$2\|\eta_1\|_X \le C_0 \sigma_k(\eta_1+\eta_2)_X + \|\Delta(|A(\eta_2-\eta_1)|) + (\eta_2-\eta_1)\|_X. \tag{4.15}$$

It follows from (4.13) and (4.15) that

$$\begin{aligned} \min\{\|\eta_1\|_X, \|\eta_2\|_X\} &\le \frac{C_0}{2}\sigma_k(\eta_1+\eta_2)_X \\ &\quad + \frac{1}{2}\min\{\|\Delta(|A(\eta_2-\eta_1)|) - (\eta_2-\eta_1)\|_X, \|\Delta(|A(\eta_2-\eta_1)|) \\ &\quad + (\eta_2-\eta_1)\|_X\} \le \frac{C_0}{2}\sigma_k(\eta_1+\eta_2)_X + \frac{C_0}{2}\sigma_k(\eta_1-\eta_2)_X. \end{aligned}$$

Here the last inequality is obtained by the instance optimality of (A, Δ). For the case where

$$\|\Delta(|A(\eta_1+\eta_2)|)-(\eta_1+\eta_2)\|_X \;\le\; \|\Delta(|A(\eta_1+\eta_2)|)+\eta_1+\eta_2\|_X,$$

we obtain

$$\min\{\|\eta_1\|_X, \|\eta_2\|_X\} \le \frac{C_0}{2}\sigma_k(\eta_1+\eta_2)_X + \frac{C_0}{2}\sigma_k(\eta_1-\eta_2)_X$$

via the same argument. The theorem is now proved. □

We next present a null space property for phaseless instance optimality, which allows us to establish parallel results for sparse phase retrieval.

Definition 4.1 We say a matrix $A \in \mathbb{R}^{m\times N}$ satisfies the *strong null space property (S-NSP) of order k with constant C* if for any index set $I \subseteq \{1,\dots,m\}$ with $|I| \ge m/2$ and $\eta \in \mathcal{N}(A_I)$ we have

$$\|\eta\|_X \le C\cdot\sigma_k(\eta)_X.$$

Theorem 4.3 *Assume that a matrix $A \in \mathbb{R}^{m\times N}$ has the strong null space property of order $2k$ with constant $C_0/2$. Then there must exist a decoder Δ having the phaseless instance optimality (1.3) with constant C_0. In particular, one such decoder is*

$$\Delta(|Ax_0|) = \underset{|Ax|=|Ax_0|}{\operatorname{argmin}} \sigma_k(x)_X.$$

Proof Assume that $I \subseteq \{1,\dots,m\}$. For any $\eta_1 \in \mathcal{N}(A_I)$ and $\eta_2 \in \mathcal{N}(A_{I^c})$ we must have either $\|\eta_1\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_1)_X$ or $\|\eta_2\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_2)_X$ by the strong null space property. If $\|\eta_1\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_1)_X$ then

$$\|\eta_1\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_1)_X \le \frac{C_0}{4}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{4}\sigma_k(\eta_1+\eta_2)_X.$$

Similarly if $\|\eta_2\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_2)_X$ we will have

$$\|\eta_2\|_X \le \frac{C_0}{2}\sigma_{2k}(\eta_2)_X \le \frac{C_0}{4}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{4}\sigma_k(\eta_1+\eta_2)_X.$$

It follows that

$$\min\{\|\eta_1\|_X, \|\eta_2\|_X\} \le \frac{C_0}{4}\sigma_k(\eta_1-\eta_2)_X + \frac{C_0}{4}\sigma_k(\eta_1+\eta_2)_X. \tag{4.16}$$

Theorem 4.2 now implies that the required decoder Δ exists. Furthermore, by the proof of the sufficiency part of Theorem 4.2,

$$\Delta(|Ax_0|) = \operatorname*{argmin}_{|Ax|=|Ax_0|} \sigma_k(x)_X$$

is one such decoder. □

4.3 The Case $X = \ell_1$

We will now apply Theorem 4.3 to the ℓ_1-norm case. The following lemma establishes a relation between S-RIP and S-NSP for the ℓ_1-norm.

Lemma 4.1 *Let a, b, k be integers. Assume that $A \in \mathbb{R}^{m\times N}$ satisfies the S-RIP of order $(a+b)k$ with constants $\theta_-,\ \theta_+ \in (0,2)$. Then A satisfies the S-NSP of order ak under the ℓ_1-norm with constant*

$$C_0 = 1 + \sqrt{\frac{a(1+\delta)}{b(1-\delta)}},$$

where δ is the restricted isometry constant and $\delta := \max\{1-\theta_-, \theta_+ - 1\} < 1$.

We remark that the above lemma is the analogous to the following lemma providing a relationship between RIP and NSP, which was shown in [5]:

Lemma 4.2 ([5, Lemma 4.1]) *Let $a = l/k$, $b = l'/k$ where $l, l' \geq k$ are integers. Assume that $A \in \mathbb{R}^{m\times N}$ satisfies the RIP of order $(a+b)k$ with $\delta = \delta_{(a+b)k} < 1$. Then A satisfies the null space property under the ℓ_1-norm of order ak with constant $C_0 = 1 + \frac{\sqrt{a(1+\delta)}}{\sqrt{b(1-\delta)}}$.*

Proof By the definition of S-RIP, for any index set $I \subseteq \{1,\dots,m\}$ with $|I| \geq m/2$, the matrix $A_I \in \mathbb{R}^{|I|\times N}$ satisfies the RIP of order $(a+b)k$ with constant $\delta_{(a+b)k} = \delta := \max\{1-\theta_-, \theta_+ - 1\} < 1$. It follows from Lemma 4.2 that

$$\|\eta\|_1 \leq \left(1 + \sqrt{\frac{a(1+\delta)}{b(1-\delta)}}\right)\sigma_{ak}(\eta)_1$$

for all $\eta \in \mathcal{N}(A_I)$. This proves the lemma. □

Set $a = 2$ and $b = 1$ in Lemma 4.1 we infer that if A satisfies the S-RIP of order $3k$ with constants $\theta_-,\ \theta_+ \in (0,2)$, then A satisfies the S-NSP of order $2k$ under the ℓ_1-norm with constant $C_0 = 1 + \sqrt{\frac{2(1+\delta)}{1-\delta}}$. Hence by Theorem 4.3, there must exist a decoder that has the instance optimality under the ℓ_1-norm with constant $2C_0$. According to Theorem 2.1, by taking $m = O(k\log(N/k))$ a Gaussian random matrix A satisfies S-RIP of order $3k$ with high probability. Hence, there exists a decoder Δ so that the pair (A, Δ) has the the ℓ_1-norm phaseless instance optimality at the cost of $m = O(k\log(N/k))$ measurements, as with the traditional instance optimality.

We are now ready to prove the following theorem on phaseless instance optimality under the ℓ_1-norm.

Theorem 4.4 *Let $A \in \mathbb{R}^{m\times N}$ satisfy the S-RIP of order tk with constants $0 < \theta_- < 1 < \theta_+ < 2$, where*

$$t \geq \max\left\{\frac{2}{\theta_-}, \frac{2}{2-\theta_+}\right\} > 2.$$

Let

$$\Delta(|Ax_0|) = \underset{x\in\mathbb{R}^N}{\operatorname{argmin}} \{\|x\|_1 : |Ax| = |Ax_0|\}. \tag{4.17}$$

Then (A, Δ) has the ℓ_1-norm phaseless instance optimality with constant $C = \frac{2C_0}{2-C_0}$, where $C_0 = 1 + \sqrt{\frac{1+\delta}{(t-1)(1-\delta)}}$ and as before

$$\delta := \max\{1-\theta_-, \theta_+ - 1\} \leq 1 - \frac{2}{t}.$$

Proof of Lemma 4.1 Let $x_0 \in \mathbb{R}^N$ and set $\hat{x} = \Delta(|Ax_0|)$. Then by definition

$$\|\hat{x}\|_1 \leq \|x_0\|_1 \quad \text{and} \quad |A\hat{x}| = |Ax_0|.$$

Denote by $I \subseteq \{1, \ldots, m\}$ the set of indices

$$I = \left\{j : \langle a_j, \hat{x}\rangle = \langle a_j, x_0\rangle\right\},$$

and thus $\langle a_j, \hat{x}\rangle = -\langle a_j, x_0\rangle$ for $j \in I^c$. It follows that

$$A_I(\hat{x} - x_0) = 0 \quad \text{and} \quad A_{I^c}(\hat{x} + x_0) = 0.$$

Set

$$\eta := \hat{x} - x_0 \in \mathcal{N}(A_I).$$

We know that A satisfies the S-RIP of order tk with constants $\theta_-,\ \theta_+$ where

$$t \geq \max\left\{\frac{2}{\theta_-}, \frac{2}{2-\theta_+}\right\} > 2.$$

For the case $|I| \geq m/2$, A_I satisfies the RIP of order tk with RIP constant

$$\delta = \delta_{tk} := \max\{1-\theta_-, \theta_+ - 1\} \leq 1 - \frac{2}{t} < 1.$$

Take $a := 1,\ b := t - 1$ in Lemma 4.1. Then A satisfies the ℓ_1-norm S-NSP of order k with constant

$$C_0 = 1 + \sqrt{\frac{1+\delta}{(t-1)(1-\delta)}} < 2.$$

This yields

$$\|\eta\|_1 \leq C_0 \|\eta_{T^c}\|_1, \tag{4.18}$$

where T is the index set for the k largest coefficients of x_0 in magnitude. Hence $\|\eta_T\|_1 \leq (C_0 - 1)\|\eta_{T^c}\|_1$. Since $\|\hat{x}\|_1 \leq \|x_0\|_1$ we have

$$\begin{aligned} \|x_0\|_1 \geq \|\hat{x}\|_1 &= \|x_0 + \eta\|_1 = \|x_{0,T} + x_{0,T^c} + \eta_T + \eta_{T^c}\|_1 \\ &\geq \|x_{0,T}\|_1 - \|x_{0,T^c}\|_1 + \|\eta_{T^c}\|_1 - \|\eta_T\|_1. \end{aligned}$$

It follows that

$$\|\eta_{T^c}\|_1 \leq \|\eta_T\|_1 + 2\sigma_k(x_0)_1 \leq (C_0 - 1)\|\eta_{T^c}\|_1 + 2\sigma_k(x_0)_1$$

and thus

$$\|\eta_{T^c}\|_1 \leq \frac{2}{2 - C_0}\sigma_k(x_0)_1.$$

Now (4.18) yields

$$\|\eta\|_1 \leq C_0\|\eta_{T^c}\|_1 \leq \frac{2C_0}{2 - C_0}\sigma_k(x_0)_1,$$

which implies

$$\|\hat{x} - x_0\|_1 \leq C_0\|\eta_{T^c}\|_1 \leq \frac{2C_0}{2 - C_0}\sigma_k(x_0)_1.$$

For the case $|I^c| \geq m/2$ identical argument yields

$$\|\hat{x} + x_0\|_1 \leq C_0\|\eta_{T^c}\|_1 \leq \frac{2C_0}{2 - C_0}\sigma_k(x_0)_1.$$

The theorem is now proved. □

By Theorem 2.1, an $m \times N$ random Gaussian matrix with $m = \mathcal{O}(tk \log(N/k))$ satisfies the S-RIP of order tk with high probability. We therefore conclude that the ℓ_1-norm phaseless instance optimality of order k can be achieved at the cost of $m = \mathcal{O}(tk \log(N/k))$ measurements.

4.4 Mixed-Norm phaseless Instance Optimality

We now consider *mixed-norm phaseless instance optimality*. Let $1 \leq q \leq p \leq 2$ and $s = 1/q - 1/p$. We seek estimates of the form

$$\min\{\|x - \Delta(|Ax|)\|_p, \|x + \Delta(|Ax|)\|_p\} \leq C_0 k^{-s} \sigma_k(x)_q \tag{4.19}$$

for all $x \in \mathbb{R}^N$. We shall prove both necessary and sufficient conditions for mixed-norm phaseless instance optimality.

Theorem 4.5 *Let $A \in \mathbb{R}^{m\times N}$ and $1 \leq q \leq p \leq 2$. Set $s = 1/q - 1/p$. Then a decoder Δ satisfying the mixed norm phaseless instance optimality (4.19) with constant C_0 exists if: for any index set $I \subseteq \{1, \ldots, m\}$ and any $\eta_1 \in \mathcal{N}(A_I)$, $\eta_2 \in \mathcal{N}(A_{I^c})$ we have*

$$\min\{\|\eta_1\|_p, \|\eta_2\|_p\} \leq \frac{C_0}{4} k^{-s} \Big(\sigma_k(\eta_1 - \eta_2)_q + \sigma_k(\eta_1 + \eta_2)_q\Big). \tag{4.20}$$

Conversely, assume a decoder Δ satisfying the mixed norm phaseless instance optimality (4.19) exists. Then for any index set $I \subseteq \{1, \ldots, m\}$ and any $\eta_1 \in \mathcal{N}(A_I)$, $\eta_2 \in \mathcal{N}(A_{I^c})$ we have

$$\min\{\|\eta_1\|_p, \|\eta_2\|_p\} \leq \frac{C_0}{2} k^{-s} \Big(\sigma_k(\eta_1 - \eta_2)_q + \sigma_k(\eta_1 + \eta_2)_q\Big).$$

Proof of Lemma 4.1 The proof is virtually identical to the proof of Theorem 4.2. We shall omit the details here in the interest of brevity. □

Definition 4.2 (*Mixed-Norm Strong Null Space Property*) We say that A has the mixed strong null space property in norms (ℓ_p, ℓ_q) of order k with constant C if for any index set $I \subseteq \{1, \ldots, m\}$ with $|I| \geq m/2$ the matrix $A_I \in \mathbb{R}^{|I|\times N}$ satisfies

$$\|\eta\|_p \leq C k^{-s} \sigma_k(\eta)_q$$

for all $\eta \in \mathcal{N}(A_I)$, where $s = 1/q - 1/p$.

The above is an extension of the standard definition of the mixed null space property of order k in norms (ℓ_p, ℓ_q) (see [5]) for a matrix A, which requires

$$\|\eta\|_p \leq C k^{-s} \sigma_k(\eta)_q$$

for all $\eta \in \mathcal{N}(A)$. We have the following straightforward generalization of Theorem 4.3.

Theorem 4.6 *Assume that $A \in \mathbb{R}^{m\times N}$ has the mixed strong null space property of order $2k$ in norms (ℓ_p, ℓ_q) with constant $C_0/2$, where $1 \leq q \leq p \leq 2$. Then there exists a decoder Δ such that the mixed-norm phaseless instance optimality (4.19) holds with constant C_0.*

We establish relationships between mixed-norm strong null space property and the S-RIP. First we present the following lemma that was proved in [5].

Lemma 4.3 ([5, Lemma 8.2]) *Let $k \geq 1$ and $\tilde{k} = \lceil k(\frac{N}{k})^{2-2/q} \rceil$. Assume that $A \in \mathbb{R}^{m\times N}$ satisfies the RIP of order $2k + \tilde{k}$ with $\delta := \delta_{2k+\tilde{k}} < 1$. Then A satisfies the mixed null space property in norms (ℓ_p, ℓ_q) of order $2k$ with constant $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$.*

Proposition 4.1 *Let $k \geq 1$ and $\tilde{k} = \lceil k(\frac{N}{k})^{2-2/q} \rceil$. Assume that $A \in \mathbb{R}^{m\times N}$ satisfies the S-RIP of order $2k + \tilde{k}$ with constants $0 < \theta_- < 1 < \theta_+ < 2$. Then A satisfies the mixed strong null space property in norms (ℓ_p, ℓ_q) of order $2k$ with constant $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$, where δ is the RIP constant and $\delta := \delta_{2k+\tilde{k}} = \max\{1 - \theta_-, \theta_+ - 1\}$.*

Proof of Lemma 4.1 By definition for any index set $I \subseteq \{1, \ldots, m\}$ with $|I| \geq m/2$, the matrix $A_I \in \mathbb{R}^{|I|\times N}$ satisfies RIP of order $2k + \tilde{k}$ with constant $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$, where δ is the RIP constant and $\delta := \delta_{2k+\tilde{k}} = \max\{1 - \theta_-, \theta_+ - 1\}$. By Lemma 4.3, we know that A_I satisfies the mixed null space property in norms (ℓ_p, ℓ_q) of order $2k$ with constant $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$, in other words for any $\eta \in \mathcal{N}(A_I)$,

$$\|\eta\|_p \leq Ck^{-s}\sigma_{2k}(\eta)_q.$$

So A satisfies the mixed strong null space property. □

Corollary 4.1 *Let $k \geq 1$ and $\tilde{k} = k(\frac{N}{k})^{2-2/q}$. Assume that $A \in \mathbb{R}^{m\times N}$ satisfies the S-RIP of order $2k + \tilde{k}$ with constants $0 < \theta_- < 1 < \theta_+ < 2$. Let $\delta := \delta_{2k+\tilde{k}} = \max\{1 - \theta_-, \theta_+ - 1\} < 1$. Define the decoder Δ for A by*

$$\Delta(|Ax_0|) = \underset{|Ax|=|Ax_0|}{\operatorname{argmin}} \sigma_k(x)_q. \tag{4.21}$$

Then (4.19) holds with constant $2C_0$, where $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$.

Proof of Lemma 4.1 By the Proposition 4.1, the matrix A satisfies the mixed strong null space property in (ℓ_p, ℓ_q) of order $2k$ with constant $C_0 = 2^{1/p+1/2}\sqrt{\frac{1+\delta}{1-\delta}} + 2^{1/p-1/q}$. The corollary now follows immediately from Theorem 4.6. □

Remark 4.1 Combining Theorem 2.1 and Corollary 4.1, the mixed phaseless instance optimality of order k in norms (ℓ_p, ℓ_q) can be achieved for the price of $\mathcal{O}(k(N/k)^{2-2/q}\log(N/k))$ measurements, just as with the traditional mixed (ℓ_p, ℓ_q)-norm instance optimality. Theorem 3.1 implies that the ℓ_1 decoder satisfies the $(p, q) = (2, 1)$ mixed-norm phaseless instance optimality at the price of $\mathcal{O}(k\log(N/k))$ measurements.

Appendix: Proof of Lemma 2.1

We will first need the following two Lemmas to prove Lemma 2.1.

Lemma 5.1 (Sparse Representation of a Polytope [2,12]) *Let $s \geq 1$ and $\alpha > 0$. Set*

$$T(\alpha, s) := \left\{ u \in \mathbb{R}^n : \|u\|_\infty \leq \alpha,\ \|u\|_1 \leq s\alpha \right\}.$$

For any $v \in \mathbb{R}^n$ let

$$U(\alpha, s, v) := \left\{ u \in \mathbb{R}^n : supp(u) \subseteq supp(v), \|u\|_0 \leq s, \|u\|_1 = \|v\|_1, \|u\|_\infty \leq \alpha \right\}.$$

Then $v \in T(\alpha, s)$ if and only if v is in the convex hull of $U(\alpha, s, v)$, i.e. v can be expressed as a convex combination of some $u_1, \ldots, u_N$ in $U(\alpha, s, v)$.

Lemma 5.2 ([1, Lemma 5.3]) *Assume that $a_1 \geq a_2 \geq \cdots \geq a_m \geq 0$. Let $r \leq m$ and $\lambda \geq 0$ such that $\sum_{i=1}^r a_i + \lambda \geq \sum_{i=r+1}^m a_i$. Then for all $\alpha \geq 1$ we have*

$$\sum_{j=r+1}^{m} a_j^\alpha \leq r \left(\sqrt[\alpha]{\frac{\sum_{i=1}^r a_i^\alpha}{r}} + \frac{\lambda}{r} \right)^\alpha. \tag{5.1}$$

In particular for $\lambda = 0$ we have

$$\sum_{j=r+1}^{m} a_j^\alpha \leq \sum_{i=1}^{r} a_i^\alpha.$$

We are now ready to prove Lemma 2.1.

Proof Set $h := \hat{x} - x_0$. Let T_0 denote the set of the largest k coefficients of x_0 in magnitude. Then

$$\begin{aligned} \|x_0\|_1 + \rho \geq \|\hat{x}\|_1 &= \|x_0 + h\|_1 \\ &= \|x_{0,T_0} + h_{T_0} + x_{0,T_0^c} + h_{T_0^c}\|_1 \\ &\geq \|x_{0,T_0}\|_1 - \|h_{T_0}\|_1 - \|x_{0,T_0^c}\|_1 + \|h_{T_0^c}\|_1. \end{aligned}$$

It follows that

$$\begin{aligned} \|h_{T_0^c}\|_1 &\leq \|h_{T_0}\|_1 + 2\|x_{0,T_0^c}\|_1 + \rho \\ &= \|h_{T_0}\|_1 + 2\sigma_k(x_0)_1 + \rho. \end{aligned}$$

Suppose that S_0 is the index set of the k largest entries in absolute value of h. Then we can get

$$\begin{aligned}\|h_{S_0^c}\|_1 \le \|h_{T_0^c}\|_1 &\le \|h_{T_0}\|_1 + 2\sigma_k(x_0)_1 + \rho \\ &\le \|h_{S_0}\|_1 + 2\sigma_k(x_0)_1 + \rho.\end{aligned}$$

Set

$$\alpha := \frac{\|h_{S_0}\|_1 + 2\sigma_k(x_0)_1 + \rho}{k}.$$

We divide $h_{S_0^c}$ into two parts $h_{S_0^c} = h^{(1)} + h^{(2)}$, where

$$h^{(1)} := h_{S_0^c} \cdot I_{\{i:\,|h_{S_0^c}(i)|>\alpha/(t-1)\}}, \quad h^{(2)} := h_{S_0^c} \cdot I_{\{i:\,|h_{S_0^c}(i)|\le\alpha/(t-1)\}}.$$

A simple observation is that $\|h^{(1)}\|_1 \le \|h_{S_0^c}\|_1 \le \alpha k$. Set

$$\ell := |\mathrm{supp}(h^{(1)})| = \|h^{(1)}\|_0.$$

Since all non-zero entries of $h^{(1)}$ have magnitude larger than $\alpha/(t-1)$, we have

$$\alpha k \ge \|h^{(1)}\|_1 = \sum_{i\in\mathrm{supp}(h^{(1)})} |h^{(1)}(i)| \ge \sum_{i\in\mathrm{supp}(h^{(1)})} \frac{\alpha}{t-1} = \frac{\alpha\ell}{t-1},$$

which implies $\ell \le (t-1)k$. Thus we have:

$$\left\langle A(h_{S_0} + h^{(1)}), Ah\right\rangle \le \|A(h_{S_0} + h^{(1)})\|_2 \cdot \|Ah\|_2 \le \sqrt{1+\delta} \cdot \|h_{S_0} + h^{(1)}\|_2 \cdot \epsilon. \tag{5.2}$$

Here we apply the facts that $\|h_{S_0} + h^{(1)}\|_0 = \ell + k \le tk$ and A satisfies the RIP of order tk with $\delta := \delta_{tk}^A$. We shall assume at first that tk as an integer. Note that $\|h^{(2)}\|_\infty \le \frac{\alpha}{t-1}$ and

$$\|h^{(2)}\|_1 = \|h_{S_0^c}\|_1 - \|h^{(1)}\|_1 \le k\alpha - \frac{\alpha\ell}{t-1} = (k(t-1)-\ell)\frac{\alpha}{t-1}. \tag{5.3}$$

We take $s := k(t-1) - \ell$ in Lemma 5.1 and obtain that $h^{(2)}$ is a weighted mean

$$h^{(2)} = \sum_{i=1}^N \lambda_i u_i, \qquad 0 \le \lambda_i \le 1, \quad \sum_{i=1}^N \lambda_i = 1$$

where $\|u_i\|_0 \le k(t-1)-\ell$, $\|u_i\|_1 = \|h^{(2)}\|_1$, $\|u_i\|_\infty \le \alpha/(t-1)$ and $\mathrm{supp}(u_i) \subseteq \mathrm{supp}(h^{(2)})$. Hence

$$\begin{aligned}\|u_i\|_2 \le \sqrt{\|u_i\|_0}\cdot\|u_i\|_\infty &= \sqrt{k(t-1)-\ell}\cdot\|u_i\|_\infty\\ &\le \sqrt{k(t-1)}\cdot\|u_i\|_\infty\\ &\le \alpha\sqrt{k/(t-1)}.\end{aligned}$$

Now for $0 \le \mu \le 1$ and $d \ge 0$, which will be chosen later, set

$$\beta_j := h_{S_0} + h^{(1)} + \mu\cdot u_j, \quad j = 1,\dots,N.$$

Then for fixed $i \in [1, N]$

$$\begin{aligned}\sum_{j=1}^N \lambda_j\beta_j - d\beta_i &= h_{S_0} + h^{(1)} + \mu\cdot h^{(2)} - d\beta_i\\ &= (1-\mu-d)(h_{S_0}+h^{(1)}) - d\mu u_i + \mu h.\end{aligned}$$

Recall that $\alpha = \frac{\|h_{S_0}\|_1 + 2\sigma_k(x_0)_1 + \rho}{k}$. Thus

$$\begin{aligned}\|u_i\|_2 &\le \sqrt{k/(t-1)}\alpha\\ &\le \frac{\|h_{S_0}\|_2}{\sqrt{t-1}} + \frac{2\sigma_k(x_0)_1+\rho}{\sqrt{k(t-1)}}\\ &\le \frac{\|h_{S_0}+h^{(1)}\|_2}{\sqrt{t-1}} + \frac{2\sigma_k(x_0)_1+\rho}{\sqrt{k(t-1)}}\\ &= \frac{z+R}{\sqrt{t-1}},\end{aligned} \tag{5.4}$$

where $z := \|h_{S_0} + h^{(1)}\|_2$ and $R := \frac{2\sigma_k(x_0)_1+\rho}{\sqrt{k}}$. It is easy to check the following identity:

$$\begin{aligned}&(2d-1)\sum_{1\le i<j\le N}\lambda_i\lambda_j\|A(\beta_i-\beta_j)\|_2^2\\ &\quad= \sum_{i=1}^N\lambda_i\Big\|A\Big(\sum_{j=1}^N\lambda_j\beta_j - d\beta_i\Big)\Big\|_2^2 - \sum_{i=1}^N\lambda_i(1-d)^2\|A\beta_i\|_2^2,\end{aligned} \tag{5.5}$$

provided that $\sum_{i=1}^N\lambda_i = 1$. Choose $d = 1/2$ in (5.5) we then have

$$\sum_{i=1}^N\lambda_i\Big\|A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i + \mu h\Big)\Big\|_2^2 - \sum_{i=1}^N\frac{\lambda_i}{4}\|A\beta_i\|_2^2 = 0.$$

Note that for $d = 1/2$,

$$\begin{aligned}
&\left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i + \mu h\Big)\right\|_2^2 \\
&\quad = \left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\Big)\right\|_2^2 \\
&\qquad + 2\Big\langle A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\Big), \mu Ah\Big\rangle + \mu^2\|Ah\|_2^2.
\end{aligned}$$

It follows from $\sum_{i=1}^N \lambda_i = 1$ and $h^{(2)} = \sum_{i=1}^N \lambda_i u_i$ that

$$\begin{aligned}
&\sum_{i=1}^N \lambda_i \left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i + \mu h\Big)\right\|_2^2 \\
&= \sum_i \lambda_i \left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\Big)\right\|_2^2 \\
&\quad + 2\Big\langle A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}h^{(2)}\Big), \mu Ah\Big\rangle + \mu^2\|Ah\|_2^2 \\
&= \sum_i \lambda_i \left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\Big)\right\|_2^2 \\
&\quad + \mu(1-\mu)\Big\langle A(h_{S_0}+h^{(1)}), Ah\Big\rangle - \sum_{i=1}^N \frac{\lambda_i}{4}\|A\beta_i\|_2^2. \qquad (5.6)
\end{aligned}$$

Set $\mu = \sqrt{t(t-1)} - (t-1)$. We next estimate the three terms in (5.6). Noting that $\|h_{S_0}\|_0 \le k$, $\|h^{(1)}\|_0 \le \ell$ and $\|u_i\|_0 \le s = k(t-1) - \ell$, we obtain

$$\|\beta_i\|_0 \le \|h_{S_0}\|_0 + \|h^{(1)}\|_0 + \|u_i\|_0 \le t \cdot k$$

and $\|(\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\|_0 \le t \cdot k$. Since A satisfies the RIP of order $t \cdot k$ with δ, we have

$$\begin{aligned}
&\left\| A\Big((\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\Big)\right\|_2^2 \\
&\quad \le (1+\delta)\|(\frac{1}{2}-\mu)(h_{S_0}+h^{(1)}) - \frac{\mu}{2}u_i\|_2^2 \\
&\quad = (1+\delta)\Big((\frac{1}{2}-\mu)^2\|(h_{S_0}+h^{(1)})\|_2^2 + \frac{\mu^2}{4}\|u_i\|_2^2\Big) \\
&\quad = (1+\delta)\Big((\frac{1}{2}-\mu)^2 z^2 + \frac{\mu^2}{4}\|u_i\|_2^2\Big)
\end{aligned}$$

and

$$\begin{aligned}
\|A\beta_i\|_2^2 &\ge (1-\delta)\|\beta_i\|_2^2 = (1-\delta)(\|h_{S_0}+h^{(1)}\|_2^2 + \mu^2 \cdot \|u_i\|_2^2) \\
&= (1-\delta)(z^2 + \mu^2 \cdot \|u_i\|_2^2).
\end{aligned}$$

Combining the result above with (5.2) and (5.4) we get

$$
\begin{aligned}
0 &\le (1+\delta)\sum_{i=1}^{N}\lambda_i\Big((\frac{1}{2}-\mu)^2 z^2+\frac{\mu^2}{4}\|u_i\|_2^2\Big)+\mu(1-\mu)\sqrt{1+\delta}\cdot z\cdot\epsilon\\
&\quad-(1-\delta)\sum_{i=1}^{N}\frac{\lambda_i}{4}(z^2+\mu^2\|u_i\|_2^2)\\
&=\sum_{i=1}^{N}\lambda_i\Big(\Big((1+\delta)(\frac{1}{2}-\mu)^2-\frac{1-\delta}{4}\Big)z^2+\frac{\delta}{2}\mu^2\|u_i\|_2^2\Big)+\mu(1-\mu)\sqrt{1+\delta}\cdot z\cdot\epsilon\\
&\le\sum_{i=1}^{N}\lambda_i\Big(\Big((1+\delta)(\frac{1}{2}-\mu)^2-\frac{1-\delta}{4}\Big)z^2+\frac{\delta}{2}\mu^2\frac{(z+R)^2}{t-1}\Big)\\
&\quad+\mu(1-\mu)\sqrt{1+\delta}\cdot z\cdot\epsilon\\
&=\Big((\mu^2-\mu)+\delta\Big(\frac{1}{2}-\mu+(1+\frac{1}{2(t-1)})\mu^2\Big)\Big)z^2\\
&\quad+\Big(\mu(1-\mu)\sqrt{1+\delta}\cdot\epsilon+\frac{\delta\mu^2 R}{t-1}\Big)z+\frac{\delta\mu^2R^2}{2(t-1)}\\
&=-t\Big((2t-1)-2\sqrt{t(t-1)}\Big)(\sqrt{\frac{t-1}{t}}-\delta)z^2\\
&\quad+\Big(\mu^2\sqrt{\frac{t}{t-1}}\sqrt{1+\delta}\cdot\epsilon+\frac{\delta\mu^2R}{t-1}\Big)z+\frac{\delta\mu^2R^2}{2(t-1)}\\
&=\frac{\mu^2}{t-1}\Big(-t(\sqrt{\frac{t-1}{t}}-\delta)z^2+(\sqrt{t(t-1)(1+\delta)}\epsilon+\delta R)z+\frac{\delta R^2}{2}\Big),
\end{aligned}
$$

which is a quadratic inequality for z. We know $\delta<\sqrt{(t-1)/t}$. So by solving the above inequality we get

$$
\begin{aligned}
z &\le \frac{(\sqrt{t(t-1)(1+\delta)}\epsilon+\delta R)+\big((\sqrt{t(t-1)(1+\delta)}\epsilon+\delta R)^2+2t(\sqrt{(t-1)/t}-\delta)\delta R^2\big)^{1/2}}{2t(\sqrt{(t-1/t)}-\delta)}\\
&\le\frac{\sqrt{t(t-1)(1+\delta)}}{t(\sqrt{(t-1)/t}-\delta)}\epsilon+\frac{2\delta+\sqrt{2t(\sqrt{(t-1)/t}-\delta)\delta}}{2t(\sqrt{(t-1)/t}-\delta)}R.
\end{aligned}
$$

Finally, noting that $\|h_{S_0^c}\|_1\le\|h_{S_0}\|_1+R\sqrt{k}$, in the Lemma 5.2, if we set $m=N$, $r=k$, $\lambda=R\sqrt{k}\ge 0$ and $\alpha=2$ then $\|h_{S_0^c}\|_2\le\|h_{S_0}\|_2+R$. Hence

$$
\begin{aligned}
\|h\|_2&=\sqrt{\|h_{S_0}\|_2^2+\|h_{S_0^c}\|_2^2}\\
&\le\sqrt{\|h_{S_0}\|_2^2+(\|h_{S_0}\|_2+R)^2}
\end{aligned}
$$

$$\leq \sqrt{2\|h_{S_0}\|_2^2} + R \leq \sqrt{2}z + R$$
$$\leq \frac{\sqrt{2(1+\delta)}}{1-\sqrt{t/(t-1)}\delta}\epsilon + \left(\frac{\sqrt{2}\delta + \sqrt{t(\sqrt{(t-1)/t}-\delta)\delta}}{t(\sqrt{(t-1)/t}-\delta)} + 1\right)R.$$

Substitute R into this inequality and the conclusion follows.

For the case where $t \cdot k$ is not an integer, we set $t^* := \lceil tk \rceil / k$, then $t^* > t$ and $\delta_{t^*k} = \delta_{tk} < \sqrt{\frac{t-1}{t}} < \sqrt{\frac{t^*-1}{t^*}}$. We can then prove the result by working on δ_{t^*k}. □

Acknowledgments Yang Wang was supported in part by the AFOSR grant FA9550-12-1-0455 and NSF grant IIS-1302285. Zhiqiang Xu was supported by NSFC grant (11171336, 11422113, 11021101, 11331012) and by National Basic Research Program of China (973 Program 2015CB856000).

References

1. Cai, T.T., Zhang, A.: Sharp RIP bound for sparse signal and low-rank matrix recovery. Appl. Comput. Harmon. Anal. **35**(1), 74–93 (2013)
2. Cai, T.T., Zhang, A.: Sparse representation of a polytope and recovery of sparse signals and low-rank matrices. IEEE Trans. Inf. Theory **60**(1), 122–132 (2014)
3. Candès, E.J., Tao, T.: Decoding by linear programming. IEEE Trans. Inf. Theory **51**(12), 4203–4215 (2005)
4. Candès, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
5. Cohen, A., Dahmen, W., Devore, R.: Compressed sensing and best k-term approximation. J. Am. Math. Soc. **22**(1), 211–231 (2009)
6. Eldar, Y.C., Mendelson, S.: Phase retrieval: stability and recovery guarantees. Appl. Comput. Harmon. Anal. **36**(3), 473–494 (2014)
7. Han, B., Xu, Z. Q.: Robustness properties of dimensionality reduction with Gaussian random matrices. arXiv:1501.01695 (2015)
8. Iwen, M., Viswanathan, A., Wang, Y.: Robust sparse phase retrieval made easy. arXiv:1410.5295 (2015)
9. Moravec, M.L., Romberg, J.K., Baraniuk, R.G.: Compressive phase retrieval. In: SPIE Proceedings, vol. 6701 (2007)
10. Voroninski, V., Xu, Z.Q.: A strong restricted isometry property, with an application to phaseless compressed sensing. Appl. Comput. Harmon. Anal. (2015). doi:10.1016/j.acha.2015.06.004
11. Wang, Y., Xu, Z.Q.: Phase retrieval for sparse signals. Appl. Comput. Harmon. Anal. **37**(3), 531–544 (2014)
12. Xu, G.W., Xu, Z.Q.: On the ℓ_1-sparse decomposition of signals. J. Oper. Res. Soc. China **1**(4), 537–541 (2013)

Journal of Functional Analysis 269 (2015) 1515–1538

Contents lists available at ScienceDirect

Journal of Functional Analysis

www.elsevier.com/locate/jfa

Gabor orthonormal bases generated by the unit cubes

Jean-Pierre Gabardo [a], Chun-Kit Lai [b,*], Yang Wang [c]

[a] *Department of Mathematics and Statistics, McMaster University, Hamilton, Ontario, L8S 4K1, Canada*
[b] *Department of Mathematics, San Francisco State University, 1600 Holloway Ave., San Francisco, CA 94132, United States*
[c] *Department of Mathematics, Hong Kong University of Science and Technology, Hong Kong*

ARTICLE INFO

Article history:
Received 26 November 2014
Accepted 10 June 2015
Available online 2 July 2015
Communicated by L. Gross

MSC:
42B05
42A85

Keywords:
Gabor orthonormal bases
Spectral sets
Translational tiles
Tiling sets

ABSTRACT

We consider the problem in determining the countable sets Λ in the time-frequency plane such that the Gabor system generated by the time-frequency shifts of the window $\chi_{[0,1]^d}$ associated with Λ forms a Gabor orthonormal basis for $L^2(\mathbb{R}^d)$. We show that, if this is the case, the translates by elements Λ of the unit cube in $\mathbb{R}^{2d}$ must tile the time-frequency space $\mathbb{R}^{2d}$. By studying the possible structure of such tiling sets, we completely classify all such admissible sets Λ of time-frequency shifts when $d = 1, 2$. Moreover, an inductive procedure for constructing such sets Λ in dimension $d \geq 3$ is also given. An interesting and surprising consequence of our results is the existence, for $d \geq 2$, of discrete sets Λ with $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ forming a Gabor orthonormal basis but with the associated "time"-translates of the window $\chi_{[0,1]^d}$ having significant overlaps.

* Corresponding author.
E-mail addresses: gabardo@mcmaster.ca (J.-P. Gabardo), cklai@sfsu.edu (C.-K. Lai), yangwang@ust.hk (Y. Wang).

http://dx.doi.org/10.1016/j.jfa.2015.06.004
0022-1236/

1. Introduction

Let g be a non-zero function in $L^2(\mathbb{R}^d)$ and let Λ be a discrete countable set on $\mathbb{R}^{2d}$, where we identify $\mathbb{R}^{2d}$ to the time-frequency plane by writing $(t,\lambda) \in \Lambda$ with $t, \lambda \in \mathbb{R}^d$. The Gabor system associated with the window g consists of the set of translates and modulates of g:

$$\mathcal{G}(g,\Lambda) = \{e^{2\pi i\langle \lambda, x\rangle} g(x-t) : (t,\lambda) \in \Lambda\}. \tag{1.1}$$

Such systems were first introduced by Gabor [5] who used them for applications in the theory of telecommunication, but there has been a more recent interest in using Gabor system to expand functions both from a theoretical and applied perspective. The branch of Fourier analysis dealing with Gabor systems is usually referred to as Gabor, or time-frequency, analysis. Gröchenig's monograph [6] provide an excellent and detailed exposition on this subject.

Recall that the Gabor system is a *frame* for $L^2(\mathbb{R}^d)$ if there exist constants $A, B > 0$ such that

$$A\|f\|^2 \leq \sum_{(t,\lambda)\in\Lambda} |\langle f, e^{2\pi i\langle \lambda,\cdot\rangle} g(\cdot - t)\rangle|^2 \leq B\|f\|^2, \quad f \in L^2(\mathbb{R}^d). \tag{1.2}$$

It is called an orthonormal basis for $L^2(\mathbb{R}^d)$ if it is complete and the elements of the system (1.1) are mutually orthogonal in $L^2(\mathbb{R}^d)$ and have norm 1, or, equivalently, $\|g\| = 1$ and $A = B = 1$ in (1.2). One of the fundamental problems in Gabor analysis is to classify the windows g and time-frequency sets Λ with the property that the associated Gabor system $\mathcal{G}(g,\Lambda)$ forms a (Gabor) frame or an orthonormal basis for $L^2(\mathbb{R}^d)$. This is of course a very difficult problem and only partial results are known. For example, to the best of our knowledge, the complete characterization of time-frequency sets Λ for which (1.1) is a frame for $L^2(\mathbb{R}^d)$ was only done when $g = e^{-\pi x^2}$, the Gaussian window. Lyubarskii, and Seip and Wallsten [15,17] showed that $\mathcal{G}(e^{-\pi x^2}, \Lambda)$ is a Gabor frame if and only if the lower Beurling density of Λ is strictly greater than 1. If we assume that Λ is a lattice of the form $a\mathbb{Z} \times b\mathbb{Z}$, then it is well known that $ab \leq 1$ is a necessary condition for (1.1) to form a frame for $L^2(\mathbb{R}^d)$. Gröchenig and Stöckler [7] showed that for totally positive functions, (1.1) is a frame if and only if $ab < 1$. If we consider $g = \chi_{[0,c)}$, the characteristic function of an interval, the associated characterization problem is known as the *abc-problem* in Gabor analysis. By rescaling, one may assume that $c = 1$. In that case, the famous Janssen tie showed that the structure of the set of couples (a,b) yielding a frame is very complicated [9,8]. A complete solution of the abc-problem was recently obtained by Dai and Sun [2].

In this paper, we focus our attention on Gabor system of the form (1.1) which yield orthonormal bases for $L^2(\mathbb{R}^d)$. Perhaps the most natural and simplest example of Gabor orthonormal basis is the system $\mathcal{G}(\chi_{[0,1]^d}, \mathbb{Z}^d \times \mathbb{Z}^d)$. The orthonormality property for this

system easily follows from that facts that the Euclidean space $\mathbb{R}^d$ can be partitioned by the $\mathbb{Z}^d$-translates of the hypercube $[0,1]^d$ and that the exponentials $e^{2\pi i\langle n,x\rangle}$ form an orthonormal basis for the space of square-integrable functions supported on any of these translated hypercubes. A direct generalization of this observation is the following:

Proposition 1.1. *Let $|g| = |K|^{-1/2}\chi_K$, where $|\cdot|$ denotes the Lebesgue measure, and $K \subset \mathbb{R}^d$ is measurable with finite Lebesgue measure. Suppose that:*

- *The translates of K by the discrete set $\mathcal{J}$ are pairwise a.e. disjoint and cover $\mathbb{R}^d$ up to a set of zero measure.*
- *For each $t \in \mathcal{J}$, the set of exponentials $\{e^{2\pi i\langle \lambda,x\rangle} : \lambda \in \Lambda_t\}$ is an orthonormal basis for $L^2(K)$.*

Let

$$\Lambda = \bigcup_{t\in\mathcal{J}} \{t\} \times \Lambda_t. \tag{1.3}$$

Then $\mathcal{G}(g,\Lambda)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^d)$.

Although its proof is straightforward and will be omitted (see also [14]), this proposition gives us a flexible way of constructing large families of Gabor orthonormal basis. The first condition above means that K is a *translational tile* (with $\mathcal{J}$ called an associated *tiling set*) and the second one that $L^2(K)$ admits an orthonormal basis of exponentials. If this last condition holds, K is called a *spectral set* (and each Λ_t is an associated *spectrum*). The connection between translational tiles and spectral sets is quite mysterious. They were in fact conjectured to be the same class of sets by Fuglede [3], but that statement was later disproved by Tao [18] and the exact relationship between the two classes remains unclear.

For the fixed window $g_d = \chi_{[0,1]^d}$, we call a countable set $\Lambda \subset \mathbb{R}^{2d}$ *standard* if it is of the form (1.3). Motivated by the complete solution to the *abc*-problem, our main objective in this paper is to characterize the discrete sets Λ (not necessarily lattices) with the property that the Gabor system $\mathcal{G}(g_d,\Lambda)$ is a Gabor orthonormal basis. First, by generalizing the notion of *orthogonal packing region* (see Section 2) in the work of Lagarias, Reeds and Wang [12] to the setting of Gabor systems, we deduce a general criterion for $\mathcal{G}(g_d,\Lambda)$ to be a Gabor orthonormal basis.

Theorem 1.2. *$\mathcal{G}(g_d,\Lambda)$ is a Gabor orthonormal basis if and only if $\mathcal{G}(g_d,\Lambda)$ is an orthogonal set and the translates of $[0,1]^{2d}$ by the elements of Λ tile $\mathbb{R}^{2d}$.*

This criterion offers a very simple solution to our problem in the one-dimensional case.

Theorem 1.3. *In dimension $d = 1$, the system $\mathcal{G}(g_1, \Lambda)$ is a Gabor orthonormal basis if and only if Λ is standard.*

However, such a simple characterization ceases to exist in higher dimensions. We will introduce an inductive procedure which allows us to construct a Gabor orthonormal basis with window g_d from a Gabor orthonormal basis with window g_n, $n < d$. This procedure can be used to produce many non-standard Gabor orthonormal basis and we call a set Λ obtained through this procedure *pseudo-standard*. Assuming a mild condition on a low-dimensional time-frequency space, we show that $\mathcal{G}(g_d, \Lambda)$ are essentially pseudo-standard (see Theorem 3.6).

Although we do not have a complete description of the sets Λ yielding Gabor orthonormal bases with window g_d in dimension $d \geq 3$, we managed to obtain a complete characterization of those discrete sets $\Lambda \subset \mathbb{R}^4$ such that $\mathcal{G}(g_2, \Lambda)$ form an orthonormal basis for $L^2(\mathbb{R}^2)$.

Theorem 1.4. *$\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^2)$ if and only if we can partition $\mathbb{Z}$ into $\mathcal{J}$ and $\mathcal{J}'$ such that either*

$$\Lambda = \bigcup_{n\in\mathcal{J}} \{(m + t_{n,k}, n, j + \mu_{k,m,n}, k + \nu_n) : m, j, k \in \mathbb{Z}\} \cup \bigcup_{m\in\mathbb{Z}} \bigcup_{n\in\mathcal{J}'} \{(m + t_n, n)\} \times \Lambda_{m,n}$$

or

$$\Lambda = \bigcup_{m\in\mathcal{J}} \{(m, n + t_{m,j}, j + \nu_m, k + \mu_{j,m,n}) : n, j, k \in \mathbb{Z}\} \cup \bigcup_{n\in\mathbb{Z}} \bigcup_{m\in\mathcal{J}'} \{(m, n + t_m)\} \times \Lambda_{m,n},$$

where $\Lambda_{m,n} + [0,1]^2$ tile $\mathbb{R}^2$ and $t_{n,k}$, $\mu_{k,m,n}$ and ν_n are real numbers in $[0,1)$ as a function of m, n or k. See Fig. 1.

We organize the paper as follows. In Section 2, we provide some preliminaries notations and prove Theorem 1.2. In Section 3, we prove Theorem 1.3 and introduce the pseudo-standard time-frequency set. In the last section, we focus on dimension 2 and prove Theorem 1.4.

2. Preliminaries

In this section, we explore the relationship between Gabor orthonormal bases and tilings in the time-frequency space. This theory will be an extension of spectral-tile duality in [12] to the setting of Gabor analysis. Denote by $|K|$ the Lebesgue measure of

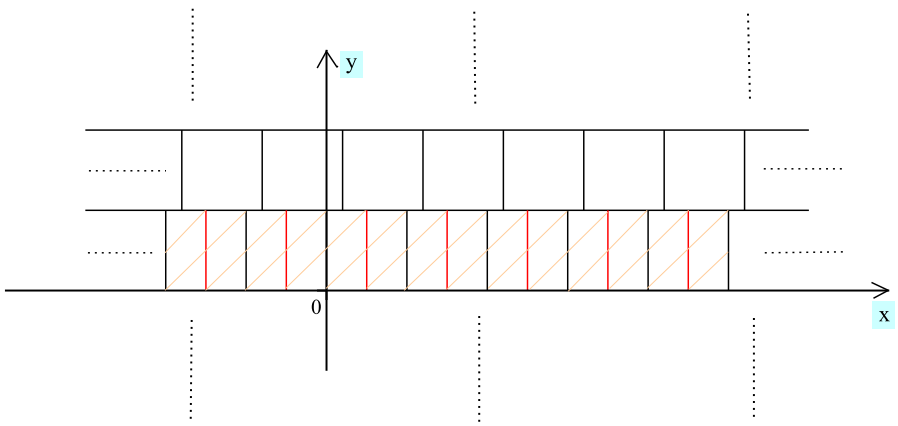


Fig. 1. This figure illustrates the time-domain of Λ in the first situation of Theorem 1.4. We basically partition $\mathbb{R}^2$ by horizontal strips. Some strips, like $\mathbb{R} \times [0,1]$ with $n = 0$, have overlapping structure. This corresponds to the first union of Λ. Some strips, like $\mathbb{R} \times [1,2]$ with $n = 1$, have tiling structures. This corresponds to the second union of Λ.

a set K. We say that a closed set T is a *region* if $|\partial T| = 0$ and $\overline{T^o} = T$. A bounded region T is called a *translational tile* if we can find a countable set $\mathcal{J}$ such that

(1) $|(T+t) \cap (T+t')| = 0$, $t, t' \in \mathcal{J}$, $t \neq t'$, and
(2) $\bigcup_{t\in\mathcal{J}} (T+t) = \mathbb{R}^d$.

In that case, $\mathcal{J}$ is called a *tiling set* for T and $T+\mathcal{J}$ a tiling of $\mathbb{R}^d$. We will say that $T+\mathcal{J}$ is a packing of $\mathbb{R}^n$ if (1) above is satisfied. We can generalize the notion of tiling and packing to measures and functions. Given a positive Borel measure μ and $f \in L^1(\mathbb{R}^n)$ with $f \geq 0$, the convolution of f and μ is defined to be

$$f * \mu(x) = \int f(x-y)\, d\mu(y), \quad x \in \mathbb{R}^n$$

(where a Borel measurable function is chosen in the equivalence class of f to define the integral above). We say that $f + \mu$ *is a tiling (resp. packing) of* $\mathbb{R}^d$ if $f * \mu = 1$ (resp. $f * \mu \leq 1$) almost everywhere with respect to the Lebesgue measure. It is clear that if $f = \chi_T$ and $\mu = \delta_{\mathcal{J}}$ where $\delta_{\mathcal{J}} = \sum_{t\in\mathcal{J}} \delta_t$, then $f * \mu = 1$ is equivalent to $T + \mathcal{J}$ being a tiling.

First, we start with the following theorem which gives us a very useful criterion to decide if a packing is actually a tiling. In fact, special cases of this theorem were proved by many different authors in different settings (see e.g. [12, Theorem 3.1], [11, Lemma 3.1] and [13]), but the following version is the most general one as far as we know.

Theorem 2.1. *Suppose that* $F, G \in L^1(\mathbb{R}^n)$ *are two functions with* $F, G \geq 0$ *and* $\int_{\mathbb{R}^n} F(x)\, dx = \int_{\mathbb{R}^n} G(x)\, dx = 1$. *Suppose that* μ *is a positive Borel measure on* $\mathbb{R}^n$ *such that*

$$F * \mu \leq 1 \quad \textit{and} \quad G * \mu \leq 1.$$

Then, $F * \mu = 1$ *if and only if* $G * \mu = 1$.

Proof. By symmetry, it suffices to prove one side of the equivalence. Assuming that $F * \mu = 1$, we have

$$1 = F * \mu \quad \Rightarrow \quad 1 = 1 * G = G * F * \mu = F * G * \mu.$$

Letting $H = G * \mu$ we have $0 \leq H \leq 1$ and $H * F = 1$. We now show that $H = 1$. Indeed letting A be the set $\{x \in \mathbb{R}^n, H(x) < 1\}$ and $B = \mathbb{R}^n \setminus A$, we have

$$(H * F)(x) = \int_{\mathbb{R}^n} H(y)\, F(x-y)\, dy = \int_A H(y)\, F(x-y)\, dy + \int_B H(y)\, F(x-y)\, dy.$$

Now, if $|A| > 0$, we have

$$\int_{\mathbb{R}^n} \int_A F(x-y)\, dy\, dx = |A| > 0$$

and there exists thus a set E with positive measure such that

$$\int_A F(x-y)\, dy > 0, \quad x \in E.$$

If $x \in E$, we have

$$\int_A H(y)\, F(x-y)\, dy + \int_B H(y)\, F(x-y)\, dy < \int_A F(x-y)\, dy + \int_B F(x-y)\, dy$$
$$= (1 * F)(x) = 1.$$

This contradicts to the fact that $H * F = 1$ almost everywhere. Hence, $|A| = 0$ and $H = 1$ follow. □

Let $f, g \in L^2(\mathbb{R}^d)$. We define the *short time Fourier transform* of f with respect to the window g be

$$V_g f(t, \nu) = \int_{\mathbb{R}^{2d}} f(x)\, \overline{g(x-t)}\, e^{-2\pi i \langle \nu, x \rangle}\, dx.$$

Let $\mathcal{G}(g, \Lambda)$ be a Gabor orthonormal basis. Since translating Λ be an element of $\mathbb{R}^{2d}$ does not affect the orthonormality nor the completeness of the given system, there is no loss of generality in assuming that $(0,0) \in \Lambda$. We say that a region D ($\subset \mathbb{R}^{2d}$) is an *orthogonal packing region* for g if

$$(D^\circ - D^\circ) \cap \mathcal{Z}(V_g g) = \emptyset.$$

Here $\mathcal{Z}(V_g g) = \{(t, \nu) : V_g g(t, \nu) = 0\}$.

Lemma 2.2. *Suppose that $\mathcal{G}(g,\Lambda)$ is a mutually orthogonal set of $L^2(\mathbb{R}^d)$. Let D be any orthogonal packing region for g. Then $\Lambda-\Lambda \subset \mathcal{Z}(V_g g)\cup\{0\}$ and $\Lambda+D$ is a packing of $\mathbb{R}^{2d}$. Suppose furthermore that $\mathcal{G}(g,\Lambda)$ is a Gabor orthonormal basis. Then $|D|\leq 1$.*

Proof. Let $(t,\lambda),(t',\lambda')\in\Lambda$ be two distinct points in Λ. Then

$$\int g(x-t')\,\overline{g(x-t)}\,e^{-2\pi i(\lambda-\lambda')x}\,dx=0,$$

or equivalently, after the change of variable $y=x-t'$,

$$\int g(x)\,\overline{g(x-(t-t'))}\,e^{-2\pi i(\lambda-\lambda')x}\,dx=0.$$

Hence, $V_g g(t-t',\lambda-\lambda')=0$ and $(t,\lambda)-(t',\lambda')\in\mathcal{Z}(V_g g)$. This means that $(t,\lambda)-(t',\lambda')\notin D^\circ-D^\circ$. Therefore, the intersection of the sets $(t,\lambda)+D$ and $(t',\lambda')+D$ has zero Lebesgue measure.

Suppose now that $\mathcal{G}(g,\Lambda)$ is a Gabor orthonormal basis. Denote by R the diameter of D. By the packing property of $\Lambda+D$,

$$\begin{aligned}|D|\cdot\frac{\#(\Lambda\cap[-T,T]^{2d})}{(2T)^{2d}}&=\frac{1}{(2T)^{2d}}\left|\bigcup_{\lambda\in\Lambda\cap[-T,T]^{2d}}(D+\lambda)\right|\\&\leq\frac{1}{(2T)^{2d}}\left|[-T-R,T+R]^{2d}\right|=(1+\frac{R}{T})^{2d}.\end{aligned}$$

Taking limit $T\to\infty$ and using the fact that Beurling density of Λ is 1 [16], we have $|D|\leq 1$. □

We say that an orthogonal packing region D for g is *tight* if we have furthermore $|D|=1$. We now apply Theorem 2.1 to the Gabor orthonormal basis problem.

Theorem 2.3. *Suppose that $\mathcal{G}(g,\Lambda)$ is an orthonormal set in $L^2(\mathbb{R}^d)$ and that D is a tight orthogonal packing region for g. Then $\mathcal{G}(g,\Lambda)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^d)$ if and only if $\Lambda+D$ is a tiling of $\mathbb{R}^{2d}$.*

Proof. Let $F=\chi_D$ and $G=|V_g f|^2/\|f\|_2^2$. Then $\int_{\mathbb{R}^{2d}}F=1$ and $\int_{\mathbb{R}^{2d}}G=\|g\|_2^2=1$. Now, as D is an orthogonal packing region for g, we have in particular

$$\sum_{\lambda\in\Lambda}\chi_D(x-\lambda)\leq 1.$$

This shows that

$$\delta_\Lambda * F=\delta_\Lambda*\chi_D\leq 1.$$

Moreover, $\Lambda + D$ is a tiling of $\mathbb{R}^{2d}$ if and only if $\delta_\Lambda * \chi_D = 1$. On the other hand, (g, Λ) being a mutually orthogonal set, Bessel's inequality yields

$$\sum_{(t,\lambda)\in\Lambda} \left| \int_{\mathbb{R}^d} f(x)\, \overline{g(x-t)}\, e^{-2\pi i\langle \lambda, x\rangle}\, dx \right|^2 \leq \|f\|^2, \quad f \in L^2(\mathbb{R}^d),$$

or, replacing f by $f(x-\tau)e^{2\pi i\nu x}$ with $(\tau,\nu) \in \mathbb{R}^{2d}$,

$$\sum_{(t,\lambda)\in\Lambda} |V_g f(\tau - t, \nu - \lambda)|^2 \leq \|f\|^2, \quad f \in L^2(\mathbb{R}^d).$$

Hence,

$$\delta_\Lambda * G = \delta_\Lambda * \frac{|V_g f|^2}{\|f\|^2} \leq 1$$

with equality if and only if the Gabor orthonormal system is in fact a basis. The conclusion follows then from Theorem 2.1. □

Proof of Theorem 1.2. Let $g_d = \chi_{[0,1]^d}$. Using Theorem 2.3, we just need to show that $[0,1]^{2d}$ is a tight orthogonal packing region for g_d.

We first consider the case $d = 1$. For $g_1 = \chi_{[0,1]}$, a direct computation shows that

$$V_{g_1} g_1(t,\nu) = \begin{cases} 0, & |t| \geq 1; \\ \frac{1}{2\pi i\nu}\left(e^{2\pi i\nu t} - e^{2\pi i\nu}\right), & 0 \leq t \leq 1; \\ \frac{1}{2\pi i\nu}\left(1 - e^{2\pi i\nu(t+1)}\right), & -1 \leq t \leq 0. \end{cases} \tag{2.1}$$

The zero set of $V_{g_1} g_1$ is therefore given by

$$\mathcal{Z}(V_{g_1} g_1) = \{(t,\nu) : |t| \geq 1\} \cup \{(t,\nu) : \nu(1-|t|) \in \mathbb{Z} \setminus \{0\}\}. \tag{2.2}$$

Hence, $(0,1)^2 - (0,1)^2 = (-1,1)^2$ does not intersect the zero set and therefore $[0,1]^2$ is a tight orthogonal packing region for g_1.

We now consider the case $d \geq 2$. As we can decompose g_d as $\chi_{[0,1]}(x_1)\ldots\chi_{[0,1]}(x_d)$, we have

$$V_{g_d} g_d(t,\nu) = V_{g_1} g_1(t_1,\nu_1)\ldots V_{g_1} g_1(t_d,\nu_d) \quad \text{where } t = (t_1,\ldots,t_d) \text{ and } \nu = (\nu_1,\ldots,\nu_d).$$

The zero set of $V_{g_d} g_d$ is therefore given by

$$\mathcal{Z}(V_{g_d} g_d) = \{(t,v) : |t|_{\max} \geq 1\} \cup \left(\bigcup_{i=1}^{d} \{(t,\nu) : \nu_i(1-|t_i|) \in \mathbb{Z} \setminus \{0\})\} \right) \tag{2.3}$$

where $|t|_{\max} = \max\{t_1, \ldots, t_d\}$. It follows that $[0,1]^{2d}$ is a tight orthogonal packing region for g_d. □

The following example will not be used in later discussion, but it demonstrates the usefulness of the theory for windows other than the unit cube.

Example 2.4. Let $g(x) = \frac{2}{e^{2x}+e^{-2x}}$ be the hyperbolic secant function. It can be shown ([10]; see also [4]) that

$$V_g g(t,\nu) = \frac{\pi \sin(\pi\nu t) e^{-\pi i \nu t}}{\sinh(2t)\sinh(\pi^2\nu/2)}$$

and the zero set is given by

$$\mathcal{Z}(V_g g) = \{(t,\nu) : t\nu \in \mathbb{Z} \setminus \{0\}\}.$$

Hence, $[0,1]^2$ is a tight orthogonal packing region for g. Note that the zero set does not contain any points on the x-axis and y-axis. There is no tiling set Λ for $[0,1]^2$ such that $\Lambda - \Lambda \subset \mathcal{Z}(V_g g) \cup \{0\}$ (see also Proposition 3.2 in the next section) and thus there is no Gabor orthonormal basis using the hyperbolic secant as a window. This can be viewed as a particular case of a version of the Balian–Low theorem valid for irregular Gabor frames which was recently obtained in [1] and which state that Gabor orthonormal bases cannot exist if the window function is in the modulation space $M^1(\mathbb{R}^d)$.

3. Gabor orthonormal bases

Using Lemma 2.2, Theorem 1.2 may be restated in the following way:

Theorem 3.1. *$\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis if and only if the inclusion $\Lambda - \Lambda \subset \mathcal{Z}(V_g g) \cup \{0\}$ holds and $\Lambda + [0,1]^{2d}$ is a tiling.*

In view of the previous result, the possible translational tilings of the unit cube on $\mathbb{R}^{2d}$ play a fundamental role in the solution of our problem. A characterization for these is not available in arbitrary $2d$ dimension but it is easily obtained when $d = 1$. We prove this result here for completeness but it should be well known.

Proposition 3.2. *Suppose that $\chi_{[0,1]^2} + \mathcal{J}$ is a tiling of $\mathbb{R}^2$ with $(0,0) \in \mathcal{J}$. Then $\mathcal{J}$ is of either of the following two forms:*

$$\mathcal{J} = \bigcup_{k\in\mathbb{Z}} (\mathbb{Z} + a_k) \times \{k\} \quad \text{or} \quad \mathcal{J} = \bigcup_{k\in\mathbb{Z}} \{k\} \times (\mathbb{Z} + a_k) \tag{3.1}$$

where a_k are any real numbers in $[0,1)$ for $k \neq 0$ and $a_0 = 0$.

Proof. By Keller's criterion for square tilings (see e.g. [12, Proposition 4.1]), for any (t_1, t_2) and (t_1', t_2') in $\mathcal{J}$, $t_i - t_i' \in \mathbb{Z} \setminus \{0\}$ for some $i = 1, 2$. Taking $(t_1', t_2') = (0, 0)$, we obtain that, for any $(t_1, t_2) \in \mathcal{J} \setminus \{(0, 0)\}$, one of t_1 or t_2 belongs to $\mathbb{Z} \setminus \{0\}$. If $\mathcal{J} \subset \mathbb{Z}$, we must have $\mathcal{J} = \mathbb{Z}$ for $\chi_{[0,1]^2} + \mathcal{J}$ to be tiling of $\mathbb{R}^2$ and $\mathbb{Z}$ can be written as either of the sets in (3.1) by taking $a_k = 0$ for all k. Suppose that there exists $(s_1, s_2) \in \mathcal{J}$ such that s_1 is not an integer and $s_2 \in \mathbb{Z}$. If $(t_1, t_2) \in \mathcal{J}$ and $t_2 \notin \mathbb{Z}$, then both t_1 and $t_1 - s_1$ must be integers which would imply that s_1 is an integer, contrary to our assumption. Hence, $(s_1, s_2) \in \mathcal{J}$ implies $s_2 \in \mathbb{Z}$ and we can write

$$\mathcal{J} = \bigcup_{k \in \mathbb{Z}} \mathcal{J}_k \times \{k\},$$

for some discrete set $\mathcal{J}_k \subset \mathbb{R}$. For $\chi_{[0,1]^2} + \mathcal{J}$ to be a tiling of $\mathbb{R}^2$, the set $\mathcal{J}_k$ must be of the form $\mathcal{J}_k = \mathbb{Z} + a_k$. In that case $\mathcal{J}$ can be expressed as one of the sets in the first collection appearing in (3.1).

Similarly, if there exists $(s_1, s_2) \in \mathcal{J}$ such that s_2 is not an integer and $s_1 \in \mathbb{Z}$, $\mathcal{J}$ can be expressed as one of the sets in the second collection appearing in (3.1). This completes the proof. □

We say that the Gabor orthonormal basis $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ is *standard* if

$$\Lambda = \bigcup_{t \in \mathcal{J}} \{t\} \times \Lambda_t,$$

where $\mathcal{J} + [0, 1]^d$ tiles $\mathbb{R}^d$ and Λ_t is a spectrum for $[0, 1]^d$. (Note that, by the result in [12], $\Lambda_t + [0, 1]^d$ must then be a tiling of $\mathbb{R}^d$ for every $t \in \mathcal{J}$.)

The following result settles the one-dimensional case.

Theorem 3.3. *$\mathcal{G}(\chi_{[0,1]}, \Lambda)$ is a Gabor orthonormal basis if and only if Λ is standard.*

Proof. We just need to show that Λ being standard is a necessary condition for $\mathcal{G}(\chi_{[0,1]}, \Lambda)$ to be a Gabor orthonormal basis. We can also assume, for simplicity, that $(0, 0) \in \Lambda$. By Proposition 3.1, if $\mathcal{G}(\chi_{[0,1]}, \Lambda)$ is a Gabor orthonormal basis, then $\Lambda - \Lambda \subset \mathcal{Z}(V_g g) \cup \{0\}$ and $\Lambda + [0, 1]^2$ must be a tiling of $\mathbb{R}^2$. By Proposition 3.2, Λ must be of either one of the forms in (3.1). Note that Λ is standard in the second case. In order to deal with the first case, suppose that

$$\Lambda = \bigcup_{k \in \mathbb{Z}} (\mathbb{Z} + a_k) \times \{k\}, \quad \text{with } a_k \in [0, 1),\ k \neq 0,\ a_0 = 0.$$

We now show that this is impossible unless $a_k = 0$ for all k (which reduces to the case $\Lambda = \mathbb{Z}^2$, which is standard). We can assume, without loss of generality, that $a_k \neq 0$ for some $k > 0$ with k being the smallest such index. If $a_k \neq 0$ for some k, then both

(a_k, k) and $(0, k-1)$ are in Λ. The orthogonality of the Gabor system then implies that $(a_k, 1) \in \mathcal{Z}(V_g g)$. Using (2.2), we deduce that $1 \cdot (1 - |a_k|) \in \mathbb{Z} \setminus \{0\}$. That means a_k must be an integer, which is a contradiction. Hence, the first case is impossible unless $a_k = 0$ for all k and the proof is completed. □

A description of all time-frequency sets Λ for which $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis however become vastly more complicated when $d \geq 2$. In particular, as we will see, the standard structure cannot cover all possible cases. Consider integers $m, n > 0$ such that $m + n = d$. For convenience and to be consistent with our previous notation, we will write the cartesian product of the two time-frequency spaces $\mathbb{R}^{2m}$ and $\mathbb{R}^{2n}$ in the non-standard form

$$\mathbb{R}^{2d} = \mathbb{R}^{2m} \times \mathbb{R}^{2n} = \{(s, t, \lambda, \nu) : (s, \lambda) \in \mathbb{R}^{2m},\ (t, \nu) \in \mathbb{R}^{2n}\}.$$

We will also denote by Π_1 the projection operator from $\mathbb{R}^{2d}$ to $\mathbb{R}^{2m}$ defined by

$$\Pi_1\left((s, t, \lambda, \nu)\right) = (s, \lambda), \quad (s, t, \lambda, \nu) \in \mathbb{R}^{2d} = \mathbb{R}^{2m} \times \mathbb{R}^{2n}. \tag{3.2}$$

To simplify the notation, we also define $g_k = \chi_{[0,1]^k}$ for any $k \geq 1$. We now build a new family of time-frequency sets on $\mathbb{R}^{2d}$ as follows. Suppose that $\mathcal{G}(\chi_{[0,1]^m}, \Lambda_1)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^m)$ and that we associate with each $(s, \lambda) \in \Lambda_1$, a discrete set $\Lambda_{(s,\lambda)}$ in $\mathbb{R}^{2n}$ such that $\mathcal{G}(\chi_{[0,1]^n}, \Lambda_{(s,\lambda)})$ is a Gabor orthonormal basis of $L^2(\mathbb{R}^n)$. We then define

$$\Lambda = \bigcup_{(s,\lambda)\in\Lambda_1} \{(s, t, \lambda, \nu) : (t, \nu) \in \Lambda_{(s,\lambda)}\}. \tag{3.3}$$

We say that a Gabor system $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ with Λ as in (3.3) is *pseudo-standard*.

Proposition 3.4. *Every pseudo-standard Gabor system $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis of $L^2(\mathbb{R}^d)$.*

Proof. If $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, we have $g_d(x, y) = g_m(x) g_n(y)$ (for $m + n = d$). This yields immediately that

$$V_{g_d} g_d(s, t, \lambda, \nu) = V_{g_m} g_m(s, \lambda)\, V_{g_n} g_n(t, \nu), \quad (s, \lambda) \in \mathbb{R}^{2m},\ (t, \nu) \in \mathbb{R}^{2n}. \tag{3.4}$$

Suppose that $\rho = (s, t, \lambda, \nu)$ and $\rho' = (s', t', \lambda', \nu')$ are distinct elements of Λ. If $(s, \lambda) = (s', \lambda')$, then (t, ν) and (t', ν') are distinct elements of $\Lambda_{(s,\lambda)}$ and we have thus

$$(t' - t, \nu' - \nu) \in \mathcal{Z}(V_{g_n} g_n)$$

which implies that $\mathcal{Z}(V_{g_d} g_d)(\rho' - \rho) = 0$. On the other hand, if $(s, \lambda) \neq (s', \lambda'))$, we have then

$$(s' - s, \lambda' - \lambda) \in \mathcal{Z}(V_{g_m} g_m)$$

which implies again that $\mathcal{Z}(V_{g_d} g_d)(\rho' - \rho) = 0$. This proves the orthonormality of the system $\mathcal{G}(\chi_{[0,1]^d}, \Lambda)$. This proposition can now be proved by invoking Theorem 3.1 if we can show that $\Lambda + [0,1]^{2d}$ is a tiling of $\mathbb{R}^{2d}$. To prove this, we note that $\Lambda_1 + [0,1]^{2m}$ is a tiling of the subspace $\mathbb{R}^{2m}$ by Theorem 3.1 and that, similarly, for each $(t, \lambda) \in \Lambda_{(t,\lambda)} + [0,1]^{2n}$ is a tiling of $\mathbb{R}^{2n}$. This easily implies the required tiling property and concludes the proof. □

Example 3.5. Consider the two-dimensional case $d = 2$. Let

$$\Lambda_1 = \bigcup_{m \in \mathbb{Z}} \{m\} \times (\mathbb{Z} + \mu_m), \quad \mu_m \in [0,1).$$

Associate with each $\gamma = (m, j + \mu_m) \in \Lambda_1$, the set

$$\Lambda_\gamma = \bigcup_{n \in \mathbb{Z}} \{n + s_{m,j}\} \times (\mathbb{Z} + \nu_{n,m,j}), \quad s_{m,j} \in \mathbb{R},\ \nu_{n,m,j} \in [0,1).$$

Then,

$$\Lambda := \{(m, n + s_{m,j}, j + \mu_m, k + \nu_{n,m,j}) : m, n, j, k \in \mathbb{Z}\}$$

(written in the form of $(t_1, t_2, \lambda_1, \lambda_2)$ where (t_1, t_2) are the translations and (λ_1, λ_2) the frequencies) has the pseudo-standard structure. Note that the parameters $s_{m,j}$ can be chosen so that the set Λ is not standard as the set

$$\{(m, n + s_{m,j}) : m, n, j \in \mathbb{Z}\} + [0,1]^2$$

will not tile $\mathbb{R}^2$ in general. For example, for $m = n = 0$, we could let $s_{0,0} = 0$ and the numbers $s_{0,j}$ could be chosen as distinct numbers in the interval $[0,1)$. The square $[0,1]^2$ would then overlap with infinitely many of its translates appearing as part of the Gabor system.

Using a similar procedure to higher dimension, we can produce many non-standard Gabor orthonormal bases with window $\chi_{[0,1]^d}$. However, the pseudo-standard structure still cannot cover all possible cases of time-frequency sets. A time-frequency set could be a mixture of pseudo-standard and standard structure. For example, consider the set

$$\Lambda = \bigcup_{n \in \mathbb{Z} \setminus \{1\}} \{(m + t_{n,k}, n, j + \mu_{k,m,n}, k + \nu_n) : j, k \in \mathbb{Z}\} \cup \{(m, 1)\} \times \Lambda_m,$$

where $\Lambda_m + [0,1]^2$ tiles $\mathbb{R}^2$. This set consists of two parts. The first part is a subset of a set having the pseudo-standard structure while the second part is a subset of a set having the standard one. Moreover, the translates of the unit square associated with the first part are disjoint with those associated with the second part, showing that $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ is a mutually orthogonal set. Since Λ is clearly a tiling of $\mathbb{R}^4$, Theorem 3.1 shows that $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ is a Gabor orthonormal basis.

In the next section, we will classify all possible sets $\Lambda \subset \mathbb{R}^4$ with the property that $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^2)$. However, we have

Theorem 3.6. *Let $d = m + n$ and let $\Pi_1 : \mathbb{R}^{2d} \to \mathbb{R}^{2m}$ be defined by (3.2). Suppose that $(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis and that $\Pi_1(\Lambda) + [0,1]^{2m}$ tiles $\mathbb{R}^{2m}$. Then Λ has the pseudo-standard structure.*

Proposition 3.7. *Let $d = m + n$ and suppose that $(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^d)$. If $(s_0, \lambda_0) \in \mathbb{R}^{2m}$, consider the translate of the unit hypercube in $\mathbb{R}^{2m}$, $C = (s_0, \lambda_0) + [0,1)^{2m}$, and define*

$$\Lambda(C) := \{(t, \nu) \in \mathbb{R}^{2n} : (s, t, \lambda, \nu) \in \Lambda \text{ and } (s, \lambda) \in C\}.$$

Then $(\chi_{[0,1]^n}, \Lambda(C))$ is a Gabor orthonormal basis for $L^2(\mathbb{R}^{2n})$.

Proof. We first show that the system $(\chi_{[0,1]^n}, \Lambda(C))$ is orthogonal. Let (t, ν) and (t', ν') be distinct elements of $\Lambda(C)$. There exist (s, λ) and (s', λ') in $\mathbb{R}^{2m}$ such that (s, t, λ, ν) and (s', t', λ', ν') both belong to Λ. Using the mutual orthogonality of the system $(\chi_{[0,1]^d}, \Lambda)$ together with (3.4), we have

$$V_{g_m} g_m(s - s', \lambda - \lambda') = 0 \quad \text{or} \quad V_{g_n} g_n(t - t', \nu - \nu') = 0.$$

Note that, as both (s, λ) and (s', λ') belong to C, we have $|s - s'|_{\max} < 1$ and $|\lambda - \lambda'|_{\max} < 1$. In particular, $V_{g_m} g_m(s - s', \lambda - \lambda') \neq 0$ and the orthogonality of the system $(\chi_{[0,1]^n}, \Lambda(C))$ follows.

If $(s, \lambda) \in \Pi_1(\Lambda)$ (as defined in (3.2)), let

$$\Lambda_{(s,\lambda)} = \{(t, \nu) : (s, t, \lambda, \nu) \in \Lambda\}.$$

Let $f_1 \in L^2(\mathbb{R}^m)$, $f_2 \in L^2(\mathbb{R}^n)$ and $(s_0, \lambda_0) \in \mathbb{R}^{2m}$. Applying Parseval's identity to the function

$$f(x, y) = e^{2\pi i \lambda_0 \cdot x} f_1(x - s_0) f_2(y), \quad x \in \mathbb{R}^m, \ y \in \mathbb{R}^n,$$

we obtain that

$$\begin{aligned}
&\int_{\mathbb{R}^m} |f_1(x)|^2\,dx \int_{\mathbb{R}^n} |f_2(y)|^2\,dy \\
&\quad = \sum_{(s,\lambda)\in\Pi_1(\Lambda)} \sum_{(t,\nu)\in\Lambda_{(s,\lambda)}} |V_{g_m} f_1(s-s_0,\lambda-\lambda_0)|^2\,|V_{g_n} f_2(t,\nu)|^2 \\
&\quad = \sum_{(s,\lambda)\in\Pi_1(\Lambda)} \sum_{(t,\nu)\in\Lambda_{(s,\lambda)}} |V_{f_1} g_m(s_0-s,\lambda_0-\lambda)|^2\,|V_{g_n} f_2(t,\nu)|^2.
\end{aligned}$$

Defining

$$w(s,\lambda) = \|f_2\|_2^{-2} \sum_{(t,\nu)\in\Lambda_{(s,\lambda)}} |V_{g_n} f_2(t,\nu)|^2 \quad \text{and} \quad \mu = \sum_{(s,\lambda)\in\Pi_1(\Lambda)} w(s,\lambda)\,\delta_{(s,\lambda)}$$

for $f_2 \neq 0$, the above identity can be written as

$$\int_{\mathbb{R}^m} |f_1(x)|^2\,dx = \sum_{(s,\lambda)\in\Pi_1(\Lambda)} w(s,\lambda)\,|V_{f_1} g_m(s_0-s,\lambda_0-\lambda)|^2 = \left(\mu * |V_{f_1} g_m|^2\right)(s_0,\lambda_0).$$

On the other hand, letting $\check{\chi}_{[0,1)^{2m}}(s,\lambda) = \chi_{[0,1)^{2m}}(-s,-\lambda)$ and defining C and $\Lambda(C)$ as above, we have also

$$\begin{aligned}
\left(\mu * \check{\chi}_{[0,1)^m}\right)(s_0,\lambda_0) &= \sum_{(s,\lambda)\in\Pi_1(\Lambda)} w(s,\lambda)\,\chi_{[0,1)^{2m}}(s-s_0,\lambda-\lambda_0) \\
&= \sum_{(s,\lambda)\in\Pi_1(\Lambda)\cap C} w(s,\lambda) \\
&= \|f_2\|_2^{-2} \sum_{(t,\nu)\in\Lambda(C)} |V_{g_n} f_2(t,\nu)|^2 \le 1,
\end{aligned}$$

where the last inequality results from the orthogonality of the system $(\chi_{[0,1]^n}, \Lambda(C))$ proved earlier. Since (s_0,λ_0) is arbitrary in $\mathbb{R}^{2m}$ and

$$\int_{\mathbb{R}^{2m}} |V_{f_1} g_m(s,\lambda)|^2\,ds\,d\lambda = \|f_1\|_2^2,$$

Theorem 2.1 can be used to deduce that $\mu * \check{\chi}_{[0,1)^m} = 1$. This shows that

$$\sum_{(t,\nu)\in\Lambda(C)} |V_{g_n} f_2(t,\nu)|^2 = \|f_2\|^2, \quad f_2 \in L^2(\mathbb{R}^n),$$

and thus that the system $(\chi_{[0,1]^n}, \Lambda(C))$ is complete, proving our claim. □

Proof of Theorem 3.6. Let $\mathcal{J} = \Pi_1(\Lambda)$ and, for any $(s,\lambda) \in \mathcal{J}$, define

$$\Lambda_{(s,\lambda)} = \{(t,\nu) : (s,t,\lambda,\nu) \in \Lambda\}.$$

If $(s_0, \lambda_0) \in \mathcal{J}$, let $C = (s_0, \lambda_0) + [0,1)^{2m}$, and

$$\Lambda(C) := \{(t, \nu) \in \mathbb{R}^{2n} : (s, t, \lambda, \nu) \in \Lambda \text{ and } (s, \lambda) \in C\}.$$

Proposition 3.7 shows that the system $(\chi_{[0,1]^n}, \Lambda(C))$ forms a Gabor orthonormal basis. By assumption $\mathcal{J} + [0,1)^{2m}$ tiles $\mathbb{R}^{2m}$. Hence, $(s_0, \lambda_0) + [0,1)^{2m}$ contains exactly one point in $\mathcal{J}$, i.e. (s_0, λ_0), and we have

$$\Lambda(C) = \{(t, \nu) : (s_0, t, \lambda_0, \nu) \in \Lambda\} = \Lambda_{(s_0, \lambda_0)}.$$

Therefore, we can write Λ as

$$\Lambda = \bigcup_{(s_0, \lambda_0) \in \mathcal{J}} \{(s_0, \lambda_0)\} \times \Lambda_{(s_0, \lambda_0)}.$$

Our proof will be complete if we can show that $\mathcal{J}$ is a Gabor orthonormal basis of $L^2(\mathbb{R}^m)$.

As $\mathcal{J}$ is a tiling set, by Proposition 3.1 it suffices to show that the inclusion $\mathcal{J} - \mathcal{J} \subset \mathcal{Z}(V_{g_m} g_m) \cup \{0\}$ holds. Let (s, λ) and (s', λ') be distinct points in $\mathcal{J}$. As $\Lambda_{(s,\lambda)} + [0,1)^{2n}$ tiles $\mathbb{R}^{2n}$, so does $\Lambda_{(s,\lambda)} + [-1,0)^{2n}$, and we can find $(t, \nu) \in \Lambda_{(s,\lambda)}$ such that $0 \in (t, \nu) + [-1,0)^{2n}$, or, equivalently, with $(t, \nu) \in [0,1)^{2n}$. Similarly, we can find $(t', \nu') \in \Lambda_{(s',\lambda')}$ such that $(t', \nu') \in [0,1)^{2n}$. Using the fact that $(\chi_{[0,1]^d}, \Lambda)$ is a Gabor orthonormal basis of $L^2(\mathbb{R}^{2d})$, we have

$$(s, t, \lambda, \nu) - (s', t', \lambda', \nu') \in \mathcal{Z}(V_{g_d} g_d),$$

or, equivalently,

$$V_{g_m} g_m(s - s', \lambda - \lambda') = 0 \quad \text{or} \quad V_{g_n} g_n(t - t', \nu - \nu') = 0.$$

Note that, since $|t - t'| < 1$ and $|\nu - \nu'| < 1$, $V_{g_n} g_n(t - t', \nu - \nu') \neq 0$. Hence $(s, \lambda) - (s', \lambda') \in \mathcal{Z}(V_{g_m} g_m)$ as claimed. □

4. Two-dimensional Gabor orthonormal bases

In this section, our goal will be to classify all possible Gabor orthonormal basis generated by the unit square on $\mathbb{R}^2$.

Given a fixed Gabor orthonormal basis $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ and a set $A \subset \mathbb{R}^2$, we define the sets

$$\Gamma(A) = \{(\lambda_1, \lambda_2) \in \mathbb{R}^2 : (t_1, t_2, \lambda_1, \lambda_2) \in \Lambda,\ (t_1, t_2) \in A\},$$

and, for any $(\lambda_1, \lambda_2) \in \mathbb{R}^2$ and any set $B \subset \mathbb{R}^2$, we let

$$T_A(\lambda_1, \lambda_2) = \{(t_1, t_2) \in \mathbb{R}^2 : (t_1, t_2, \lambda_1, \lambda_2) \in \Lambda,\ (t_1, t_2) \in A\}$$

and

$$T_A(B) = \{(t_1, t_2) \in \mathbb{R}^2 : (t_1, t_2, \lambda_1, \lambda_2) \in \Lambda,\ (t_1, t_2) \in A,\ (\lambda_1, \lambda_2) \in B\}.$$

In particular, the set $T_A(\Gamma(A))$ collects all the couples $(t_1, t_2) \in A$ such that $(t_1, t_2, \lambda_1, \lambda_2) \in \Lambda$ for some $(\lambda_1, \lambda_2) \in \mathbb{R}^2$.

We say that a square is *half-open* if it is a translate of one of the sets

$$[0,1)^2, \quad (0,1]^2, \quad [0,1) \times (0,1] \quad \text{or} \quad (0,1] \times [0,1).$$

Two measurable subsets of $\mathbb{R}^d$ will be called *essentially disjoint* if their intersection has zero Lebesgue measure. In the derivation below, we will make use of the identity

$$V_{g_2} g_2(t_1, t_2, \lambda_1, \lambda_2) = V_{g_1} g_1(t_1, \lambda_1) V_{g_1} g_1(t_2, \lambda_2), \quad (t_1, t_2, \lambda_1, \lambda_2) \in \mathbb{R}^4,$$

which implies, in particular, that

$$V_{g_2} g_2(t_1, t_2, \lambda_1, \lambda_2) = 0 \quad \Leftrightarrow \quad V_{g_1} g_1(t_1, \lambda_1) = 0 \quad \text{or} \quad V_{g_1} g_1(t_2, \lambda_2) = 0.$$

Moreover, using (2.3), the zero set of $V_{g_2} g_2$ is given by

$$\mathcal{Z}(V_{g_2} g_2) = \{(t, \lambda) : |t|_{\max} \geq 1\} \cup \left(\bigcup_{i=1}^{2} \{(t, \nu) : \lambda_i (1 - |t_i|) \in \mathbb{Z} \setminus \{0\})\} \right). \tag{4.1}$$

This implies that if $|t|_{\max} < 1$ and $(t, \lambda) \in \mathcal{Z}(V_{g_2} g_2)$, then, there exists $i \in \{1, 2\}$ and for some integer $m \neq 0$ such that

$$|\lambda_i| = \frac{|m|}{1 - |t_i|} \geq 1,$$

with a strict inequality if $t_i \neq 0$. These properties will be used throughout this section.

Lemma 4.1. *Let $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ be a Gabor orthonormal basis for $L^2(\mathbb{R}^2)$ and let C be a half-open square. Then,*

(i) *$\Gamma(C) + [0,1]^2$ is a packing of $\mathbb{R}^2$.*
(ii) *If $(\lambda_1, \lambda_2) \in \Gamma(C)$, then $T_C(\lambda_1, \lambda_2)$ consists of one point.*

Proof. (i) Let (λ_1, λ_2) and (λ_1', λ_2') be distinct elements of $\Gamma(C)$. By definition, we can find (t_1, t_2) and (t_1', t_2') in C such that $(t_1, t_2, \lambda_1, \lambda_2), (t_1', t_2', \lambda_1', \lambda_2') \in \Lambda$. We then have

$$0 = V_{g_1} g_1(t_1 - t_1', \lambda_1 - \lambda_1')\, V_{g_1} g_1(t_2 - t_2', \lambda_2 - \lambda_2').$$

If, without loss of generality, the first factor on the right-hand side of the previous equality vanishes, the fact that $|t_1 - t_1'| < 1$ shows the existence of an integer $k > 0$ such that

$$|\lambda_1 - \lambda_1'| = k/(1 - |t_1 - t_1'|) \geq 1.$$

Hence, the cubes $(\lambda_1, \lambda_2) + [0,1]^2$ and $(\lambda_1', \lambda_2') + [0,1]^2$ are essentially disjoint.

(ii) Suppose that $T_C(\lambda_1, \lambda_2)$ contains two distinct points (t_1, t_2) and (t_1', t_2'). Then,

$$0 = V_{g_1} g_1(t_1 - t_1', 0)\, V_{g_1} g_1(t_2 - t_2', 0).$$

As $V_{g_1} g_1(t, 0) \neq 0$ for any t with $|t| < 1$, we must have $|t_1 - t_1'| \geq 1$ or $|t_2 - t_2'| \geq 1$, contradicting the fact that both (t_1, t_2) and (t_1', t_2') belong to C. □

In the following, we will denote by ∂A the boundary of a set A. The next result will be useful.

Lemma 4.2. *Under the hypotheses of the previous lemma, consider an element $\lambda = (\lambda_1, \lambda_2)$ of $\Gamma(C)$ and let $T_C(\lambda) = \{(t_1, t_2)\}$. Then for any $x \in \partial(\lambda + [0,1]^2)$, we can find $\lambda_x = (\lambda_{1,x}, \lambda_{2,x}) \in \Gamma(C)$ such that $x \in \partial(\lambda_x + [0,1]^2)$. Moreover, for any such λ_x, letting $T_C(\lambda_x) = \{t_x\}$, where $t_x = (t_{1,x}, t_{2,x})$, we can find $i_0 \in \{1,2\}$ such that $t_{i_0,x} = t_{i_0}$ and $\lambda_{i_0,x} = \lambda_{i_0} + 1$ or $\lambda_{i_0} - 1$.*

Proof. We can write $x = (\lambda_1 + \epsilon_1, \lambda_2 + \epsilon_2)$, where $0 \leq \epsilon_i \leq 1$, $i = 1, 2$ and $\epsilon_i \in \{0, 1\}$ for at least one index i. Let $a = (a_1, a_2) \in \mathbb{R}^2$ with $0 < a_i < 1$ for $i = 1, 2$ and consider the point $(t_a, x) := (t_1 + a_1, t_2 + a_2, \lambda_1 + \epsilon_1, \lambda_2 + \epsilon_2)$ in $\mathbb{R}^4$. Since $\Lambda + [0,1]^4$ is a tiling on $\mathbb{R}^4$ and the point (t_a, x) is a point on the boundary of $(t, \lambda) + [0,1]^4$, we can find some point $(t_{x,a}, \lambda_{x,a}) \in \Lambda \setminus \{(t, \lambda)\}$ such that $(t_a, x) \in (t_{x,a}, \lambda_{x,a}) + [0,1]^4$. Let $t_{x,a} = (t_1', t_2')$ and $\lambda_{x,a} = (\lambda_1', \lambda_2')$. We have

$$\begin{cases} -a_i \leq t_i - t_i' \leq 1 - a_i, \\ -\epsilon_i \leq \lambda_i - \lambda_i' \leq 1 - \epsilon_i, \end{cases} \quad i = 1, 2. \tag{4.2}$$

Using the orthogonality of the system $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$, we can find $i_0 \in \{1, 2\}$ such that $V_{g_1} g_1(t_{i_0} - t_{i_0}', \lambda_{i_0} - \lambda_{i_0}') = 0$. Note that $t_{i_0} - t_{i_0}' \neq 0$ would imply that $|\lambda_{i_0} - \lambda_{i_0}'| > 1$ which is impossible from (4.2). Hence, $t_{i_0} = t_{i_0}'$ and $\lambda_{i_0} - \lambda_{i_0}' \neq 0$.

Moreover, as $V_{g_1} g_1(0, v) \neq 0$ if $|v| < 1$, $V_{g_1} g_1(t_{i_0} - t_{i_0}', \lambda_{i_0} - \lambda_{i_0}') = 0$ can only occur if $|\lambda_{i_0} - \lambda_{i_0}'| = 1$. This shows also that $\epsilon_{i_0} \in \{0, 1\}$ in that case. This proves the last statement of our claim and the fact that $x \in \partial(\lambda_{x,a} + [0,1]^2)$. The proof will be complete if we can show that $\lambda_{x,a} \in \Gamma(C)$ for some choice of a.

For simplicity, we consider the half-open square to be $C = [b_1, b_1 + 1) \times [b_2, b_2 + 1)$. Our assertion will be true if the point $t_{x,a} = (t_1', t_2')$ constructed above satisfies the inequalities $b_i \leq t_i' < b_i + 1$ for $i = 1, 2$. As $t_{i_0} = t_{i_0}'$, the inequalities clearly hold for

$i = i_0$. Suppose that the other index j falls out of the range, say $t'_j < b_j$ (the case $t'_j \geq b_j + 1$ is similar). We consider $(t_{a'}, x)$ with $a'_j = t'_j + 1 - t_j + \delta$ for some small $\delta > 0$. Note that, by (4.2), we have $t_i + a_i - 1 \leq t'_i \leq t_i + a_i$ for $i = 1, 2$, and, in particular,

$$a'_j = t'_j + 1 - t_j + \delta \geq a_j + \delta > 0.$$

We have also $a'_j < 1$. Indeed, the inequality $t'_j - t_j + 1 + \delta \geq 1$ would imply that $t'_j + 1 + \delta \geq 1 + t_j$. This is not possible, as $b_j \leq t_j < b_j + 1$, so $1 + t_j \geq b_j + 1$. But $t'_j < b_j$, so $t'_j + 1 < b_j + 1$, so for δ small,

$$t'_j + 1 + \delta < b_j + 1 \leq 1 + t_j$$

which yields a contradiction.

Using the previous argument with a' replacing a, we guarantee the existence of t''_j such that $t'_j + \delta = t_j + a'_j - 1 \leq t''_j \leq t_j + a'_j = t'_j + 1 + \delta$ and the associated point $(t_{a'}, \lambda_{x,a'}) = (t''_1, t''_2, \lambda''_1, \lambda''_2)$ in Λ with the property that $x \in \partial(\lambda_{x,a'} + [0,1]^2)$ for some index i'_0 such that $|\lambda_{i'_0} - \lambda''_{i'_0}| = 1$, $t_{i'_0} = t''_{i'_0}$ and $\epsilon_{i'_0} \in \{0, 1\}$. We claim that $t''_j = t'_j + 1$. Now, $(t'_1, t'_2, \lambda'_1, \lambda'_2)$ and $(t''_1, t''_2, \lambda''_1, \lambda''_2)$ are in Λ. The mutual orthogonality property implies that $V_{g_1} g_1(t'_i - t''_i, \lambda'_i - \lambda''_i) = 0$ for some $i = 1, 2$.

Suppose that x is not of the corner points of $\lambda + [0,1]^2$. In that case, the index i such that $\epsilon_i \in \{0, 1\}$ is unique and it follows that $i_0 = i'_0$. This implies in particular, that $t'_{i_0} = t''_{i_0}$ (as $t'_{i_0} = t_{i_0} = t_{i'_0} = t''_{i'_0} = t''_{i_0}$). Furthermore, the second set of inequalities in (4.2) shows that $\lambda'_{i_0} = \lambda''_{i_0} = \lambda_{i_0} - 1$ if $\epsilon_{i_0} = 0$ and $\lambda'_{i_0} = \lambda''_{i_0} = \lambda_{i_0} + 1$ if $\epsilon_{i_0} = 1$. We have thus $\lambda'_{i_0} = \lambda''_{i_0}$ in both cases. We have thus

$$V_{g_1} g_1(t'_{i_0} - t''_{i_0}, \lambda'_{i_0} - \lambda''_{i_0}) = V_{g_1} g_1(0, 0) = 1.$$

Therefore, the other index j must satisfy $V_{g_1} g_1(t'_j - t''_j, \lambda'_j - \lambda''_j) = 0$. The inequalities

$$-\epsilon_j \leq \lambda_j - \lambda'_j \leq 1 - \epsilon_j \quad \text{and} \quad -\epsilon_j \leq \lambda_j - \lambda''_j \leq 1 - \epsilon_j$$

yield $-1 \leq \lambda'_j - \lambda''_j \leq 1$. However, $\delta \leq t''_j - t'_j \leq 1 + \delta$. The $V_{g_1} g_1$ would not be zero unless $t''_j \geq t'_j + 1$ ($\geq b_j$). Hence, $t'_j + 1 \leq t''_j \leq t'_j + 1 + \delta$. This forces that $t''_j = t'_j + 1$. This completes the proof for non-corner points. If x is of the corner point, as the square constructed for the non-corner will certainly cover the corner point. Therefore, the proof is completed. □

With the help of the previous two lemmas, the following tiling result for $\Gamma(C)$ follows immediately.

Corollary 4.3. *Let C be a half-open square. Then $\Gamma(C) + [0,1]^2$ is a tiling of $\mathbb{R}^2$.*

Proof. It suffices to prove the following statement: suppose that $\mathcal{J} + [0,1]^2$ is non-empty packing of $\mathbb{R}^2$. If, for any $x \in \partial(t + [0,1]^2)$ where $t \in \mathcal{J}$, we can find $t_x \in \mathcal{J}$ with $t_x \neq t$

such that $x \in \partial(t_x + [0,1]^2)$, then $\mathcal{J} + [0,1]^2$ is a tiling of $\mathbb{R}^2$. Indeed, by Lemma 4.1(i) and Lemma 4.2, $\Gamma(C) + [0,1]^2$ is a packing of $\mathbb{R}^2$ and satisfies the stated property. It is thus a tiling of $\mathbb{R}^2$.

To prove the previous statement, we note that as $\mathcal{J} + [0,1]^2$ is packing, it is a closed set. Suppose that $\mathcal{J} + [0,1]^2$ satisfies the property above and that $\mathbb{R}^d \setminus (\mathcal{J} + [0,1]^2) \neq \emptyset$. Let $x \in \partial(\mathcal{J} + [0,1]^2)$ and assume that $x \in t + [0,1]^2$. We can then find $t_x \in \mathcal{J}$ with $t_x \neq t$ such that $x \in \partial(t_x + [0,1]^2)$. Note that if x were not a corner point of either $t + [0,1]^2$ or $t_x + [0,1]^2$, then x would be in the interior of $\mathcal{J} + [0,1]^2$. Hence, x must be a corner point of $t + [0,1]^2$ or $t_x + [0,1]^2$. As the set of all the corner points of the squares in $\mathcal{J} + [0,1]^2$ is countable, the Lebesgue measure of the open set $\mathbb{R}^d \setminus (\mathcal{J} + [0,1]^2)$ is zero and $\mathbb{R}^d \setminus (\mathcal{J} + [0,1]^2)$ is thus empty, proving our claim. □

Lemma 4.4. *Let C be a half-open square and suppose that $(\lambda_1, \lambda_2) \in \Gamma(C)$ with $T_C(\lambda_1, \lambda_2) = \{(t_1, t_2)\}$. Then all the sets $T_C(\lambda_1', \lambda_2')$ with $(\lambda_1', \lambda_2') \in \Gamma(C)$ are either of the form $\{(t_1, t_2 + s)\}$ or $\{(t_1 + s, t_2)\}$ for some real s with $|s| < 1$ depending on (λ_1, λ_2).*

Proof. We first make the following remark. If $(\alpha_1, \alpha_2), (\beta_1, \beta_2) \in \Gamma(C)$ are such that the two squares $(\alpha_1, \alpha_2) + [0,1]^2$ and $(\beta_1, \beta_2) + [0,1]^2$ intersect each other and also both intersect a third square $(\gamma_1, \gamma_2) + [0,1]^2$ with $(\gamma_1, \gamma_2) \in \Gamma(C)$, then, letting $T_C(\gamma_1, \gamma_2) = \{(r_1, r_2)\}$, we have

$$T_C(\alpha_1, \alpha_2) = \{(r_1 + a, r_2)\} \quad \text{and} \quad T_C(\beta_1, \beta_2) = \{(r_1 + b, r_2)\}$$

or

$$T_C(\alpha_1, \alpha_2) = \{(r_1, r_2 + a)\} \quad \text{and} \quad T_C(\beta_1, \beta_2) = \{(r_1, r_2 + b)\},$$

for some real a, b. Indeed, using Lemma 4.2, we have $T_C(\alpha_1, \alpha_2) = \{(r_1 + a, r_2)\}$ or $\{(r_1, r_2 + a)\}$ and $T_C(\beta_1, \beta_2) = \{(r_1 + b, r_2)\}$ or $\{(r_1, r_2 + b)\}$. Suppose, for example, that $T_C(\alpha_1, \alpha_2) = \{(r_1 + a, r_2)\}$ and $T_C(\beta_1, \beta_2) = \{(r_1, r_2 + b)$. Since the two squares intersect each other, we must have $|\alpha_1 - \beta_1| \leq 1$ and $|\alpha_2 - \beta_2| \leq 1$. The orthogonality property also implies that either $(a, \alpha_1 - \beta_1)$ or $(-b, \alpha_2 - \beta_2)$ is in the zero set of $V_{g_1} g_1$. But since we have $|a|, |b| < 1$, this would imply that $|\alpha_1 - \beta_1| > 1$ or $|\alpha_1 - \beta_2| > 1$, which cannot happen. As $\Gamma(C) + [0,1]^2$ is a tiling of $\mathbb{R}^2$, for any square $(\sigma_1, \sigma_2) + [0,1]^2$ intersecting the square $(\lambda_1, \lambda_2) + [0,1]^2$ and with $(\sigma_1, \sigma_2) \in \Gamma(C)$, we can find another square $(\delta_1, \delta_2) + [0,1]^2$, with $(\delta_1, \delta_2) \in \Gamma(C)$ and with $(\delta_1, \delta_2) + [0,1]^2$ intersecting both squares $(\sigma_1, \sigma_2) + [0,1]^2$ and $(\lambda_1, \lambda_2) + [0,1]^2$. By the previous remark, the conclusion of the lemma holds for all the squares that neighbor the square $(\lambda_1, \lambda_2) + [0,1]^2$. Replacing this original square by one of the neighboring squares and continuing this process, we obtain the conclusion of the lemma for all the squares in the tiling $\Gamma(C) + [0,1]^2$ by an induction argument. This proves our claim. □

Suppose that the system $\mathcal{G}(\chi_{[0,1]^2}, \Lambda)$ gives rise to a non-standard Gabor orthonormal basis of $L^2(\mathbb{R}^2)$. Then, some of the squares will have overlaps and, without loss of generality, we can assume that

$$|[0,1]^2 \cap [0,1]^2 + (t_1, t_2)| > 0$$

for some (t_1, t_2) in the translation component of Λ.

Lemma 4.5. *If $(0,0,0,0) \in \Lambda$, then the sets $T_{[0,1)^2}(\lambda_1, \lambda_2)$ where $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$ are either all of the form $\{(t,0)\}$ or all of the form $\{(0,t)\}$ with some t (depending on (λ_1, λ_2)) with $|t| < 1$. In the first case, if there exists some $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$ with $T_{[0,1)^2}(\lambda_1, \lambda_2) = (t, 0)$ and $t \neq 0$, then*

$$\Gamma([0,1)^2) = \bigcup_{k \in \mathbb{Z}} (\mathbb{Z} + \mu_{k,0}) \times \{k\} \tag{4.3}$$

for some $0 \leq \mu_{k,0} < 1$. Moreover, we can find $0 \leq t_k < 1$ such that

$$T_{[0,1)^2}((\mathbb{Z} + \mu_{k,0}) \times \{k\}) = \{(t_k, 0)\}, \quad k \in \mathbb{Z}, \tag{4.4}$$

and

$$\Lambda \cap ([0,1)^2 \times \mathbb{R}^2) = \{(t_k, 0, j + \mu_{k,0}, k) : j, k \in \mathbb{Z}\}. \tag{4.5}$$

(In the second case, $\Gamma([0,1)^2) = \bigcup_{k \in \mathbb{Z}} \{k\} \times (\mathbb{Z} + \mu_{k,0})$ and $T_{[0,1)^2}(\{k\} \times (\mathbb{Z} + \mu_{k,0})) = \{(0, t_k)\}$, $\Lambda \cap ([0,1)^2 \times \mathbb{R}^2) = \{(0, t_k, k, j + \mu_{k,0}) : j, k \in \mathbb{Z}\}$).

Proof. If $\lambda = (0,0)$, we have $T_{[0,1)^2}(\lambda) = \{(0,0)\}$ as $(0,0,0,0) \in \Lambda$. By Lemma 4.4, any $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$ with the square $(\lambda_1, \lambda_2) + [0,1]^2$ intersecting $[0,1]^2$ on the λ_1, λ_2-plane satisfies $T_{[0,1)^2}(\lambda_1, \lambda_2) = \{(t,0)\}$ or $T_{[0,1)^2}(\lambda_1, \lambda_2)\{(0,t)\}$ with $|t| < 1$. Without loss of generality, we assume that the first case holds. As $\Gamma([0,1)^2) + [0,1]^2$ is a tiling of $\mathbb{R}^2$, for any square $C = (\lambda_1, \lambda_2) + [0,1]^2$, with $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$, we can find squares $C_i = (\lambda_{1,i}, \lambda_{2,i}) + [0,1]^2$ for $i = 0, \ldots, k$ with $(\lambda_{1,i}, \lambda_{2,i}) \in \Gamma([0,1)^2)$ and such that $C_0 = [0,1]^2$, $C_k = C$, and with C_i and C_{i+1} touching each other for all $i = 0, \ldots, k-1$.

We have $T_{[0,1)^2}(\lambda_{1,1}, \lambda_{2,1}) = \{(t_1, 0)\}$ for some number t_1 with $|t_1| < 1$. Since C_2 and C_0 both intersect C_1, $T_{[0,1)^2}(\lambda_{1,2}, \lambda_{2,2}) = \{(t_2, 0)\}$ by Lemma 4.4 again. Inductively, we have $T_{[0,1)^2}(\lambda_{1,i}, \lambda_{2,i}) = \{(t_i, 0)\}$, $i = 1, \ldots, k$, which proves the first part.

Consider the case where, for any $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$, there exists a number $t = t(\lambda_1, \lambda_2)$ such that $T_{[0,1)^2}(\lambda_1, \lambda_2) = \{(t,0)\}$ and assume that $t(\lambda_1, \lambda_2) \neq 0$ for at least one couple $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$. Suppose that $\Gamma([0,1)^2)$ is not of the form in (4.3). By Corollary 4.3 and Proposition 3.2, we must have $\Gamma([0,1)^2) = \bigcup_{k \in \mathbb{Z}} \{k\} \times (\mathbb{Z} + a_k)$ with $0 \leq a_k < 1$ and at least one $a_k \neq 0$. Consider the distinct points

$$(t, 0, k, a_k + j) \quad \text{and} \quad (t', 0, k-1, a_{k-1} + j), \quad \text{both in } \Lambda.$$

We must have that either $(t-t',1) \in \mathcal{Z}(V_{g_1} g_1)$ or $(0, a_k - a_{k-1}) \in \mathcal{Z}(V_{g_1} g_1)$. However, since $|a_k - a_{k-1}| < 1$, the second case is impossible. This means that $(t-t',1) \in \mathcal{Z}(V_{g_1} g_1)$ which is possible only if $t = t'$. Therefore the fact that $(t, 0, k, a_k + j) \in \Lambda$ implies that $t = t_j$ for some real t_j. We now prove by induction on $|j|$ that $t_j = 0$ for all $j \in \mathbb{Z}$. The case $j = 0$ is clear as $(0,0,0,0) \in \Lambda$ by assumption. If our claim is true for all $|j| \leq J$ where $J \geq 0$, choose $k \in \mathbb{Z}$ such that $a_{k+1} \neq 0$ and $a_k = 0$ if such k exists. Suppose first that $j > 0$. There exist thus $t \in [0,1)$ such that

$$(t_{j+1}, 0, k, j+1) \quad \text{and} \quad (0, 0, k+1, a_{k+1} + j) \quad \text{both belong to } \Lambda.$$

This implies that either $(t, -1) \in \mathcal{Z}(V_{g_1} g_1)$ or $(0, a_{k+1} - 1) \in \mathcal{Z}(V_{g_1} g_1)$. This last case is impossible and the first one is only possible if $t = 0$, showing that $t_{j+1} = 0$. Similarly by considering the points

$$(t_{j-1}, 0, k+1, a_{k+1} + j - 1) \quad \text{and} \quad (0, 0, k, j) \quad \text{which both belong to } \Lambda,$$

we can conclude that $t_{j-1} = 0$ for $j < 0$. If k as above does not exist, there exists a choice $k' \in \mathbb{Z}$ such that $a_{k'-1} \neq 0$ and $a_{k'} = 0$. By considering the points

$$(t_{j+1}, 0, k', j+1) \quad \text{and} \quad (0, 0, k'-1, a_{k'-1} + j) \quad \text{if } j > 0$$

and the points

$$(t_{j-1}, 0, k'-1, a_{k'-1} + j - 1) \quad \text{and} \quad (0, 0, k', j) \quad \text{if } j < 0$$

which all belong to Λ, we conclude that $t_j = 0$ if $|j| = J + 1$. This proves (4.3).

If we are in the first case, i.e.

$$\Gamma([0,1)^2) = \bigcup_{k \in \mathbb{Z}} (\mathbb{Z} + \mu_{k,0}) \times \{k\},$$

let m, m' be distinct integers. We have then

$$T_{[0,1)^2}(m + \mu_{n,0}, n) = \{(t_m, 0)\} \quad \text{and} \quad T_{[0,1)^2}(m' + \mu_{n,0}, n) = \{(t_{m'}, 0)\}$$

which implies that $V_{g_1} g_1(t_m - t_{m'}, m - m') = 0$ or $V_{g_1} g_1(0,0) = 0$. The second case is clearly impossible while the first one is possible only when $t_m = t'_m$. This shows that (4.4) and (4.5) follow immediately from (4.3) and (4.4). □

Note that Lemma 4.5 implies that $\Gamma([0,1)^2) = \Gamma(\{(x,0) : 0 \leq x < 1\})$ and $\Gamma((0,1)^2) = \emptyset$ if $(0,0,0,0) \in \Lambda$.

Lemma 4.6. *Under the assumptions of Lemma 4.5, suppose that there exists* $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$ *with* $T_{[0,1)^2}(\lambda_1, \lambda_2) = (t, 0)$ *and* $t \neq 0$. *Then we can find numbers* t_k *with* $0 \leq t_k < 1$ *and* $\mu_{k,m}$, $k, m \in \mathbb{Z}$, *with* $0 \leq \mu_{k,m} < 1$, *such that*

$$\Lambda \cap (\mathbb{R} \times [0,1) \times \mathbb{R}^2) = \{(m + t_k, 0, j + \mu_{k,m}, k) : j, k, m \in \mathbb{Z}\}.$$

Proof. By the result of Lemma 4.5, we have the identities (4.4) and (4.5). Let $T = \{t_k,\ k \in \mathbb{Z}\} \subset [0,1)$ where t_k, $k \in \mathbb{Z}$, are the numbers appearing in (4.4). Let $s_1, s_2 \in T$ with $s_1 < s_2$. Consider the half-open squares $C = (s_1, 0) + [0,1)^2$ and $C' = (s_1, 0) + ((0,1] \times [0,1))$. Then we know that $\Gamma(C) + [0,1]^2$ and $\Gamma(C') + [0,1]^2$ both tile $\mathbb{R}^2$. Let $P_0 = \{(s_1, y) : 0 \leq y < 1\}$ and $P_1 = \{(s_1 + 1, y) : 0 \leq y < 1\}$. Note that $\Gamma(P_0) = \Gamma(\{(s_1, 0)\})$. Moreover,

$$\Gamma(C) = \Gamma(P_0) \cup \Gamma(C \setminus P_0), \qquad \Gamma(C') = \Gamma(C' \setminus P_1) \cup \Gamma(P_1)$$

and since $C \setminus P_0 = C' \setminus P_1$, $\Gamma(P_0) = \Gamma(P_1)$. We have

$$T_{C'}(\Gamma(P_1)) \subset \{(s_1 + 1, y),\ 0 \leq y < 1\}$$

but since $(s_2, 0) \in C'$, we must have $T_{C'}(\Gamma(P_1)) = (s_1 + 1, 0)$ by Lemma 4.4. Since

$$\Gamma(P_0) = \{(j + \mu_{k,0}, k) : j, k \in \mathbb{Z},\ t_k = s_1\}$$

and $\pi_2(\Gamma(P_0)) = \pi_2(\Gamma(P_1))$, where π_2 is the projection to the second coordinate, we have

$$\Gamma(\{(1 + s_1, 0)\}) = \Gamma(P_1) = \{(j + \mu_{k,1}, k) : j, k \in \mathbb{Z},\ t_k = s_1\},$$

for some constants $\mu_{k,1}$ with $0 \leq \mu_{k,1} < 1$ using Proposition 3.2. Applying this argument to $s_1 = 0$ and $s_2 = t$, we obtain that

$$\Lambda \cap \big(\{1\} \times [0,1) \times \mathbb{R}^2\big) = \{(j + \mu_{k,1}, k) : j, k \in \mathbb{Z},\ t_k = 0\}.$$

Similar arguments applied to $s_1 = s$ and $s_2 = 1$ show that, for any $s \in T$, we have

$$\Lambda \cap \big(\{s + 1\} \times [0,1) \times \mathbb{R}^2\big) = \{(j + \mu_{k,1}, k) : j, k \in \mathbb{Z},\ t_k = s\},$$

and that $\Lambda \cap \big(\{s + 1\} \times [0,1) \times \mathbb{R}^2\big)$ is empty if $s \in [0,1) \setminus T$. The same idea can also be used to show the existence of constants $\mu_{k,-1}$ with $0 \leq \mu_{k,1} < 1$ such that

$$\Lambda \cap \big(\{s - 1\} \times [0,1) \times \mathbb{R}^2\big) = \begin{cases} \{(j + \mu_{k,-1}, k) : j, k \in \mathbb{Z},\ t_k = s\}, & s \in T, \\ \emptyset, & s \in [0,1) \setminus T, \end{cases}$$

and, more generally using induction, that, for any $m \in \mathbb{Z}$, we can find constants $\mu_{k,m}$ with $0 \leq \mu_{k,m} < 1$ such that

$$\Lambda \cap \big(\{s+m\} \times [0,1) \times \mathbb{R}^2\big) = \begin{cases} \{(j+\mu_{k,m}, k) : j,k \in \mathbb{Z},\ t_k = s\}, & s \in T, \\ \emptyset, & s \in [0,1) \setminus T. \end{cases}$$

This proves our claim. □

We can now complete the proof of the main result of this section which gives a characterization for the subsets Λ of $\mathbb{R}^4$ with the property that the associated set of time-frequency shifts applied to the window $\chi_{[0,1]^2}$ yields an orthonormal basis for $L^2(\mathbb{R}^2)$.

Proof of Theorem 1.4. It follows from Lemma 4.4 that either all $T_{[0,1)^2}(\lambda_1, \lambda_2)$, $(\lambda_1, \lambda_2) \in \Gamma([0,1)^2)$ are either of the form $\{(t,0)\}$ or all are of the form $\{(0,t)\}$ with some $t \neq 0$. In the first case, we deduce from Lemma 4.6 that

$$\Lambda \cap (\mathbb{R} \times [0,1) \times \mathbb{R}^2) = \{(m + t_k, 0, j + \mu_{k,m}, k) : j,k,m \in \mathbb{Z}\}$$

for certain numbers t_k and $\mu_{k,m}$ in the interval $[0,1)$. We now show that Λ will be of the first of the two possible forms given in the theorem. (Similarly, the second form follows from the second case of Lemma 4.6.)

Letting $C = [0,1)^2$ and $C' = [0,1) \times (0,1]$, we note that both $\Gamma(C) + [0,1]^2$ and $\Gamma(C') + [0,1]^2$ tile $\mathbb{R}^2$ but $\Gamma((0,1)^2)$ is empty. Hence, $\Gamma(C') = \Gamma(\{(x,1) : 0 \leq x < 1\})$. It means that any set $T_{C'}(\lambda_1, \lambda_2)$ with $(\lambda_1, \lambda_2) \in \Gamma(C')$ is of the form $\{(t,1)\}$ for some $t = t(\lambda_1, \lambda_2)$ with $0 \leq t < 1$. We now have two possible cases: either the cardinality of $T_{C'}(\Gamma(C')$ is larger than one or equal to one. In the first case, we can find two distinct elements of $T_{C'}(\Gamma(C'))$ and we can then replicate the proof of Lemma 4.6 to obtain that

$$\Lambda \cap (\mathbb{R} \times [1,2) \times \mathbb{R}^2) = \{(m + t_k, 1, j + \mu_{k,m,1}, k) : j,k \in \mathbb{Z}\}.$$

In the other case, $T_{C'}(\Gamma(C')) = \{(t_1, 1)\}$ for some t_1 with $0 \leq t_1 < 1$. If we translate C' horizontally and use the same argument as in the proof of Lemma 4.6, we see that

$$\Lambda \cap (\mathbb{R} \times [1,2) \times \mathbb{R}^2) = \{(m + t_1, 1)\} \times \Lambda_{m,1},$$

where $\Lambda_{m,1}$ is a spectrum for the unit square $[0,1]^2$. This last property is equivalent to $\Lambda_{m,1} + [0,1]^2$ being a tiling of $\mathbb{R}^2$ by the result in [12].

We can then prove the theorem inductively by translating the square C' in the vertical direction using integer steps. □

References

[1] G. Ascensi, H.G. Feichtinger, N. Kaiblinger, Dilation of the Weyl symbol and the Balian–Low theorem, Trans. Amer. Math. Soc. 366 (2014) 3865–3880.
[2] X.-R. Dai, Q.-Y. Sun, The abc-problem for Gabor systems, preprint, http://arxiv.org/abs/1304.7750.
[3] B. Fuglede, Commuting self-adjoint partial differential operators and a group theoretic problem, J. Funct. Anal. 16 (1974) 101–121.

[4] J.-P. Gabardo, Tight Gabor frames associated with non-separable lattices and the hyperbolic Secant, Acta Appl. Math. 107 (2009) 49–73.
[5] D. Gabor, Theory of communication, J. Inst. Elec. Eng. (London) 93 (1946) 429–457.
[6] K. Gröchenig, Foundations of Time-Frequency Analysis, Appl. Numer. Harmon. Anal., Birkhäuser, Boston, Basel, Berlin, 2001.
[7] K. Gröchenig, J. Stöckler, Gabor frames and totally positive functions, Duke Math. J. 162 (2013) 1003–1031.
[8] Q. Gu, D. Han, When a characteristic function generates a Gabor frame, Appl. Comput. Harmon. Anal. 24 (2008) 290–309.
[9] A. Janssen, Zak transform with few zeros and the ties, in: Advances in Gabor Analysis, in: Appl. Numer. Harmon. Anal., Birkhäuser, Boston, MA, 2003, pp. 31–70.
[10] A. Janssen, On generating tight Gabor frames at critical density, J. Fourier Anal. Appl. 9 (2003) 175–214.
[11] M. Kolountzakis, The study of translational tiling with Fourier analysis, in: Fourier Analysis and Convexity, in: Appl. Numer. Harmon. Anal., Birkhäuser, Boston, 2004, pp. 131–187.
[12] J. Lagarias, J. Reeds, Y. Wang, Orthonormal bases of exponentials for the n-cubes, Duke Math. J. 103 (2000) 25–37.
[13] J.-L. Li, On characterization of spectra and tilings, J. Funct. Anal. 213 (2004) 31–44.
[14] Y.M. Liu, Y. Wang, The uniformity of non-uniform Gabor bases, Adv. Comput. Math. 18 (2003) 345–355.
[15] Y. Lyubarskii, Frames in the Bargmann Space of Entire Functions. Entire and Subharmonic Functions, Adv. Soviet Math., vol. 11, Amer. Math. Soc., Providence, RI, 1992, pp. 167–180.
[16] J. Ramanathan, T. Steger, Incompleteness of sparse coherent states, Appl. Comput. Harmon. Anal. 2 (1995) 148–153.
[17] K. Seip, R. Wallsten, Density theorems for sampling and interpolation in the Bargmann–Fock space, II, J. Reine Angew. Math. 429 (1992) 107–113.
[18] T. Tao, Fuglede's conjecture is false in 5 or higher dimensions, Math. Res. Lett. 11 (2004) 251–258.

Journal of Mathematics and Statistics

Original Research Paper

Probabilistic Estimates of the Largest Strictly Convex Singular Values of Pregaussian Random Matrices

Yang Liu

Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA

Article history
Received: 28-02-2015
Revised: 27-03-2015
Accepted: 30-03-2015

Abstract: In this study, the p-singular values of random matrices with Gaussian entries defined in terms of the l_p-p-norm for $p>1$, as is studied. Mainly, using analytical techniques, we show the probabilistic estimate, precisely, the decay, on the upper tail probability of the largest strictly convex singular values, when the number of rows of the matrices becomes very large and the lower tail probability of theirs as well. These results provide probabilistic description or picture on the behaviors of the largest p-singular values of random matrices in probability for $p>1$. Also, we show some numerical experiential results, which verify the theoretical results.

Keywords: Probability, Random Matrices, Singular Value, Banach Norm

Introduction

The largest singular value and the smallest singular value of random matrices in l_2-norm, including Gaussian random matrices, Bernoulli random matrices, subgaussian random matrices, etc, have attracted major research interest in recent years and have applications in compressed sensing, a technique for recovering sparse or compressible signals. For instance, (Soshnikov, 2002; Soshnikov and Fyodorov, 2004) studied the largest singular value of random matrices and (Rudelson and Vershynin, 2008a; 2008b; Tao and Vu, 2010) and some others, studied the smallest singular values.

In the study of the asymptotic behavior of eigenvalues of symmetric random matrices, Wigner symmetric matrix is a typical example, whose upper (or lower) diagonal entries are independent random variables with uniform bounded moments. Wigner proved in (Wigner, 1958) that the normalized eigenvalues are asymptotically distributed in the semicircular distribution. Precisely, let A be a symmetric gaussian random matrix of size $n\times n$ whose upper diagonal entries are independent and identically-distributed copies of the standard gaussian random variable, then the empirical distribution function of the eigenvalues of $\frac{1}{\sqrt{n}}A$ is asymptotically:

$$p(x) := \begin{cases} \frac{1}{2\pi}\sqrt{4-x^2}dx & for\ |x| \le 2 \\ 0 & for\ |x| > 2 \end{cases} \quad (1.1)$$

As the matrix size n goes to infinity. This is the well-known Wigner's Semicircle law, which provides the precise description of the statistical behavior of eigenvalues of matrix of large size. In another case, for a random matrix whose entries are independent and identically-distributed (i.i.d.) copies of a complex random variable with mean 0 and variance 1, Tao and Vu, (2008; Tao *et al*., 2010) that the eigenvalues of $\frac{1}{\sqrt{n}}$ a converges to the uniform distribution on the unit circle as n goes to ∞ and that holds not only for the random matrices with real entries but also for complex entries. Their result has also generalized (Girko, 1985) and solved the circular law conjecture open since the 1950's, that the smallest eigenvalue converges to the uniform distribution over the unit disk as n tends to infinity (Bai, 1997).

The largest singular values of matrices are actually their p-norm, which, from a geometric perspective, has connectionsa with the Minkowski space, complex l^p space, in differential geometry, for which one can refer to (Liu, 2013; 2011), because one can view the p-norm of a matrix as a generalization of the p-norm of a vector.

For random matrices whose entries are i.i.d. random variable satisfying certain moment conditions, the largest singular value was studied in (Geman, 1980; Yin *et al*., 1988). Tracy and Widom (1996) that the limiting law of largest eigenvalue distributions of Gaussian Orthogonal Ensemble (GOE) is given in terms of a particular Painlevé II function, which is the well-known Tracy-Widom law. Furthermore, the distribution of the

eigenvalue of Wishart matrices, $W_{N,n} = AA^*$, where $A = A_{N,n}$ is a Gaussian random matrix of size $N \times n$, was studied in (Johansson, 2000; Johnstone, 2001). They showed that the distribution of largest eigenvalue of Wishart matrices converges to the Tracy-Widom law as $\frac{n}{N}$ tend to some positive constant. More generally, the non-gaussian random matrices were studied in (Soshnikov, 2002). Seginer (2000) compared the Euclidean operator norm of a random matrix with i.i.d. mean zero entries to the Euclidean norm of its rows and columns. Later, (Latala, 2005) gave the upper bound on the expectation (or average value) of largest singular value namely the norm of any random matrix whose entries are independent mean zero random variables with uniformly bounded fourth moment.

The condition number, which is the ratio of the largest singular value over the smallest singular value of a matrix, is critical to the stability of linear systems. In (Edelman, 1988), the distribution of the condition number of Gaussian random matrices, was particularly investigated in numerical experiments. As a typical example of subgaussian random matrices, the invertibility of Bernoulli random matrices was also studied. Tao and Vu (2007) the probability of Bernoulli random matrices to be singular is shown to be at most $\left(\frac{3}{4}+o(1)\right)^n$, where n is the size of the matrices. Their result shows that the probability of the smallest singular value of Bernoulli random matrices to be zero is exponentially small as n tends to infinity. Recently, the singularity probability $\left(\frac{3}{4}+o(1)\right)^n$ has been improved to $\left(\frac{1}{2}+o(1)\right)^n$ by (Bourgain *et al*., 2010).

The recent studies of the smallest singular value have also been motivated, in a large sense, by some open questions or conjectures. Spielman and Teng (2002) the following conjecture was proposed in the International Congress of Mathematicians in 2002.

Conjecture 1.1

Let ξ be Bernoulli random variable, in other words, $P(\xi-1)=P(\xi=-1)=\frac{1}{2}$. Then:

$$P\left(s_n\left(M(\xi)\le\frac{t}{\sqrt{n}}\right)\le t+c^n\right) \tag{1.2}$$

for all $t>0$ and some $0<c<1$.

In the breakthrough work on the estimate on the smallest singular value, (Rudelson and Vershynin, 2008a), Rudelson and Vershynin obtained the upper tail probabilistic estimate on the smallest value in l_2-norm for square matrices of centered random variables, with unit variance and appropriate moment assumptions. In particular, they proved the Spielman-Teng conjecture up to a constant. The lower tail probabilistic estimate on the smallest value in l_2-norm for square matrices was estimated in (Rudelson and Vershynin, 2008b). These results have shown that the smallest singular value of the $n \times n$ subgaussian random matrices is of order $n^{-\frac{1}{2}}$ in high probability for large n. In a more explicit way, the distribution of the smallest singular value of random was given in (Tao and Vu, 2010) by using property testing from combinatorics and theoretical computer science. The pregaussian matrices were used to recover sparse image in (Rauhut, 2010) and matrix recovery, on which one can refer to (Oymak *et al*., 2011; Lai *et al*., 2012). Very recently, Rudelson and Vershynin (2010) gave a comprehensive survey on the extreme singular values of random matrices.

It is well-known that the classic singular value is defined in terms of l_2-norm, then a natural question would be what if one defines the singular value by the l_q-quasinorm for $0<q\le1$ and l_p-norm for $p>1$. There were some remarkable results by other researchers on the largest singular values of random matrices in the l_2-norm. Geman (1980; Yin *et al*., 1988) showed that the largest singular value of random matrices of size $m \times N$ with independent entries of mean 0 and variance 1 tends to $\sqrt{m}+\sqrt{N}$ almost surely. The largest and smallest q-singular values of pregaussian random matrices for $0<q\le1$ were studied in (Lai and Liu, 2014), which has applications in a technique of signal processing (Foucart and Lai, 2010; 2009; Lai and Liu, 2011) and other areas. Similar to the q-singular value when $0<q\le1$, the strictly convex largest p-singular value, in which $p>1$, can be defined and we will show the probabilistic estimate, precisely, the decay, on the upper tail probability of the largest strictly convex p-singular value, when the number of rows of the matrices becomes very large and the lower tail probability of theirs as well. These results provide probabilistic description or picture on the behaviors of the largest p-singular values of random matrices in probability.

The Largest *p*-Singular Value

The p-singular values of a matrix, in general, can be defined in the way of maximum of minimums or supremum of infimums. In largest p-sigular values can be defined as follows:

Definition 2.1

For an $m \times N$ matrix A, the largest p-singular value of A denoted as $s_1^{(p)}(A)$ is defined as:

$$s_1^{(p)}(A) := sup\left\{\| Ax \|_p : x \in \mathbb{R}^N \; with \| x \|_p = 1\right\} \tag{2.1}$$

For given $p>1$.

Lai and Liu (2014), the following lemma on a linear bound for partial binomial expansion was established.

Lemma 2.2

For every positive integer n:

$$\sum_{k=\left\lfloor \frac{n}{2} \right\rfloor+1}^{n} \binom{n}{k} x^k (1-x)^{n-k} \le 8x \tag{2.2}$$

For all $x \in [0, 1]$.

The above lemma can be applied to estimate probabilities.

Lemma 2.3

Suppose $\xi_1, \xi_2, \cdots, \xi_n$ are i.i.d copies of a random variable ξ, then for any $\varepsilon>0$:

$$P\left(\sum_{i=1}^{n} |\xi_i|^p \le \frac{n\varepsilon}{2}\right) \le 8P\left(|\xi| \le \varepsilon\right) \tag{2.3}$$

For any given $p>1$.

Proof. Given p>1, we have the relation on the probability events that:

$$\left\{(\xi_1,\ldots\xi_n): \sum_{i=1}^{n} |\xi_i|^p \le \frac{m\varepsilon}{2}\right\} \tag{2.4}$$

Is contained in:

$$\bigcup_{k=\left\lfloor \frac{n}{2} \right\rfloor+1}^{n} \left\{(\xi_1,\ldots\xi_n): |\xi_{i1}|^p \le \varepsilon,\ldots |\xi_{ik}|^p > \varepsilon,\ldots |\xi_{in}|^p > \varepsilon\right\} := \varepsilon \tag{2.5}$$

where, $\{i_1, i_2, \cdots, i_k\}$ is a subset of $\{1, 2 \cdots, n\}$ and $\{i_{k+1}, \cdots, i_n\}$ is its complement.

Let $x = \mathbb{P}(|\xi_1|^p \le \varepsilon)$, then by the union probability:

$$P(\varepsilon) = \sum_{k=\left\lfloor \frac{n}{2} \right\rfloor+1}^{n} \binom{n}{k} x^k (1-x)^{n-k} \tag{2.6}$$

And applying Lemma 2.2, we have:

$$P(\varepsilon) \le 8x = 8P\left(|\xi_1| \le \varepsilon\right) \tag{2.7}$$

Since the event (2.4) is contained in the event (2.5):

$$P\left(\sum_{i=1}^{n} |\xi_i|^p \le \frac{n\varepsilon}{2}\right) \le P(\varepsilon) \le 8P\left(|\xi_1|^p \le \varepsilon\right) \tag{2.8}$$

To estimate the lower tail probability of the largest p-singular value, we have the following theorem on the lower tail probability of the largest p-singular value for $p>1$.

Theorem 2.4

Let ξ be a pregaussian variable normalized to have variance 1 and A is an $m \times N$ matrix with i.i.d. copies of ξ in its entries, then for every p>1 and any $\varepsilon>0$, there exists $\gamma>0$ such that:

$$P\left(s_1^{(p)}(A) \le \gamma m^{\frac{1}{p}}\right) \le \varepsilon \tag{2.9}$$

Which γ only depends on p, ε and the pregaussian variable ξ.

Proof. Since a_{ij} is pregaussian with variance 1, then any $\varepsilon>0$, there is some $\delta>0$, such that:

$$P\left(|a_{ij}|^p \le \delta\right) \le \frac{\varepsilon}{8} \tag{2.10}$$

But we know:

$$s\frac{(p)}{1}(A) \ge \left(\sum_{i=1}^{m} |a_{ij}|^p\right)^{\frac{1}{p}} \tag{2.11}$$

For all j, because by the definition of the largest p-singular value 2.1, choosing x to be the standard basis vectors of $\mathbb{R}^N$ gives us $\max_j \left(\sum_{i=1}^{m} |a_{ij}|^p\right)^{\frac{1}{p}} \le s_1^{(p)}(A)$.

Therefore, by Lemma 2.3:

$$\begin{aligned} &P\left(s_1^{(p)}(A) \le \left(\frac{\delta}{2}\right)^{\frac{1}{p}} m^{\frac{1}{p}}\right) \le P\left(\sum_{i=1}^{m} |a_{ijo}|^p \le \frac{m\delta}{2}\right) \\ &\le 8P\left(|a_{ij}|^p \le \delta\right) \le \varepsilon \end{aligned} \tag{2.12}$$

Thus let $\gamma = \left(\frac{\delta}{2}\right)^{\frac{1}{p}}$, then (2.9) follows.

For the upper tail probability of the largest p-singular value, $p>1$, we can derive the following lemma first by using the Minkowski inequality and discrete Hölder inequality.

Lemma 2.5

For $p \ge 1$, (2.1) defines a norm on the space of $m \times N$ matrices and:

$$\max_j \| a_j \|_p \le s_1^{(p)}(A) \le N^{\frac{p-1}{p}} \max_j \| a_j \|_p \tag{2.13}$$

In which $aj, j = 1, 2, \cdots, N$, are the column vectors of A.

Applying the above lemma, an estimate we can derive easily for Bernoulli random matrices, whose every entry equals to 1 or-1 with equal probability (Tao and Vu, 2009), is the following theorem on the upper tail probability of the largest *p*-singular value of Bernoulli matrices for $p>1$.

Theorem 2.6

Let ξ be a Bernoulli random variable normalized to have variance 1 and A be an $m\times N$ matrix with i.i.d. copies of ξ in its entries, then:

$$m^{\frac{1}{p}} \le s_1^{(p)}(A) \le m^{\frac{1}{p}} N^{\frac{p-1}{p}} \tag{2.14}$$

One may conjecture that the bound might be $m^{\frac{1}{p}}$. However, considering the Bernoulli matrices, whose entries are in Bernoulli distribution, as special subgaussian matrices, the expectation of the largest *p*-singular value may not be $m^{\frac{1}{p}}$. Indeed, let A be an $m\times m$ Bernoulli matrix and x be a non-zero vector in $\mathbb{R}^m$. The expectation of the largest *p*-singular value:

$$E\left(s_1^{(p)}(A)\right) \le E\frac{\| Ax \|_p}{\| x \|_p} \tag{2.15}$$

For all $x\in\mathbb{R}^m$ and particularly for $x = (1,\cdots, 1) \in \mathbb{R}^m$, we have:

$$E\frac{\| Ax \|_p}{\| x \|_p} == n^{-\frac{1}{p}} E\left(\sum_{i=1}^{n} |\in_{i1} +\ldots+ \in_{in}|^p\right)^{\frac{1}{p}} \tag{2.16}$$

Now let X_i: $= \in_{i1} +\cdots+\in_{in}$, then $E\left(\sum_{i=1}^{n}\in_{i1} +\ldots+ \in_{in}|^p\right)^{\frac{1}{p}}$ is the expectation of the l_p-norm of the vector $(X_1, X_2, \cdots, X_n)$.

We also have the following result on the upper tail probability of the largest *p*-singular value of Bernoulli matrices for $p>1$.

Theorem 2.7

Let A be an $m\times m$ Bernoulli matrix with every entry equal to 1 or-1 with equal probability, then one has:

$$P\left(s_1^{(p)}(A) \ge K_m\right) \le exp(-cm) \tag{2.17}$$

For some $K>0$ and some absolute constant $c>0$.

Proof. Let $A = (\in_{ij})_{m\times m}$ and S_p^{m-1} be the unit sphere with respect to l_p-norm in $\mathbb{R}^m$, then for any $x \in S_p^{m-1}$, by the convexity of the function $f(t) := t^p$ for $p>1$:

$$\begin{aligned} \| Ax \|_p &= \left(\sum_{i=1}^{m}\left|\sum_{j-i}^{m}\in_{ij} x_j\right|^p\right)^{\frac{1}{p}} \le m^{\frac{p-1}{p}} \\ &\left(\sum_{i=1}^{m}\sum_{j=1}^{m}|\in_{ij} x_j|^p\right)^{\frac{1}{p}} = m\| x \|_p = m \end{aligned} \tag{2.18}$$

Therefore we have:

$$E\| Ax \|_p \le m \tag{2.19}$$

For all $x \in S_p^{m-1}$. By Chernoff bound, we get:

$$\begin{aligned} &P\left(\| Ax \|_p \ge Km\right) \le P\left(\| Ax \|_p \ge KE\| Ax \|_p\right) \\ &\le exp(-cKm) \end{aligned} \tag{2.20}$$

For any $K>2$ and some absolute constant $c>0$.

By Lemma 4.10 in (Pisier, 1999), there is a subset N which is a δ-net of S_p^{m-1} with cardinality:

$$card(N) \le \left(1+\frac{2}{\delta}\right)^m \tag{2.21}$$

Finally, using the union bound of probability and an approximation of any point on the sphere by points of the δ-net, we obtain (2.17).

For the rectangular matrices, we have the following theorem on the upper tail probability of the largest *p*-singular value of rectangular matrices for $1<p\le 2$.

Theorem 2.8

Let ξ be a pregaussian variable normalized to have variance 1 and A is an $m\times N$ matrix with i.i.d. copies of ξ in its entries, then for every $1<p\le 2$ and any $\varepsilon>0$, there exists $K>0$ such that:

$$P\left(s_1^{(p)}(A) \ge K\left(m^{\frac{1}{p}} + m^{\frac{2-p}{2p}} N^{\frac{1}{2}}\right)\right) \le \varepsilon \tag{2.22}$$

where, K only depends on p, ε and the pregaussian variable ξ.

Proof. By the discrete Hölder inequality and the definition of the largest *p*-singular value:

$$\begin{aligned} s_1^{(p)}(A) &= \sup_{x\in\mathbb{R}^N, x\neq 0}\frac{\| Ax \|_p}{\| x \|_p} \le \sup_{x\in\mathbb{R}^N, x\neq 0} \\ &\frac{m^{\frac{1}{p}-\frac{1}{2}}\| Ax \|_2}{\| x \|_2} = m^{\frac{1}{p}-\frac{1}{2}} s_1^{(2)}(A) \end{aligned} \tag{2.23}$$

We also know that there exists $K>0$ such that:

$$P\left(s_1^{(2)}(A) \geq K\left(m^{\frac{1}{2}} + N^{\frac{1}{2}}\right)\right) \leq \varepsilon \tag{2.24}$$

Therefore, we have:

$$\begin{aligned} &P\left(s_1^{(2)}(A) \geq K\left(m^{\frac{1}{2}} + m^{\frac{1}{p}-\frac{1}{2}} N^{\frac{1}{2}}\right)\right) \\ &\leq P\left(s_1^{(2)}(A) \geq K\left(m^{\frac{1}{2}} + N^{\frac{1}{2}}\right)\right) \leq \varepsilon \end{aligned} \tag{2.25}$$

To have a full generalization, let us derive the following useful lemma.

In general, for the relation between $s_1^{(q)}$ and $s_1^{(q)}, \frac{1}{p} + \frac{1}{q} = 1, q > 1,$ one can deduce the following duality lemma on general rectangular matrices.

Lemma 2.9

For any $q \geq 1$ and $m \times N$ matrix A:

$$s_1^{(q)}(A) = s_1^{(p)}\left(A^T\right) \tag{2.26}$$

where, $\frac{1}{p} + \frac{1}{q} = 1$.

Proof. By the discrete Hölder inequality, we know that if $\frac{1}{p} + \frac{1}{q} = 1$ then:

$$\langle Ax, y \rangle \leq \| Ax \|_q \tag{2.27}$$

For all $x \in \mathbb{R}^N$ and $y \in \mathbb{R}^m$ with $\|y\|_p = 1$ and furthermore the equality holds for some y_0 with $\|y_0\|_p = 1$. Thus:

$$\| Ax \|_p = \sup_{y \in \mathbb{R}^m, \|y\|_p = 1} \langle Ax, y \rangle \tag{2.28}$$

By the definition of the largest q-singular value:

$$\begin{aligned} &s_1^{(q)}(A) = sup_{x \in \mathbb{R}^N, \|x\|_q = 1} \| Ax \|_q \\ &= sup_{x \in \mathbb{R}^N, \|x\|_q = 1} sup_{y \in \mathbb{R}^m, \|y\|_p = 1} \langle Ax, y \rangle \end{aligned} \tag{2.29}$$

In the same way, we also have:

$$s_1^{(p)}\left(A^T\right) = \sup_{y \in \mathbb{R}^m, \|y\|_{p=1}} \sup_{x \in \mathbb{R}^N, \|x\|_q = 1} \langle Ax, y \rangle \tag{2.30}$$

Finally, using $\langle Ax, y \rangle = \langle A^T y, x \rangle$ and switching the supremums, we get $s_1^{(q)}(A) = s_1^{(p)}\left(A^T\right)$.

We have the following remarks on the above lemma.

Remark 2.10

One can also obtain the above lemma the operator duality on the dual spaces.

Remark 2.11

The above lemma allows us to obtain the probabilistic estimates on $s_1^{(p)}(A)$ for $p>2$ by taking the transpose of A and using the estimates on $s_1^{(q)}\left(A^T\right)$.

Thus using the duality lemma, we obtain.

Theorem 2.12

(Lower tail probability of the largest p-singular value of rectangular matrices, p>2). Let ξ be a pregaussian random variable normalized to have variance 1 and A be an $m \times N$ matrix with i.i.d. copies of ξ in its entries, then for every p>2 and any $\varepsilon>0$, there exists $\gamma>0$ such that:

$$P\left(s_1^{(p)}(A) \leq \gamma m^{\frac{p-1}{p}}\right) \leq \varepsilon \tag{2.31}$$

which, γ only depends on p, ε and the pregaussian random variable ξ.

Moreover, we have the upper tail probability of the largest p-singular value of rectangular matrices for $p>2$.

Theorem 2.13

(Upper tail probability of the largest p-singular value of rectangular matrices, p>2). Let ξ be a pregaussian variable normalized to have variance 1 and A is an $m \times N$ matrix with i.i.d. copies of ξ in its entries, then for every p>2 and any $\varepsilon>0$, there exists K>0 such that:

$$P\left(s_1^{(p)}(A) \geq K\left(N^{\frac{p-1}{p}} + m^{\frac{1}{2}} N^{\frac{p-2}{2p}}\right)\right) \leq \varepsilon \tag{2.32}$$

where, K only depends on p, ε and the pregaussian variable ξ.

Numerical Experiments

In general, matrix p-norms are, in fact, NP-hard to approximate if $p \neq 1,2,\infty$, on which one can refer to (Hendrickx and Olshevsky, 2010; Liu, 2014; Higham, 1992). In this section, however, we would like to show the results from some numerically computable experiments on the p-singular value for $p>1$ and q-singular value for $0<q\leq 1$ of random matrices.

For $p = 2$, we plot the largest 2-singular value of Gaussian random matrices of size $n \times n$, where n runs

from 1 through 100. Figure 1 this graph shows that the 2-singular value is $O\left(\sqrt{n}\right)$.

For p = 1, in the first numerical experiment we plot the largest 1-singular value of Gaussian random of size $n\times n$, where n runs from 1 through 100. Figure 2 the graph shows that the largest 1-singular value is $O(n)$.

In the second numerical experiment for p = 1, we plot the largest 1-singular value of Gaussian random matrices of size $n\times n$, where n runs from 1 through 200. Figure 3 the graph shows that the largest 1-singular value is $O(n)$.

In the third experiment for p = 1, we plot the largest 1-singular value of Gaussian random matrices of size $n\times n$, where n runs from 1 through 400. Figure 4 the graph shows that the largest 1-singular value is $O(n)$.

For $p = \infty$, we plot the largest ∞-singular value of Gaussian random matrices of size $n\times n$, where n runs from 1 through 500. Figure 5 this graph shows that the ∞-singular value is $O(n)$.

Higham (1992), the p-norm of a matrix of size m by n was estimated reliably in $O(mn)$ operations and an algorithm that can estimate the p-norm in a specific accuracy, within a factor of $n^{1-\frac{1}{p}}$, was provided. Using this algorithm, we plot the largest 4-singular value of Gaussian random matrices and Bernoulli random matrices of size $m\times n$, where m and n run from 1 through 81 Fig. 6 and 7.

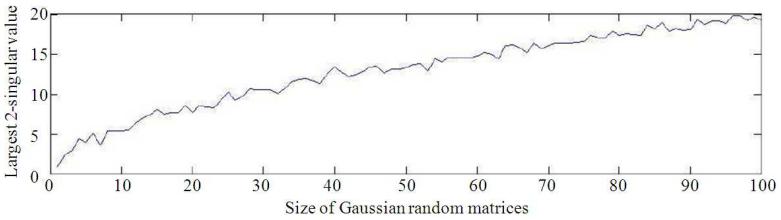


Fig.1. Largest 2-singular value of Gaussian random matrices

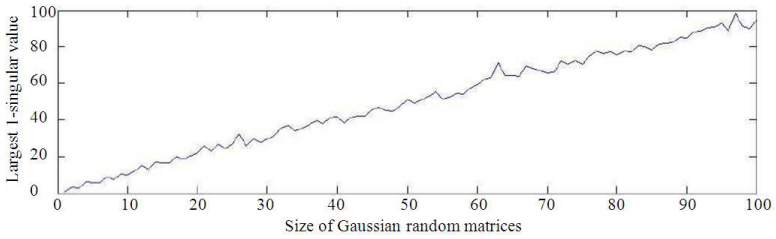


Fig. 2. Largest 1-singular value of Gaussian random matrices: Experiment 1

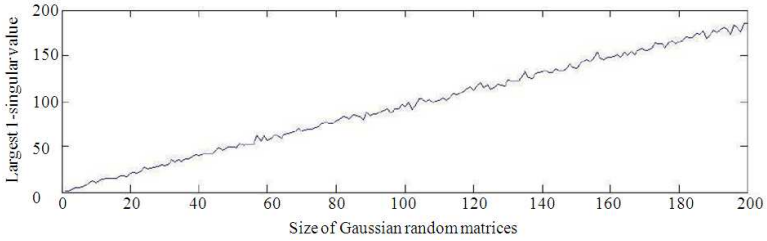


Fig. 3. Largest 1-singular value of Gaussian random matrices: Experiment 2

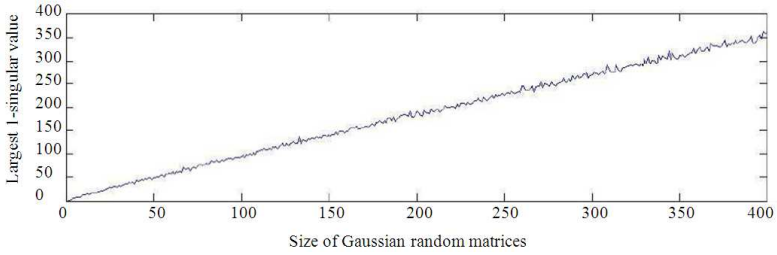


Fig. 4. Largest 1-singular value of Gaussian random matrices: Experiment 3

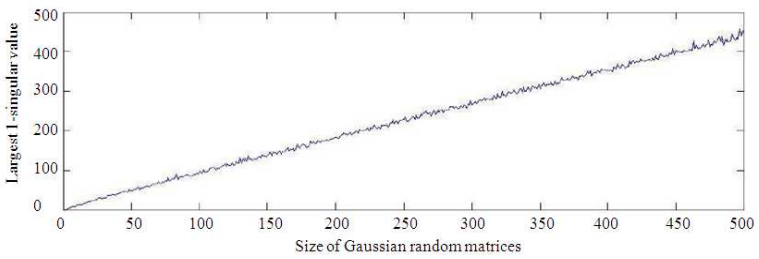


Fig. 5. Largest ∞-singular value of Gaussian random matrices

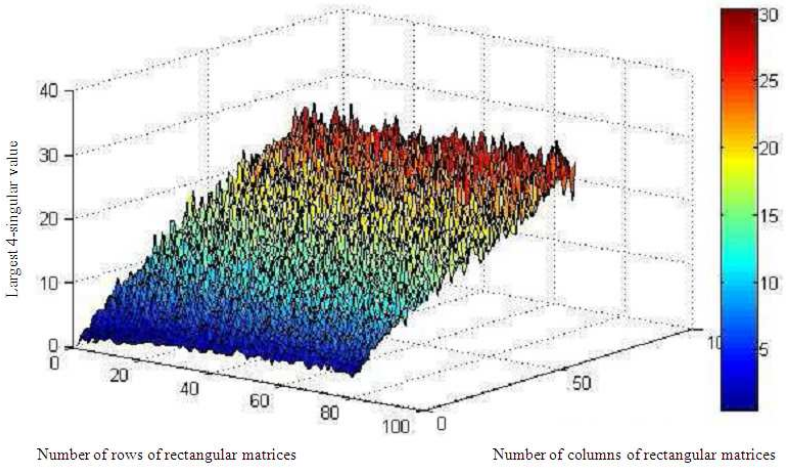


Fig. 6. Largest 4-singular value of Gaussian random matrices

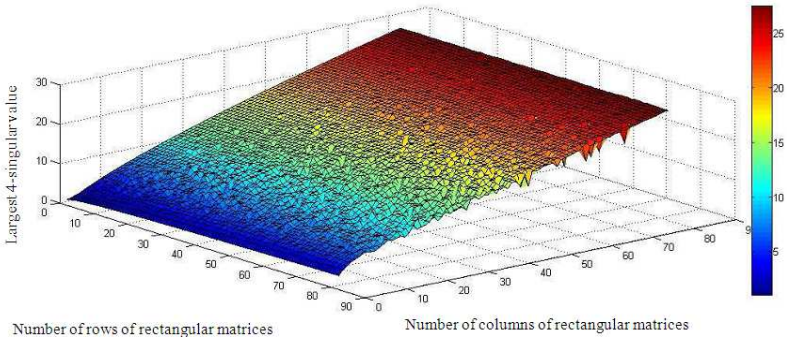


Fig. 7. Largest 4-singular value of Bernoulli random matrices

Acknowledgment

The author would like to thank Prof. M.J. Lai for suggesting the research problem. The author is partially supported by the Air Force Office of Scientific Research under grant AFOSR 9550-12-1-0455.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Bai, Z.D., 1997. Circular law. Ann. Probab., 25: 494-529.

Bourgain, J., V.H. Vu and P.M. Wood, 2010. On the singularity probability of discrete random matrices. J. Functional Analysis, 258: 559-603. DOI: 10.1016/j.jfa.2009.04.016

Edelman, A., 1988. Eigenvalues and condition numbers of random matrices. SIAM J. Matrix Anal. Appl., 9: 543-560. DOI: 10.1137/0609045

Foucart, S. and M.J. Lai, 2009. Sparsest solutions of underdetermined linear systems via ℓq-minimization for 0<q≤1. Applied Comput. Harmonic Analysis, 26: 395-407, 2009. DOI: 10.1016/j.acha.2008.09.001

Foucart, S. and M.J. Lai, 2010. Sparse recovery with pre-Gaussian random matrices. Studia Math., 200: 91-102. DOI: 10.4064/sm200-1-6

Geman, S., 1980. A limit theorem for the norm of random matrices. Ann. Probab., 8: 252-261. DOI: 10.1214/aop/1176994775

Girko, V.L., 1985. Circular law. Theory Probab. Appl., 29: 694-706. DOI: 10.1137/1129095

Hendrickx, J.M. and A. Olshevsky, 2010. Matrix p-norms are NP-hard to approximate if $p\neq1,2,\infty$. SIAM. J. Matrix Anal. Appl., 31: 2802-2812. DOI: 10.1137/09076773X

Higham, N.J., 1992. Estimating the matrixp-norm. Numerische Mathematik, 62: 539-555. DOI: 10.1007/BF01396242

Johansson, K., 2000. Shape fluctuations and random matrices. Commun. Math. Phys., 209: 437-476. DOI: 10.1007/s002200050027

Johnstone, I.M., 2001. On the distribution of the largest eigenvalue in principal components analysis. Ann. Statist., 29: 295-327.

Lai, M.J., S. Li, L.Y. Liu and H. Wang, 2012. Two results on the schatten p-quasi-norm minimization for low-rank matrix recovery.

Lai, M.J. and Y. Liu, 2011. The null space property for sparse recovery from multiple measurement vectors. Applied Comput. Harmonic Anal., 30: 402-406. DOI: 10.1016/j.acha.2010.11.002

Lai, M.J. and Y. Liu, 2014. The probabilistic estimates on the largest and smallest *q*-singular values of random matrices. Math. Comp. DOI: 10.1090/S0025-5718-2014-02895-0

Latala, R., 2005. Some estimates of norms of random matrices. Proc. Am. Math. Soc., 133: 1273-1282. DOI: 10.1090/S0002-9939-04-07800-1

Liu, Y., 2011. On the lagrangian subspaces of complex minkowski space. J. Math. Sci. Adv. Appl., 7: 87-93.

Liu, Y., 2013. On the kähler form of complex *lp* space and its lagrangian subspaces.

Liu, Y., 2014. Low rank approximations of linear operators in *p*-norms and their algorithms.

Oymak, S., K. Mohan, M. Fazel and B. Hassibi, 2011. A simplified approach to recovery conditions for low rank matrices. Proceedings of the IEEE International Symposium on Information Theory, Jul. 31-Aug. 5, IEEE Xplore press, St. Petersburg, pp: 2318-2322. DOI: 10.1109/ISIT.2011.6033976

Pisier, G., 1999. The Volume of Convex Bodies and Banach Space Geometry. 1st Edn., Cambridge University Press, Cambridge, ISBN-10: 052166635X, pp: 250.

Rauhut, H., 2010. Compressive Sensing and Structured Random Matrices. In: Theoretical Foundations and Numerical Methods for Sparse Recovery, Fornasier, M. (Ed.), Walter de Gruyter, New York, ISBN-10: 3110226154, pp: 1-92.

Rudelson, M. and R. Vershynin, 2008a. The least singular value of a random square matrix is $O(n^{-1/2})$. Comptes Rendus Mathematique, 346: 893-896. DOI: 10.1016/j.crma.2008.07.009

Rudelson, M. and R. Vershynin, 2008b. The littlewood-offord problem and invertibility of random matrices. Adv. Math., 218: 600-633. DOI: 10.1016/j.aim.2008.01.010

Rudelson, M. and R. Vershynin, 2010. Non-asymptotic theory of random matrices: Extreme singular values. Proceedings of the International Congress of Mathematicians, (ICM' 10), Hindustan Book Agency, New Delhi, pp: 1576-1602.

Seginer, Y., 2000. The expected norm of random matrices. Combinatorics Probability Comput., 9: 149-166. DOI: 10.1017/S096354830000420X

Soshnikov, A., 2002. A note on universality of the distribution of the largest eigenvalues in certain sample covariance matrices. J. Stat. Phys., 108:1033-1056. DOI: 10.1023/A:1019739414239

Soshnikov, A. and Y.V. Fyodorov, 2004. On the largest singular values of random matrices with independent cauchy entries. J. Math. Phys., 46: 033302-033316. DOI: 10.1063/1.1855932

Spielman, D. and S.H. Teng, 2002. Smoothed analysis of algorithms. Proceedings of the International Congress of Mathematicians, (ICM' 02), Beijing, pp: 597-606.

Tao, T. and V. Vu, 2007. On the singularity probability of random Bernoulli matrices. J. Am. Math. Soc., 20: 603-628. DOI: 10.1090/S0894-0347-07-00555-3

Tao, T. and V. Vu, 2008. Random matrices: The circular law. Commun. Contemp. Math., 10: 261-307. DOI: 10.1142/S0219199708002788

Tao, T. and V. Vu, 2009. On the permanent of random bernoulli matrices. Adv. Math., 220: 657-669. DOI: 10.1016/j.aim.2008.09.006

Tao, T. and V. Vu, 2010. Random matrices: The distribution of the smallest singular values. Geometric Functional Analysis, 20: 260-297. DOI: 10.1007/s00039-010-0057-8

Tao, T., V. Vu and M. Krishnapur, 2010. Random matrices: Universality of ESDs and the circular law. Ann. Probab., 38: 2023-2065. DOI: 10.1214/10-AOP534

Tracy, C.A. and H. Widom 1996. On orthogonal and symplectic matrix ensembles. Commun. Math. Phys., 177: 727-754. DOI: 10.1007/BF02099545

Wigner, E.P., 1958. On the distribution of the roots of certain symmetric matrices. Ann. Math., 67: 325-327.

Yin, Y.Q., Z.D. Bai and P.R. Krishnaiah, 1988. On the limit of the largest eigenvalue of the large dimensional sample covariance matrix. Probability Theory Related Fields, 78: 509-521. DOI: 10.1007/BF00353874

MATHEMATICS OF COMPUTATION
Volume 84, Number 294, July 2015, Pages 1775–1794
S 0025-5718(2014)02895-0
Article electronically published on October 30, 2014

THE PROBABILISTIC ESTIMATES ON THE LARGEST AND SMALLEST q-SINGULAR VALUES OF RANDOM MATRICES

MING-JUN LAI AND YANG LIU

Abstract. We study the q-singular values of random matrices with pre-Gaussian entries defined in terms of the ℓ_q-quasinorm with $0 < q \leq 1$. In this paper, we mainly consider the decay of the lower and upper tail probabilities of the largest q-singular value $s_1^{(q)}$, when the number of rows of the matrices becomes very large. Based on the results in probabilistic estimates on the largest q-singular value, we also give probabilistic estimates on the smallest q-singular value for pre-Gaussian random matrices.

1. Introduction

The extremal spectrum or the largest and smallest singular values of random matrices have been of interest to many research communities including numerical analysis and multivariate statistics. For example, the condition numbers of random matrices were of interest as early as in von Neumann and Goldstein'1947, [28] and Smale'1985, [19], and distribution of the largest and smallest eigenvalues of Wishart matrices was studied in Wishart'1928, [30]. Some estimates for the probability distribution of the norm of a random matrix transformation were obtained in Bennett, Goodman and Newman'1975, [2]. In 1988, Edelman presented a comprehensive study on the distribution of the condition numbers of Gaussian random matrices together with many numerical experiments (cf. [5]). In particular, Edelman explained several interesting applications of eigenvalues of random matrices in graph theory, the zeros of Riemann zeta functions, as well as in nuclear physics (cf. [6]). Indeed, the well-known semi-circle law (cf. Wigner'1962, [29]) states that the histogram for the eigenvalues of a large random matrix is roughly a semi-circle. To be more precise, let A be a Gaussian random matrix and $M(x)$ denote the proportion of eigenvalues of the Gaussian orthogonal ensemble $(A + A^T)/(2\sqrt{n})$ (the symmetric part of $A/\sqrt{n}$) that are less than x. Then the semi-circle law asserts that

$$\frac{d}{dx}M(x) \to \begin{cases} \frac{2}{\pi}\sqrt{1-x^2}, & \text{if } x \in [-1,1], \\ 0, & \text{otherwise.} \end{cases}$$

This interesting property has made a long lasting impact and attracted many researchers to extend and generalize the semi-circle law. See recent papers of

Received by the editor November 26, 2012 and, in revised form September 23, 2013.

2010 *Mathematics Subject Classification.* Primary 60B20; Secondary 60F10, 60G50, 60G42.

Key words and phrases. Random matrices, probability, pre-Gaussian random variable, generalized singular values.

The first author was partly supported by the National Science Foundation under grant DMS-0713807.

The second author was partially supported by the Air Force Office of Scientific Research under grant AFOSR 9550-12-1-0455.

Tao and Vu'2008, [24] and Rudelson and Vershynin'2010, [17] for new results and surveys and the references therein. It is known that the largest eigenvalue of $M_s = \frac{1}{s} V_{n\times s}(V_{n\times s})^T$ converges to $(1+\sqrt{y})^2$ almost surely (cf. Geman'1980, [10]) and the smallest eigenvalue converges to $(1-\sqrt{y})^2$ almost surely (cf. Silverstein'1985, [18]), where $V_{n\times s}$ is a Gaussian random matrix of size $n \times s$ with $n/s \to y \in (0,1]$ and $V_{n\times s}(V_{n\times s})^T$ is called a Wishart matrix. The behavior of the largest singular value of random matrices A with i.i.d. entries is well studied. If a random variable ξ has a bounded fourth moment, then the largest eigenvalue $s_1(A)$ of an $n \times n$ random matrix A with i.i.d. copies of ξ satisfies the following property:

$$\lim_{n\to\infty} \frac{s_1(A)}{\sqrt{n}} = 2\sqrt{\mathbb{E}\xi^2}$$

almost surely. See, e.g., Yin, Bai, Krishnaiah'1988, [31] and Bai, Silverstein and Yin'1988, [1]. The bounded fourth moment is necessary and sufficient in this case. However, the behavior of the smallest singular value for general random matrices has been much less known. Although Edelman showed that for every $\epsilon > 0$, the smallest eigenvalue $s_n(A)$ of Gaussian random matrix A of size $n \times n$ has

$$\mathbb{P}\left(s_n(A) \le \frac{\epsilon}{\sqrt{n}}\right) \le \epsilon$$

for any $\epsilon > 0$, the probability estimates for $s_n(A)$ for general random matrix A were not known until the results in Rudelson and Vershynin'2008, [14]. In fact, Rudelson in [16] presented a less accurate probability estimate for $s_n(A)$, and soon both Rudelson and Vershynin found a simpler proof of much accurate estimate in [15]. More precisely, Rudelson and Vershynin first showed (cf. [15]) the following results:

Theorem 1.1. *If A is a matrix of size $n \times n$ whose entries are independent random variables with variance* 1 *and bounded fourth moment, then*

$$\lim_{\epsilon\to 0_+} \limsup_{n\to\infty} \mathbb{P}\left(s_n(A) \le \frac{\epsilon}{\sqrt{n}}\right) = 0.$$

Furthermore, in Rudelson and Vershynin'2008, [14], they presented a proof of the following

Theorem 1.2. *Let A be an $n \times n$ matrix whose entries are i.i.d. centered random variables with unit variance and fourth moment bounded by B. Then*

$$\lim_{K\to +\infty} \limsup_{n\to\infty} \mathbb{P}\left(s_n(A) \ge \frac{K}{\sqrt{n}}\right) = 0.$$

These two results settled down a conjecture by Smale in [18] (the results on the Gaussian case were established by Edelman and Szarek; see [6] and [22]). More precise estimates for largest and smallest eigenvalues are given for sub-Gaussian random matrices, Bernoulli matrices, covariance matrices, and general random matrices of the form $M + A$ with deterministic matrix M and random matrix A in the last ten years. See, e.g. [25], [20], [14], [26], [23] and the references in [17].

In this paper, we extend these studies on the probability estimate of the largest and smallest singular values of random matrices in the ℓ_2-norm and give estimates for these extremal spectra in the setting of the ℓ_q-quasinorm for $0 < q \le 1$. Not only is it interesting to know if the probability estimates for largest and smallest

singular values of random matrices in the ℓ_2-norm can be extended to the setting of the ℓ_q-quasinorm, there are also some definite advantages of using the general ℓ_q-quasinorm when studying the restricted isometry property of random matrices as suggested in Chartrand and Steneva'2008, [4], Foucart and Lai'2009, [8] and Foucart and Lai'2010, [9]. In addition to Gaussian and sub-Gaussian random matrices, we would like to study the probability estimates for pre-Gaussian random matrices. A random variable ξ is pre-Gaussian if ξ has mean zero and the moment growth condition $\mathbb{E}(|\xi|^k) \leq k!\lambda^k/2$, i.e. $(\mathbb{E}(|\xi|^k))^{1/k} \leq C\lambda k$ for $k \geq 1$ (cf. Buldygin and Kozachenko'2000, [3]). Note that the moment growth condition for a sub-Gaussian random variable η is $\left(\mathbb{E}\left(|\eta|^k\right)\right)^{1/k} \leq BC\sqrt{k}$.

To be precise on what we are going to study in this paper, for any vector $\mathbf{x} = (x_1, \cdots, x_n)^T$ in $\mathbb{R}^n$, let

$$\|\mathbf{x}\|_q^q = \sum_{i=1}^n |x_i|^q$$

for $q \in (0, \infty)$. It is known that for $q \geq 1$, $\|\cdot\|_q$ is a norm for $\mathbb{R}^n$ and $\|\cdot\|_q^q$ is a quasi-norm for $\mathbb{R}^n$ for $q \in (0,1)$ that satisfies all the properties for a norm except the triangle inequality. Let $A = (a_{ij})_{1\leq i\leq m, 1\leq j\leq n}$ be a matrix. The standard largest q-singular value is defined by

$$s_1^{(q)}(A) := \sup\left\{\frac{\|Ax\|_q}{\|x\|_q} : x \in \mathbb{R}^n \text{ with } x \neq 0\right\}. \tag{1.1}$$

It is known that for $q \geq 1$, the equation in (1.1) defines a norm on the space of $m \times n$ matrices. In addition, we know

$$\max_j \|a_j\|_q \leq s_1^{(q)}(A) \leq n^{\frac{q-1}{q}} \max_j \|a_j\|_q, \tag{1.2}$$

where a_j, $j = 1, 2, \cdots, n$, are the column vectors of A. We refer to any book on matrix theory for the properties of the largest singular value $s_1^q(A)$ when $q \geq 1$, for example, [11]. However, for $q \in (0,1)$, the properties of $s_1^q(A)$ are not well-known. For convenience, we shall explain some useful properties in the Preliminaries section.

The purpose of this paper is to study the matrix spectrum, e.g. $s_1^q(A)$ for random matrix A with pre-Gaussian entries. Two sets of our main results are the following

Theorem 1.3 (Upper tail probability of the largest q-singular value)**.** *Let ξ be a pre-Gaussian variable normalized to have variance 1 and A be an $m \times m$ matrix with i.i.d. copies of ξ in its entries. Then for any $0 < q < 1$,*

$$\mathbb{P}\left(s_1^{(q)}(A) \geq Cm^{\frac{1}{q}}\right) \leq \exp\left(-C'm\right) \tag{1.3}$$

for some C, $C' > 0$ only dependent on the pre-Gaussian variable ξ.

Theorem 1.4 (Lower tail probability of the largest q-singular value)**.** *Let ξ be a pre-Gaussian variable normalized to have variance 1 and A be an $m \times m$ matrix with i.i.d. copies of ξ in its entries. Then there exists a constant $K > 0$ such that*

$$\mathbb{P}\left(s_1^{(q)}(A) \leq Km^{\frac{1}{q}}\right) \leq c^m \tag{1.4}$$

for some $0 < c < 1$, where K only depends on the pre-Gaussian variable ξ.

These results have their counterparts in papers by Yin, Bai, Krishnaiah'1988, [31], Bai, Silverstein and Yin'1988, [1] and Sosnikov'2002, [20] for the ℓ_2-norm. It

is interesting to know if the above results hold for general random matrices whose entries are i.i.d. copies of a random variable of the bounded fourth moment.

Next we would like to study the smallest singular values. In general we can define the k-th q-singular value as follows.

Definition 1.1. The k-th q-singular value of an $m \times n$ matrix A is defined by

$$s_k^{(q)}(A) := \inf\left\{\sup\left\{\frac{\|Ax\|_q}{\|x\|_q} : x \in V \setminus \{0\}\right\} : V \subseteq \mathbb{R}^n,\ \dim(V) \geq n-k+1\right\}. \tag{1.5}$$

It is easy to see that

$$s_1^{(q)}(A) \geq s_2^{(q)}(A) \geq \ldots \geq s_{\min(m,n)}^{(q)}(A) \geq 0. \tag{1.6}$$

The smallest singular value $s^q_{\min(m,n)}$ is also of special interest in various studies. In the lower tail probability estimate, we divide the study in two cases when $m > n$ (tall matrices) and $m = n$ (square matrices) under the assumption that A is of full rank. The study is heavily dependent on the known results on the compressible and incompressible vectors. In the upper tail probability estimate, we use the known estimates on the projection in the ℓ_2-norm. Another set of main results is as follows. For tall random matrices, we have

Theorem 1.5 (Lower tail probability on the smallest q-singular value)**.** *Let us fix $0 < q \leq 1$. Let ξ be the pre-Gaussian random variable with mean 0 and variance 1. Suppose that A is an $m \times n$ matrix with i.i.d. copies of ξ in its entries with $m > n$. Then there exist some $\varepsilon > 0, c > 0$ and $\lambda \in (0,1)$ dependent on q and ε such that*

$$\mathbb{P}\left(s_m^{(q)}(A) \leq \varepsilon m^{1/q}\right) < e^{-cm} \tag{1.7}$$

when $n \leq \lambda m$.

For square random matrices, we have

Theorem 1.6 (Lower tail probability on the smallest q-singular value)**.** *Let us fix $0 < q \leq 1$. Let ξ be the pre-Gaussian random variable with variance 1 and A be an $n \times n$ matrix with i.i.d. copies of ξ in its entries. Then for any $\varepsilon > 0$, one has*

$$\mathbb{P}\left(s_n^{(q)}(A) \leq \gamma n^{-1/q}\right) < \varepsilon, \tag{1.8}$$

where $\gamma > 0$ depends only on the pre-Gaussian variable ξ.

The above theorem is an extension of Theorem 1.1. Finally we have

Theorem 1.7 (Upper tail probability on the smallest q-singular value)**.** *Given any $0 < q \leq 1$, let ξ be a pre-Gaussian random variable with variance 1 and A be an $n \times n$ matrix with i.i.d. copies of ξ in its entries. Then for any $K > e$, there exist some $C > 0$, $0 < c < 1$, and $\alpha > 0$ which are only dependent on the pre-Gaussian variable ξ such that*

$$\mathbb{P}\left(s_n^{(q)}(A) > Kn^{-1/2}\right) \leq \frac{C(\ln K)^\alpha}{K^\alpha} + c^n. \tag{1.9}$$

In particular, for any $\varepsilon > 0$, there exist some $K > 0$ and n_0 such that

$$\mathbb{P}\left(s_n^{(q)}(A) > Kn^{-1/2}\right) < \varepsilon \tag{1.10}$$

for all $n \geq n_0$.

The above theorem is an extension of Theorem 1.2. Note that we are not able to prove

$$\mathbb{P}\left(s_n^{(q)}(A) > Kn^{-1/q}\right) < \varepsilon \tag{1.11}$$

under the assumptions in Theorem 1.7. However, we strongly believe that the above inequality holds. We leave it as a conjecture.

The remainder of the paper is devoted to the proof of these five theorems which give a good understanding of the spectrum of pre-Gaussian random matrices in ℓ_q-quasinorm with $0 < q \leq 1$. We shall present the analysis in four separate sections after the Preliminaries section.

2. Preliminaries

First of all, one can easily derive the following

Lemma 2.1. *For $0 < q < 1$, the equation in* (1.1)*) defines a quasinorm on the space of $m \times N$ matrices. In particular, we have*

$$\left(s_1^{(q)}(A+B)\right)^q \leq \left(s_1^{(q)}(A)\right)^q + \left(s_1^{(q)}(B)\right)^q$$

for any $m \times N$ matrices A and B. Moreover,

$$s_1^{(q)}(A) = \max_j \|a_j\|_q \tag{2.1}$$

for $0 < q \leq 1$, where a_j, $j = 1, \ldots, N$, are the columns of matrix A.

Proof. It is straightforward and not hard to show that $s_1^{(q)}(A)$, $q \leq 1$, defines a quasinorm on matrices by using the quasi-norm properties of $\|\mathbf{x}\|_q$, the ℓ_q-quasinorm on vectors.

To prove equation (2.1), on one hand, we have

$$\|Ax\|_q^q \leq \sum_{j=1}^{N} |x_j|^q \cdot \|a_j\|_q^q \leq \|x\|_q^q \max_j \|a_j\|_q^q \tag{2.2}$$

for $0 < q \leq 1$, which implies

$$s_1^{(q)}(A) \leq \max_j ||a_j||_q. \tag{2.3}$$

On the other hand, by (1.1), we have

$$s_1^{(q)}(A) = \sup_{x \in \mathbb{R}^N, \|x\|_q = 1} \|Ax\|_q \geq \|Ae_j\|_q = ||a_j||_q \tag{2.4}$$

for every j, where e_j is the j-th standard basis vector of $\mathbb{R}^N$, and then it follows that

$$s_1^{(q)}(A) \geq \max_j ||a_j||_q. \tag{2.5}$$

Thus, combined with (2.3), we obtain the equation (2.1) for $0 < q \leq 1$ as desired. □

Next we need the following elementary estimate. Mainly we need a linear bound for partial binomial expansion.

Lemma 2.2 (Linear bound for partial binomial expansion)**.** *For every positive integer n,*

$$\sum_{k=\left\lfloor \frac{n}{2}\right\rfloor+1}^{n}\left(\begin{array}{c} n \\ k \end{array}\right)x^k\left(1-x\right)^{n-k}\leq 8x$$

for all $x\in[0,1]$.

Proof. Let us start with an even integer. For every $x\in\left[\frac{1}{8},1\right]$, we have

$$\sum_{k=n+1}^{2n}\left(\begin{array}{c} 2n \\ k \end{array}\right)x^k\left(1-x\right)^{2n-k}\leq\sum_{k=0}^{2n}\left(\begin{array}{c} 2n \\ k \end{array}\right)x^k\left(1-x\right)^{2n-k}=1\leq 8x. \tag{2.6}$$

But for $x\in\left[0,\frac{1}{8}\right]$, we let

$$f\left(x\right):=\sum_{k=n+1}^{2n}\left(\begin{array}{c} 2n \\ k \end{array}\right)x^k\left(1-x\right)^{2n-k}.$$

By De Moivre-Stirling's formula (see e.g. [7]) and furthermore the estimate in [13],

$$n!=\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\lambda_n},$$

where $\frac{1}{12n+1}<\lambda_n<\frac{1}{12n}$. We have

$$\left(\begin{array}{c} 2n \\ n \end{array}\right)=\frac{\sqrt{2\pi 2n}\left(\frac{2n}{e}\right)^{2n}e^{\lambda_{2n}}}{\left(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\lambda_n}\right)^2}=\frac{4^n}{\sqrt{\pi n}}e^{\lambda_{2n}-2\lambda_n}\leq\frac{4^n}{\sqrt{\pi n}}. \tag{2.7}$$

Since $\left(\begin{array}{c} 2n \\ k \end{array}\right)\leq\left(\begin{array}{c} 2n \\ n \end{array}\right)$ for $n+1\leq k\leq 2n$,

(2.8)

$$f\left(x\right)\leq\sum_{k=n+1}^{2n}\left(\begin{array}{c} 2n \\ n \end{array}\right)x^k\left(1-x\right)^{2n-k}\leq\sum_{k=n+1}^{2n}\left(\begin{array}{c} 2n \\ n \end{array}\right)x^k\leq n\left(\begin{array}{c} 2n \\ n \end{array}\right)x^{n+1}$$

for all $x\in[0,1]$. Using (2.7), we have

$$f\left(x\right)\leq 4^n\sqrt{\frac{n}{\pi}}x^{n+1}. \tag{2.9}$$

Letting $g(x)=4^n\sqrt{\frac{n}{\pi}}x^n$, we have

$$\ln(g(x))=n\ln(4x)+\frac{1}{2}\ln n-\frac{1}{2}\ln\pi\leq -n\ln 2+\frac{1}{2}\ln n-\frac{1}{2}\ln\pi\leq 0$$

for $x\in[0,1/8]$. Thus we have $f(x)\leq x\leq 8x$. Also, we can have a similar estimate for odd integers. These complete the proof. □

Remark 2.1. The coefficient on the right-hand side can be improved by Markov's inequality, but the estimate obtained by the analytic technique above is actually good enough for the purposes of this paper.

Next we review the smallest q-singular values. Without loss of generality, we consider $m\geq n$. Then the n-th q-singular value is the smallest q-singular value which can also be expressed in another way.

Lemma 2.3. *Let A be an $m \times n$ matrix with $m \geq n$. Then the smallest q-singular value*

$$s_n^{(q)}(A) = \inf \left\{ \frac{\|Ax\|_q}{\|x\|_q} : x \in \mathbb{R}^n \text{ with } x \neq 0 \right\}. \tag{2.10}$$

Proof. By the definition,

$$\begin{aligned} s_n^{(q)}(A) &= \inf \left\{ \sup \left\{ \frac{\|Ax\|_q}{\|x\|_q} : x \in V \setminus \{0\} \right\} : V \subseteq \mathbb{R}^n, \dim(V) \geq 1 \right\} \\ &\leq \inf \left\{ \sup \left\{ \frac{\|Av\|_q}{\|v\|_q} : v \in V \setminus \{0\} \right\} : V = \operatorname{span}(x) : x \in \mathbb{R}^n \setminus \{0\} \right\} \\ &= \inf \left\{ \frac{\|Ax\|_q}{\|x\|_q} : x \in \mathbb{R}^n \text{ with } x \neq 0 \right\}. \end{aligned} \tag{2.11}$$

We also know the infimum can be achieved by considering the unit $\mathcal{S}_q$-sphere in the finite-dimensional space, and so the claim follows. □

In particular, if A is an $n \times n$ matrix, we know

$$\begin{aligned} s_n^{(q)}(A) &= \inf \left\{ \frac{\|Ax\|_q}{\|x\|_q} : x \in \mathbb{R}^n \text{ with } x \neq 0 \right\} \\ &= \frac{1}{\sup \left\{ \frac{\|A^{-1}x\|_q}{\|x\|_q} : x \in \mathbb{R}^n \text{ with } x \neq 0 \right\}} \\ &= \frac{1}{s_1^{(q)}(A^{-1})}. \end{aligned} \tag{2.12}$$

The estimate of the largest q-singular value can be used to estimate the smallest q-singular values based on this relation.

As we see, the q-singular value is defined by the ℓ_q-quasinorm, as opposed to the ℓ_2-norm, but using a similar proof for the relationship between the rank of a matrix and its smallest singular value in ℓ_2, one has the following relationship between the rank of a matrix and its smallest q-singular value.

Lemma 2.4. *For any positive integer m and n, an $m \times n$ matrix A is of full rank if and only if $s_{\min(m,n)}^{(q)}(A) > 0$.*

Remark 1. One could also derive this lemma by the properties of singular values defined by the ℓ_2-norm and by using the inequalities on the relations between the ℓ_2-norm and the ℓ_q-quasinorm.

We shall need the following result to estimate the smallest q-singular values.

Lemma 2.5. *Let A be a matrix of size $m \times N$. Suppose that $m \geq N$. Then*

$$s_{\min(m,N)}^{(q)}(A) \leq \min_j \|a_j\|_q.$$

Proof. Choose e_{j_0} to be a standard basis vector of $\mathbb{R}^N$ such that $\|Ae_{j_0}\|_q = \min_j \|a_j\|_q$ and use the definition of $s_{\min(m,N)}^{(q)}(A)$ for $m \geq N$. □

The following generalization of Lemma 4.10 in Pisier'1999, [12] will be used in a later section.

Lemma 2.6. *For $0 < q \le 1$, let $\mathcal{S}_q := \{x \in \mathbb{R}^n : |x|_q = 1\}$ denote the unit sphere of $\mathbb{R}^n$ in the ℓ_q-quasinorm. For any $\delta > 0$, there exists a finite set $\mathcal{U}_q \subseteq \mathcal{S}_q$ with*

$$\min_{\mathbf{u}\in\mathcal{U}_q} \|x-u\|_q^q \le \delta \quad \textit{for all } x \in \mathcal{S}_q \qquad \textit{and} \qquad \operatorname{card}(\mathcal{U}_q) \le \left(1+\frac{2}{\delta}\right)^{n/q}.$$

Proof. Let $(u_1, \dots, u_k)$ be a set of k points on the sphere $\mathcal{S}_q$ such that $|u_i -_j |_q^q > \delta$ for all $i \ne j$. We choose k as large as possible. Thus, it is clear that

$$\min_{1\le i\le k} |x-u_i|_q^q \le \delta \quad \text{for all } x \in \mathcal{S}_q.$$

Let $\mathcal{B}_q := \{x \in \mathbb{R}^n : |x|_q \le 1\}$ be the unit ball of $\mathbb{R}^n$ relative to the quasinorm $|\cdot|_q$. It is easy to see that the $(\delta/2)$-balls centered at $\mathbf{u}_i$,

$$u_i + \left(\frac{\delta}{2}\right)^{1/q} \mathcal{B}_q, \quad 1 \le i \le k,$$

are disjoint. Indeed, if x would belong to the $(\delta/2)$-ball centered at x_i as well as the $(\delta/2)$-ball centered at x_j, we would have

$$|u_i - u_j|_q^q \le |u_i - x|_q^q + |u_j - x|_q^q \le \frac{\delta}{2} + \frac{\delta}{2} = \delta,$$

which is a contradiction. Besides, it is easy to see that

$$u_i + \left(\frac{\delta}{2}\right)^{1/q} \mathcal{B}_q \subseteq \left(1+\frac{\delta}{2}\right)^{1/q} \mathcal{B}_q, \quad 1 \le i \le k.$$

By comparison of volumes, we get

$$k\mathrm{Vol}\left(\left(\frac{\delta}{2}\right)^{1/q}\mathcal{B}_q\right) = \sum_{i=1}^{k} \mathrm{Vol}\left(\mathbf{u}_i + \left(\frac{\delta}{2}\right)^{1/q}\mathcal{B}_q\right) \le \mathrm{Vol}\left(\left(1+\frac{\delta}{2}\right)^{1/q}\mathcal{B}_q\right).$$

Then, by homogeneity of the volumes, we have

$$k\left(\frac{\delta}{2}\right)^{n/q} \mathrm{Vol}\left(\mathcal{B}_q\right) \le \left(1+\frac{\delta}{2}\right)^{n/q} \mathrm{Vol}\left(\mathcal{B}_q\right),$$

which implies that $k \le \left(1+\frac{2}{\delta}\right)^{n/q}$. This completes the proof. □

3. The upper tail probability of the largest q-singular value

We begin with the following

Theorem 3.1 (Upper tail probability of the largest 1-singular value)**.** *Let ξ be a pre-Gaussian variable normalized to have variance* 1 *and A be an $m \times m$ matrix with i.i.d. copies of ξ in its entries. Then*

$$\mathbb{P}\left(s_1^{(1)}(A) \ge Cm\right) \le \exp\left(-C'm\right) \tag{3.1}$$

for some C, $C' > 0$ only dependent on the pre-Gaussian variable ξ.

Proof. Since a_{ij} are i.i.d. copies of the pre-Gaussian variable ξ, $\mathbb{E}a_{ij} = 0$, and there exist some $\lambda > 0$ such that $\mathbb{E}\,|a_{ij}|^k \le k!\lambda^k$ for all k. Without loss of generality, we may assume that $\lambda \ge 1$. With the variance $\mathbb{E}a_{ij}^2 = 1$, we have

$$\mathbb{E}\,|a_{ij}|^k \le \frac{\mathbb{E}a_{ij}^2}{2} H^{k-2} k!$$

for $H := 2\lambda^3$ and all $k \geq 2$. By the Bernstein inequality (cf. Theorem 5.2 in [3]), we know that

$$\mathbb{P}\left(\left|\sum_{i=1}^{m} a_{ij}\right| \geq t\right) \leq 2\exp\left(-\frac{t^2}{2\left(m+tH\right)}\right) = 2\exp\left(-\frac{t^2}{2\left(m+2t\lambda^3\right)}\right)$$

for all $t > 0$ and for each $j = 1, \cdots, N$. In particular, when $t = Cm$,

$$\mathbb{P}\left(\left|\sum_{j=1}^{m} a_{ij}\right| \geq Cm\right) \leq 2\exp\left(-\frac{C^2 m}{4C\lambda^3+2}\right). \tag{3.2}$$

Here a condition on C will be determined later.

On the other hand, by Lemma 2.1,

$$s_1^{(1)}(A) = \max_j ||a_j||_1 = \sum_{i=1}^{m} |a_{ij_0}|$$

for some j_0. Furthermore, for any $t > 0$, by the probability of the union,

$$\mathbb{P}\left(\sum_{i=1}^{m} |a_{ij}| \geq t\right) \leq \sum_{(\epsilon_1,\ldots,\epsilon_m)\in\{-1,1\}^m} \mathbb{P}\left(\sum_{i=1}^{m} \epsilon_i a_{ij} \geq t\right). \tag{3.3}$$

But $-a_{ij}$ has the same pre-Gaussian properties as a_{ij_0}, precisely, $\mathbb{E}\left(-a_{ij}\right) = 0$ and $\mathbb{E}\left|-a_{ij}\right|^k \leq k!\lambda^k$. Thus we have

$$\begin{aligned}\mathbb{P}\left(s_1^{(1)}(A) \geq Cm\right) &\leq m \quad \mathbb{P}\left(\sum_{i=1}^{m} |a_{ij}| \geq Cm\right) \\ &\leq \quad 2^m m\mathbb{P}\left(\left|\sum_{i=1}^{m} a_{ij}\right| \geq Cm\right) \\ &\leq \quad 2^m m\exp\left(-\tfrac{C^2 m}{4C\lambda^3+2}\right) \\ &\leq \quad \exp\left(-\left(\tfrac{C^2}{4C\lambda^3+2} - \ln 2 - 1\right)m\right).\end{aligned} \tag{3.4}$$

To obtain an exponential decay for the probability $\mathbb{P}\left(s_1^{(1)}(A) \geq Cm\right)$, we require that $\frac{C^2}{4C\lambda^3+2} - \ln 2 - 1 > 0$, for which

$$C > 2\lambda^3 + 2\lambda^3 \ln 2 + \sqrt{2 + 2\ln 2 + 4\lambda^6 + 8\lambda^6 \ln 2 + 4\lambda^6 \ln^2 2}. \tag{3.5}$$

That is, choosing $C' = \frac{C^2}{4C\lambda^3+2} - \ln 2 - 1$, we get (3.1). □

The previous theorem allows us to estimate the largest q-singular value for $0 < q < 1$. The estimate can follow easily from Theorem 3.1, but it is one of the tail probabilistic estimates we wanted to obtain, so let us state it as a theorem, which is Theorem 1.3.

Proof of Theorem 1.3. By Hölder's inequality, we have $\|a_j\|_q \leq m^{\frac{1}{q}-1}\|a_j\|_1$ for $0 < q < 1$. It follows from Lemma 2.1 that

$$s_1^{(q)}(A) = \max_j \|a_j\|_q \leq m^{\frac{1}{q}-1} s_1^{(1)}(A). \tag{3.6}$$

From (3.1), we have

$$\begin{aligned}\mathbb{P}\left(s_1^{(q)}(A) \geq C m^{\frac{1}{q}}\right) &\leq \mathbb{P}\left(m^{\frac{1}{q}-1} s_1^{(1)}(A) \geq C m^{\frac{1}{q}}\right) \\ &= \mathbb{P}\left(s_1^{(1)}(A) \geq C m\right) \\ &\leq \exp\left(-C' m\right)\end{aligned} \tag{3.7}$$

for some C, $C' > 0$. □

4. The lower tail probability of the largest q-singular value

Let us use the result in Lemma 2.2 to give estimates on the lower tail probabilities of the largest q-singular value.

Lemma 4.1. *Suppose ξ_1, ξ_2, $\cdots$, ξ_n are i.i.d. copies of a random variable ξ. Then for any $\varepsilon > 0$,*

$$\mathbb{P}\left(\sum_{i=1}^{n} |\xi_i| \leq \frac{n\varepsilon}{2}\right) \leq 8\mathbb{P}\left(|\xi| \leq \varepsilon\right). \tag{4.1}$$

Proof. First, we have the relation on the probability events that

$$\left\{(\xi_1, \ldots, \xi_n) : \sum_{i=1}^{n} |\xi_i| \leq \frac{n\varepsilon}{2}\right\} \tag{4.2}$$

is contained in

$$\bigcup_{k=\lfloor \frac{n}{2} \rfloor+1}^{n} \quad \bigcup_{\substack{\{i_1,\ldots,i_k\} \\ \subset \{1,\ldots,n\}}} \quad \left\{(\xi_1,\ldots,\xi_n) : |\xi_{i_1}| \leq \varepsilon, \ldots, |\xi_{i_k}| \leq \varepsilon, \left|\xi_{i_{k+1}}\right| > \varepsilon, \ldots, |\xi_{i_n}| > \varepsilon\right\}, \tag{4.3}$$

where $\{i_1, i_2, \ldots, i_k\}$ is a subset of $\{1, 2, \ldots, n\}$ and $\{i_{k+1}, \ldots, i_n\}$ is its complement, and let us denote the set (4.3) by $\mathcal{E}$.

Let $x = \mathbb{P}\left(|\xi_1| \leq \varepsilon\right)$. Then by the union probability,

$$\mathbb{P}\left(\mathcal{E}\right) = \sum_{k=\lfloor \frac{n}{2} \rfloor+1}^{n} \begin{pmatrix} n \\ k \end{pmatrix} x^k \left(1-x\right)^{n-k}, \tag{4.4}$$

and applying Lemma 2.2, we have

$$\mathbb{P}\left(\mathcal{E}\right) \leq 8x = 8\mathbb{P}\left(|\xi_1| \leq \varepsilon\right). \tag{4.5}$$

Since the event (4.2) is contained in the event (4.3), we have

$$\mathbb{P}\left(\sum_{i=1}^{n} |\xi_i| \leq \frac{n\varepsilon}{2}\right) \leq \mathbb{P}\left(\mathcal{E}\right) \leq 8\mathbb{P}\left(|\xi_1| \leq \varepsilon\right). \tag{4.6}$$

□

We start with a lower tail probability for the 1-singular values.

Theorem 4.1 (Lower tail probability of the largest 1-singular value)**.** *Let ξ be a pre-Gaussian variable normalized to have variance 1 and A be an $m \times m$ matrix with i.i.d. copies of ξ in its entries. Then there exists a constant $K > 0$ such that*

$$\mathbb{P}\left(s_1^{(1)}(A) \leq K m\right) \leq c^m \tag{4.7}$$

for some $0 < c < 1$*, where* K *only depends on the pre-Gaussian variable* ξ*.*

Proof. Since a_{ij} has variance 1, there exists $\delta > 0$ and $0 \le \beta < 1$ such that

$$\mathbb{P}\left(|a_{ij}| \le \delta\right) = \beta. \tag{4.8}$$

Let B_j be the number of variables in $\{a_{ij}\}_{i=1}^m$ that are less than or equal to δ. Then if $\sum_{i=1}^m |a_{ij}| \le \delta \cdot \lambda m$ for $0 < \lambda < 1$, then $B_j \ge (1-\lambda)\, m$, because otherwise $\sum_{i=1}^m |a_{ij}| > \delta \cdot \lambda m$. It follows that

$$\mathbb{P}\left(\sum_{i=1}^m |a_{ij}| \le \delta \cdot \lambda m\right) \le \mathbb{P}\left(B_j \ge (1-\lambda)\, m\right). \tag{4.9}$$

By Markov's inequality,

$$\mathbb{P}\left(B_j \ge (1-\lambda)\, m\right) \le \frac{\mathbb{E}B_i}{(1-\lambda)\, m}, \tag{4.10}$$

but B_j satisfies a binomial distribution of m independent experiments, each of which yields success with probability β; therefore

$$\mathbb{P}\left(B_j \ge (1-\lambda)\, m\right) \le \frac{\beta}{1-\lambda}. \tag{4.11}$$

By choosing suitable λ, we can make $0 < \frac{\beta}{1-\lambda} < 1$. Thus

$$\mathbb{P}\left(\sum_{i=1}^m |a_{ij}| \le \delta \cdot \lambda m\right) \le c \tag{4.12}$$

for some $0 < c < 1$. It follows that

$$\begin{aligned} \mathbb{P}\left(s_1^{(1)}(A) \le \lambda\delta m\right) &= \mathbb{P}\left(\max_{1\le j\le N}\left(\textstyle\sum_{i=1}^m |a_{ij}|\right) \le \lambda\delta m\right) \\ &= \textstyle\prod_{j=1}^m \mathbb{P}\left(\left(\sum_{i=1}^m |a_{ij}|\right) \le \lambda\delta m\right) \\ &\le c^m. \end{aligned} \tag{4.13}$$

Thus letting $K = \lambda\delta$, we obtain (3.1).

□

For general $0 < q < 1$, we have Theorem 1.4.

Proof of Theorem 1.4*.* We can use the same method as in the proof of Theorem 4.1. Since a_{ij} has nonzero variance, there exists $\delta > 0$ and $0 \le \beta < 1$ such that

$$\mathbb{P}\left(|a_{ij}|^q \le \delta\right) = \beta. \tag{4.14}$$

Then by Lemma 4.1 and substituting a_{ij} in the proof of Theorem 4.1 by $|a_{ij}|^q$,

$$\begin{aligned} \mathbb{P}\left(s_1^{(q)}(A) \le (\lambda\delta)^{\frac{1}{q}}\, m^{\frac{1}{q}}\right) &= \mathbb{P}\left(\max_{1\le j\le N}\left(\textstyle\sum_{i=1}^m |a_{ij}|^q\right) \le \lambda\delta m\right) \\ &= \textstyle\prod_{j=1}^m \mathbb{P}\left(\left(\sum_{i=1}^m |a_{ij}|^q\right) \le \lambda\delta m\right) \\ &\le c^m \end{aligned} \tag{4.15}$$

for some $0 < c < 1$. Thus letting $K = (\lambda\delta)^{\frac{1}{q}}$, (1.4) follows. □

Remark 2. If one uses the quasinorm comparison inequality $s_1^{(q)}(A) \le s_1^{(1)}(A)$ for $0 < q \le 1$, one can get

$$\mathbb{P}\left(s_1^{(q)}(A) \le Km\right) \le c^m \tag{4.16}$$

for $0 < q \le 1$, but with a loss of the estimate on $\mathbb{P}\left(s_1^{(q)}(A) \le Km^{\frac{1}{q}}\right)$.

Since the bounded moment growth condition for pre-Gaussian variables is not needed in the proof of Theorem 4.1, the above proofs also show that the theorem holds for any random variable with nonzero variance. Therefore, more generally, we have

Theorem 4.2. *Let ξ be a random variable with non-zero variance and A be an $m \times m$ matrix with i.i.d. copies of ξ in its entries. Then there exists a constant $K > 0$ such that*

$$\mathbb{P}\left(s_1^{(q)}(A) \leq K m^{\frac{1}{q}}\right) \leq c^m \tag{4.17}$$

for some $0 < c < 1$, where K only depends on ε and the random variable ξ.

5. The lower tail probability of the smallest q-singular value

In this section, we first study the probability estimates of the smallest q-singular value of rectangular random matrices with $m > n$. Then we give some estimates for square random matrices.

5.1. **The tall random matrix case.** In this subsection, we assume that $n \leq \lambda m$ with $\lambda \in (0, 1)$ and consider the smallest q-singular value of random matrices of size $m \times n$.

Theorem 5.1. *Given any $0 < q \leq 1$, let ξ be the pre-Gaussian random variable with variance 1 and A be an $m \times n$ matrix with i.i.d. copies of ξ in its entries. Then there exist some $\gamma > 0, b > 0$ and $\nu \in (0, 1)$ dependent on the pre-Gaussian random variable ξ such that*

$$\mathbb{P}\left(s_n^{(q)}(A) < \gamma m^{1/q}\right) < e^{-bm} \tag{5.1}$$

with $n \leq \nu m$.

To prove this result, we need to establish a few lemmas.

Lemma 5.1. *Fix any $0 < q \leq 1$. For any $\xi_1, \cdots, \xi_m$ that are i.i.d. copies of a pre-Gaussian variable with non-zero variance, for any $c \in (0, 1)$ there exists $\lambda \in (0, 1)$, that does not depend on m, such that*

$$\mathbb{P}\left(\sum_{k=1}^{m} |\xi_k|^q < \lambda m\right) \leq c^m. \tag{5.2}$$

Proof. For any $\xi_1, \cdots, \xi_m$ that are i.i.d. copies of a pre-Gaussian variable with non-zero variance, we know that there exists some $\delta > 0$ such that

$$\varepsilon_0 := \mathbb{P}\left(|\xi_k| \leq \delta\right) < 1 \tag{5.3}$$

for $k = 1, 2, \cdots, m$, because otherwise the pre-Gaussian variable would have a zero variance. Then using the Riemann–Stieltjes integral for expectation, we have

$$\begin{aligned} \mathbb{E}\exp\left(-\frac{|\xi_k|^q}{\lambda}\right) &= \int_0^\infty \exp\left(-\frac{t^q}{\lambda}\right) d\mathbb{P}\left(|\xi_k| \le t\right) \\ &\le \int_0^\delta d\mathbb{P}\left(|\xi_k| \le t\right) + \int_\delta^\infty \exp\left(-\frac{t^q}{\lambda}\right) d\mathbb{P}\left(|\xi_k| \le t\right) \\ &= \varepsilon_0 + \int_\delta^\infty \exp\left(-\frac{t^q}{\lambda}\right) d\mathbb{P}\left(|\xi_k| \le t\right). \end{aligned}$$

Choose $\lambda > 0$ to be small enough such that

$$\exp\left(-\frac{t^q}{\lambda}\right) \le \exp\left(-\frac{\delta^q}{\lambda}\right) < \frac{\varepsilon_0}{2\left(1-\varepsilon_0\right)}$$

for $t \ge \delta$. Therefore, it follows that

$$\mathbb{E}\exp\left(-\frac{|\xi_k|^q}{\lambda}\right) \le \varepsilon_0 + \frac{\varepsilon_0}{2\left(1-\varepsilon_0\right)} \int_\delta^\infty d\mathbb{P}\left(|\xi_k| \le t\right) \le \varepsilon_0 + \frac{\varepsilon_0}{2} = \frac{3}{2}\varepsilon_0.$$

Finally, applying Markov's inequality, we obtain

$$\begin{aligned} \mathbb{P}\left(\sum_{k=1}^m |\xi_k|^q < \lambda m\right) &= \mathbb{P}\left(\exp\left(m - \frac{1}{\lambda}\sum_{k=1}^m |\xi_k|^q\right) > 1\right) \\ &\le \mathbb{E}\left(\exp\left(m - \frac{1}{\lambda}\sum_{k=1}^m |\xi_k|^q\right)\right) \\ &= e^m \prod_{k=1}^m \mathbb{E}\exp\left(-\frac{|\xi_k|^q}{\lambda}\right). \\ &\le \left(3e\epsilon_0/2\right)^m. \end{aligned}$$

For any $c \in (0, 1)$, we choose ϵ_0 such that $3e\epsilon_0/2 = c$. This completes the proof. $\square$

The following lemma is a property of the linear combination of pre-Gaussian variables, which allows us to obtain the probabilistic estimate on $\|Av\|_q$ for the pre-Gaussian ensemble A.

Lemma 5.2 (Linear combination of pre-Gaussian variables). *Let a_{ij}, $i = 1, 2, \cdots, m$ and $j = 1, 2, \ldots, n$ be pre-Gaussian variables and $\eta_i = \sum_{j=1}^n a_{ij} x_j$. Then η_i are pre-Gaussian variables for $i = 1, 2, \ldots, m$.*

Proof. Since a_{ij} are pre-Gaussian variables, $\mathbb{E}a_{ij} = 0$, and there are constants $\lambda_{ij} > 0$ such that $\mathbb{E}\left|a_{ij}\right|^k \le k!\lambda_{ij}^k$ for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, N$. It is easy to see

$$\mathbb{E}\eta_i = \sum_{j=1}^N x_j \mathbb{E}a_{ij} = 0.$$

Letting $\|x\|_1 = \sum_{i=1}^N |x_j|$, we use the convexity to have

$$\begin{aligned}\mathbb{E}\left(|\eta_i|^k\right) &\leq \mathbb{E}\left(\|x\|_1 \sum_{j=1}^N |a_{ij}| \frac{|x_j|}{\|x\|_1}\right)^k \\ &\leq \|x\|_1^k \sum_{j=1}^N \frac{|x_j|}{\|x\|_1} \mathbb{E}\left(|a_{ij}|\right)^k \leq k! \|x\|_1^k (\max_j\{\lambda_{ij}\})^k\end{aligned}$$

for all integers $k \geq 1$. Thus, η_k is a pre-Gaussian random variable. □

Combining two lemmas above, we obtain the following

Lemma 5.3. *Given any $0 < q \leq 1$ and letting A be an $m \times n$ pre-Gaussian matrix, for any $c \in (0,1)$ there exists $\lambda \in (0,1)$ such that*

$$\mathbb{P}\left(\|Av\|_q < \lambda^{1/q} m^{1/q}\right) \leq c^m \tag{5.4}$$

for each $v \in \mathbb{S}_q$, where $\mathbb{S}_q$ is the $(n-1)$-dimensional unit sphere in the ℓ_q-quasinorm.

We are now ready to prove Theorem 5.1.

Proof of Theorem 5.1. By using Lemma 2.6, for any $\delta > 0$ there exists a δ-net $\mathcal{U}_q$ in unit sphere $\mathbb{S}_q$ such that

$$\min_{u \in \mathcal{U}_q} \|x-u\|_q^q \leq \delta \quad \text{for all } x \in \mathcal{S}_q \qquad \text{and} \qquad \operatorname{card}(\mathcal{U}_q) \leq \left(1+\frac{2}{\delta}\right)^{n/q}.$$

By Lemma 5.3, for all $v \in \mathcal{U}_q$ we have

$$\mathbb{P}\left(\|Av\|_q^q < \lambda m, \text{ for all } v \in \mathcal{U}_q\right) \leq \left(1+\frac{2}{\delta}\right)^{n/q} c^m. \tag{5.5}$$

Since the event $s_n^{(q)}(A) < \gamma m^{\frac{1}{q}}$ implies $\|Av'\|_q < 2\gamma m^{\frac{1}{q}}$ for some $v' \in \mathbb{S}_q$,

$$\mathbb{P}(s_n^{(q)}(A) < \gamma m^{1/q}) \leq \mathbb{P}\left(\|Av\|_q < 2\gamma m^{1/q} \text{ for some } v \in \mathbb{S}_q\right).$$

If $v \in \mathcal{U}_q$, we use (5.5) with $2\gamma < \lambda^{1/q}$ to have

$$\mathbb{P}(s_n^{(q)}(A) < \gamma m^{1/q}) \leq \left(1+\frac{2}{\delta}\right)^{n/q} c^m. \tag{5.6}$$

If $v \notin \mathcal{U}_q$, we use Theorem 1.3 to have

$$\begin{aligned}&\mathbb{P}\left(\|Av\|_q < 2\gamma m^{1/q} \text{ with } v \in \mathbb{S}_q \backslash \mathcal{U}_q\right) \\ &\leq e^{-c_1 m} + \mathbb{P}\left(s_1^{(q)}(A) \leq K m^{1/q} \text{ and } \|Av\|_q < 2\gamma m^{1/q} \text{ with } v \in \mathbb{S}_q \backslash \mathcal{U}_q\right).\end{aligned}$$

When $v \in \mathbb{S}_q \backslash \mathcal{U}_q$ in the event that $s_1^{(q)}(A) \leq K m^{1/q}$ and $\|Av\|_q < 2\gamma m^{1/q}$, there exists a $u \in \mathcal{U}_q$ within a q-distance δ such that

$$\begin{aligned}\|Au\|_q^q &\leq \|A(v-u)\|_q^q + \|Av\|_q^q \\ &\leq \left(s_1^{(q)}(A)\right)^q \|v-u\|_q^q + \|Av\|_q^q \\ &\leq K^q m\delta + (2\gamma)^q m \\ &< \lambda^q m\end{aligned}$$

if $\delta < \dfrac{\lambda^q - (2\gamma)^2}{K^q}$. We can use (5.5) again to conclude

(5.7)
$$\mathbb{P}\left(s_1^{(q)}(A) \le Km^{1/q} \text{ and } \|Av\|_q < 2\gamma m^{1/q} \text{ for some } v \in \mathbb{S}_q \backslash \mathcal{U}_q\right) \le \left(1 + \frac{2}{\delta}\right)^{n/q} c^m.$$

If we choose ν and c small enough in Lemma 5.1 with $n = \nu m$ such that

$$c_2 := \left(1 + \frac{2}{\delta}\right)^{\nu/q} c < 1,$$

we have thus completed the proof by choosing $b > 0$ such that $e^{-c_1 m} + e^{-c_2 m} \le e^{-bm}$. □

5.2. **The square random matrix case.** Now let us consider the square random matrices with pre-Gaussian entries.

Theorem 5.2. *Given any $0 < q \le 1$, let ξ be the pre-Gaussian random variable with variance 1 and A be an $n \times n$ matrix with i.i.d. copies of ξ in its entries. Then for any $\varepsilon > 0$ and $0 < q \le 1$, there exist some $K > 0$ and $c > 0$ dependent on ε and the pre-Gaussian random variable ξ such that*

$$\mathbb{P}\left(s_n^{(q)}(A) < \varepsilon n^{-\frac{1}{q}}\right) < C\varepsilon + C\alpha^n + \mathbb{P}\left(\|A\| > Kn^{-\frac{1}{2}}\right), \tag{5.8}$$

where $\alpha \in (0, 1)$ and $C > 0$ depend only on the pre-Gaussian variable and K.

To prove the above theorem, we generalize the ideas in Rudelson and Vershynin'2008, [15] to the setting of the ℓ_q-quasinorm. We first decompose $\mathbb{S}_q^{n-1}$ into the set of compressible vectors and the set of incompressible vectors. The concepts of compressible and incompressible vectors in $\mathbb{S}_2^{n-1}$ were introduced in [15]. See also Tao and Vu'2009, [27]. We shall use a generalized version of these concepts. Recall that $\|x\|_0$ denotes the number of nonzero entries of the vector $x \in \mathbb{R}^n$.

Definition 5.1 (Compressible and incompressible vectors in $\mathbb{S}_q^{n-1}$)**.** Fix $\rho, \lambda \in (0, 1)$. Let $Comp_q(\lambda, \rho)$ be the set of vectors $v \in \mathbb{S}_q^{n-1}$ such that there is a vector v' with $\|v'\|_0 \le \lambda n$ satisfying $\|v - v'\|_q \le \rho$. The set of incompressible vectors is defined as

$$Incomp_q(\lambda, \rho) := \mathbb{S}_q^{n-1} \setminus Comp_q(\lambda, \rho). \tag{5.9}$$

Now using the decomposition in Definition 5.1, we have

$$\begin{aligned} \mathbb{P}\left(s_n^{(q)}(A) < \varepsilon n^{-\frac{1}{q}}\right) \quad \le \quad & \mathbb{P}\left(\inf_{v \in Comp_q(\lambda,\rho)} \|Av\|_q < \varepsilon n^{-\frac{1}{q}}\right) \\ & + \mathbb{P}\left(\inf_{v \in Incomp_q(\lambda,\rho)} \|Av\|_q < \varepsilon n^{-\frac{1}{q}}\right). \end{aligned} \tag{5.10}$$

In the following we are going to consider each of the two terms on the right hand side of the above equation. For the first term on compressible vectors, we can apply Lemma 5.3 since

$$\mathbb{P}\left(\inf_{v \in Incomp_q(\lambda,\rho)} \|Av\|_q < \varepsilon n^{-\frac{1}{q}}\right) \le \mathbb{P}\left(\inf_{v \in Incomp_q(\lambda,\rho)} \|Av\|_q < \nu n^{\frac{1}{q}}\right), \tag{5.11}$$

to conclude that it actually decays exponentially for $n > 1$, where $\nu = \lambda^{1/q}$ as in Lemma 5.3.

However, for incompressible vectors, we first consider $dist_q(X_j, H_j)$, which denotes the distance between column X_j of an $n \times n$ random matrix A and the span of

other columns $H_j := span\left(X_1, \cdots, X_{j-1}, X_{j+1}, \ldots, X_n\right)$ in the ℓ_q-quasinorm. We generalize a result on the probability estimate of the distance in the ℓ_2-norm in [15] to the ℓ_q-quasinorm setting. This allows us to transform the probabilistic estimate on $\|Av\|_q$ for $v \in Incomp_q(\lambda, \rho)$ to the probabilistic estimate on the average of the distances $dist_q(X_j, H_j)$, $j = 1, 2, \ldots, n$.

Lemma 5.4. *Let A be an $n \times n$ random matrix with columns $X_1, \ldots, X_n$, and let*

$$H_j := span\left(X_1, \cdots, X_{j-1}, X_{j+1}, \cdots, X_n\right).$$

Then for any $\rho, \lambda \in (0,1)$ and $\varepsilon > 0$, one has

$$\mathbb{P}\left(\inf_{v \in Incomp_q(\lambda,\rho)} \|Av\|_q < \varepsilon \rho n^{-\frac{1}{q}}\right) < \frac{1}{\lambda n} \sum_{j=1}^{n} \mathbb{P}\left(dist_q\left(X_j, H_j\right) < \varepsilon\right), \tag{5.12}$$

in which $dist_q$ is the distance defined by the ℓ_q-quasinorm.

Proof. For every $v \in Incomp_q(\lambda, \rho)$, by Definition 5.1, there are at least λn components of v, v_j, satisfying $|v_j| \geq \rho n^{-\frac{1}{q}}$, because otherwise, v would be within ℓ_q-distance ρ of the sparse vector, the restriction of v on the components v_j satisfying $|v_j| \geq \rho n^{-\frac{1}{q}}$ with sparsity less than λn, and thus v would be compressible. Thus if we let $\mathcal{I}_1(v) := \left\{j : |v_j| \geq \rho n^{-\frac{1}{q}}\right\}$, then $|\mathcal{I}_1(v)| \geq \lambda n$.

Next, let $\mathcal{I}_2(A) := \{j : dist_q(X_j, H_j) < \varepsilon\}$ and $\mathcal{E}$ be the event such that for the cardinality of $\mathcal{I}_2(A)$, $|\mathcal{I}_2(A)| \geq \lambda n$. Applying Markov's inequality, we have

$$\begin{aligned}
\mathbb{P}(\mathcal{E}) &= \mathbb{P}\left(\{\mathcal{I}_2(A) : |\mathcal{I}_2(A, \varepsilon)| \geq \lambda n\}\right) \\
&\leq \frac{1}{\lambda n} \mathbb{E}\left|\mathcal{I}_2(A)\right| \\
&= \frac{1}{\lambda n} \mathbb{E}\left\{j : dist_q\left(X_j, H_j\right) < \varepsilon\right\} \\
&= \frac{1}{\lambda n} \sum_{j=1}^{n} \mathbb{P}\left(dist_q\left(X_j, H_j\right) < \varepsilon\right).
\end{aligned}$$

Since $\mathcal{E}^c$ is the event such that

$$\left|\{j : dist_q(X_j, H_j) \geq \varepsilon\}\right| > (1 - \lambda) n$$

for random matrix A, if $\mathcal{E}^c$ occurs, then for every $v \in Incomp_q(\lambda, \rho)$,

$$|\mathcal{I}_1(v)| + |\mathcal{I}_2(A)| > \lambda n + (1 - \lambda) n = n.$$

Hence there is some $j_0 \in \mathcal{I}_1(v) \cap \mathcal{I}_2(A)$. So we have

$$\|Av\|_q \geq dist_q\left(Av, H_{j_0}\right) = dist_q\left(v_{j_0} X_{j_0}, H_{j_0}\right) = |v_{j_0}|\, dist_q\left(X_{j_0}, H_{j_0}\right) \geq \varepsilon \rho n^{-\frac{1}{q}}.$$

If the events $\|Av\|_q < \varepsilon \rho n^{-\frac{1}{q}}$ occur, then $\mathcal{E}$ also occurs. Thus

$$\mathbb{P}\left(\inf_{v \in Incomp_q(\lambda,\rho)} \|Av\|_q < \varepsilon \rho n^{-\frac{1}{q}}\right) \leq \mathbb{P}(\mathcal{E}) \leq \frac{1}{\lambda n} \sum_{j=1}^{n} \mathbb{P}\left(dist_q\left(X_j, H_j\right) < \varepsilon\right).$$

These complete the proof. □

Note that $dist_q(X_j, H_j) \geq dist(X_j, H_j)$ because $\|\cdot\|_q \geq \|\cdot\|_2$. Thus we can take the advantage of the estimate on $\mathbb{P}(dist(X_j, H_j) < \varepsilon)$ given in [15] to obtain the estimate on $\mathbb{P}(dist_q(X_j, H_j) < \varepsilon)$.

Theorem 5.3 (Distance bound (cf. [15])). *Let A be a random matrix whose entries are independent variables with variance at least 1 and fourth moment bounded by B. Let $K \geq 1$. Then for every $\varepsilon > 0$,*

$$\mathbb{P}\left(dist\left(X_j, H_j\right) < \varepsilon \text{ and } \|A\| \leq K n^{-\frac{1}{2}}\right) \leq C\left(\varepsilon + \alpha^n\right), \tag{5.13}$$

where $\alpha \in (0,1)$ and $C > 0$ depend only on B and K.

The above theorem implies that

$$\mathbb{P}\left(dist_q\left(X_j, H_j\right) < \varepsilon\right) \leq \mathbb{P}\left(dist\left(X_j, H_j\right) < \varepsilon\right) \leq C\left(\varepsilon + \alpha^n\right) + \mathbb{P}\left(\|A\| \leq K n^{-\frac{1}{2}}\right). \tag{5.14}$$

Combining (5.10) and applying Lemma 5.4, we now reach the desired inequality in Theorem 5.2.

Furthermore, since A is pre-Gaussian, using a standard concentration bound we know that for every $\varepsilon > 0$ there exists some $K > 0$ such that $\mathbb{P}\left(\|A\| \leq K n^{-\frac{1}{2}}\right) < \varepsilon$. Thus, we have proved Theorem 1.6.

6. The upper tail probability of the smallest q-singular value

In this section, we continue to study the estimate of the upper tail probability of the smallest q-singular value of an $n \times n$ pre-Gaussian matrix. Mainly we are going to prove Theorem 1.7. To do so, we need some preparation.

Let X_j be the j-th column vector of A and π_j be the projection onto the subspace $H_j := span\left(X_1, \ldots, X_{j-1}, X_{j+1}, \cdots, X_n\right)$. We first have

Lemma 6.1. *For every $\alpha > 0$, one has*

$$\mathbb{P}\left(\left\|X_j - \pi_j\left(X_j\right)\right\|_q \geq \alpha n^{\frac{1}{q}-\frac{1}{2}}\right) \leq c_1 e^{-c_2\alpha} + c_3 n^{-c_4} \tag{6.1}$$

for each $j = 1, 2, \ldots, n$, where $c_1, c_2, c_3, c_4 > 0$ are constants independent of j, n, and q.

Proof. Without loss of generality, assume $j = 1$. Write $(a_1, a_2, \ldots, a_n) := X_1 - \pi_1\left(X_1\right)$. Applying the Bessy-Esseen theorem (see for instance [21]), we know that

$$\mathbb{P}\left(\left\|X_j - \pi_j\left(X_j\right)\right\|_2 \geq \alpha\right) = \mathbb{P}\left(\left|\frac{\sum_{i=1}^n a_i\xi_i}{\sqrt{\sum_{i=1}^n a_i^2}}\right| \geq \alpha\right) = \mathbb{P}\left(|\mathbf{g}| \geq \alpha\right) + O\left(n^{-c}\right) \tag{6.2}$$

for some $c > 0$, where $\mathbf{g}$ is a standard normal random variable.

By the Hölder inequality,

$$\left\|X_j - \pi_j\left(X_j\right)\right\|_q \leq n^{\frac{1-q}{q}} \left\|X_j - \pi_j\left(X_j\right)\right\|_1 \leq n^{\frac{1}{q}-\frac{1}{2}} \left\|X_j - \pi_j\left(X_j\right)\right\|_2.$$

It follows that

$$\begin{aligned} \mathbb{P}\left(\left\|X_j - \pi_j\left(X_j\right)\right\|_q \geq n^{\frac{1}{q}-\frac{1}{2}}\alpha\right) &\leq \mathbb{P}\left(n^{\frac{1}{q}-\frac{1}{2}} \left\|X_j - \pi_j\left(X_j\right)\right\|_2 \geq n^{\frac{1}{q}-\frac{1}{2}}\alpha\right) \\ &= \mathbb{P}\left(\left\|X_j - \pi_j\left(X_j\right)\right\|_2 \geq \alpha\right). \end{aligned}$$

Therefore it follows from (6.2) that

$$\begin{aligned} \mathbb{P}\left(\left\|X_j - \pi_j\left(X_j\right)\right\|_q \geq \alpha n^{\frac{1}{q}-\frac{1}{2}}\right) &\leq \mathbb{P}\left(|\mathbf{g}| \geq \alpha\right) + O\left(n^{-c}\right) \\ &= \frac{2}{\sqrt{2\pi}} \int_\alpha^\infty e^{-\frac{1}{2}x^2} dx + O\left(n^{-c}\right) \\ &\leq c_1 e^{-c_2\alpha} + c_3 n^{-c_4} \end{aligned}$$

for some positive constants c_1, c_2, c_3, c_4. ∎

We are now ready to prove Theorem 1.7.

Proof of Theorem 1.7. From Section 5.2 and by Lemma 2.4, we know that the $n\times n$ pre-Gaussian matrix A is invertible with very high probability. Thus, we have

$$\mathbb{P}\left(s_n^{(q)}(A)\le\frac{\alpha t}{\varepsilon}\cdot n^{-1/q}\right)\ge\mathbb{P}\left(\|v\|_q\le\alpha,\left\|A^{-1}v\right\|_q\ge\frac{\varepsilon}{t}\cdot n^{1/q}\text{ for some }v\in\mathbb{R}^n\right). \tag{6.3}$$

Thus it suffices to show that

$$\mathbb{P}\left(\|v\|_q\le\alpha,\left\|A^{-1}v\right\|_q\ge\frac{\varepsilon}{t}\cdot n^{1/q}\right)\ge 1-\varepsilon \tag{6.4}$$

for some vector $v\neq 0$.

Using the result established in Rudelson and Vershynin'2008, [14], we can easily get the desired probability of the event that $\left\|A^{-1}v\right\|_q\le\frac{\varepsilon}{t}\cdot n^{\frac{1}{q}}$ occurs. Indeed, since $\left\|A^{-1}v\right\|_q\ge\left\|A^{-1}v\right\|_2$, we know that

$$\begin{aligned}\mathbb{P}\left(\left\|A^{-1}v\right\|_q\le\tfrac{\varepsilon}{t}\cdot n^{-1/q}\right) &\le \mathbb{P}\left(\left\|A^{-1}v\right\|_2\le\tfrac{\varepsilon}{t}\cdot n^{1/q}\right)\\ &= \mathbb{P}\left(\left\|A^{-1}v\right\|_2\le\tfrac{\varepsilon}{t}\cdot\left(n^{2/q}\right)^{1/2}\right)\\ &\le 2p\left(4\varepsilon,t,n^{2/q}\right),\end{aligned} \tag{6.5}$$

where $p(\varepsilon,t,n):=c_5\left(\varepsilon+e^{-c_6t^2}+e^{-c_7n}\right)$ for some positive constants c_5,c_6,c_7.

Next let us choose $v=X_1-\pi_1(X_1)$. Lemma 6.1 together with the estimate in (6.5) yield (6.4). Indeed, letting $u=t=\sqrt{\ln M}$ with $M>1$ and $\varepsilon=\frac{1}{M}$, we have

$$\mathbb{P}\left(s_n^{(q)}(A)>M\ln M\cdot n^{-1/2}\right)\le\frac{C}{M^\alpha}+c^n \tag{6.6}$$

for some $C>0$, $0<c<1$, and $\alpha>0$. Then choosing $K:=M\ln M$, we have

$$\mathbb{P}\left(s_n^{(q)}(A)>Kn^{-1/2}\right)\le\frac{C\,(\ln M)^\alpha}{K^\alpha}+c^n\le\frac{C\,(\ln(M\ln M))^\alpha}{K^\alpha}+c^n=\frac{C\,(\ln K)^\alpha}{K^\alpha}+c^n \tag{6.7}$$

if $M\ge e$, which requires $K>e$. These complete the proof. □

Acknowledgment

The authors would like to thank the referees for useful comments.

References

[1] Z. D. Bai, Jack W. Silverstein, and Y. Q. Yin, *A note on the largest eigenvalue of a large-dimensional sample covariance matrix*, J. Multivariate Anal. **26** (1988), no. 2, 166–168, DOI 10.1016/0047-259X(88)90078-4. MR963829 (89i:62083)

[2] G. Bennett, V. Goodman, and C. M. Newman, *Norms of random matrices*, Pacific J. Math. **59** (1975), no. 2, 359–365. MR0393085 (52 #13896)

[3] V. V. Buldygin and Yu. V. Kozachenko, *Metric Characterization of Random Variables and Random Processes*, Translations of Mathematical Monographs, vol. 188, American Mathematical Society, Providence, RI, 2000. Translated from the 1998 Russian original by V. Zaiats. MR1743716 (2001g:60089)

[4] Rick Chartrand and Valentina Staneva, *Restricted isometry properties and nonconvex compressive sensing*, Inverse Problems **24** (2008), no. 3, 035020, 14, DOI 10.1088/0266-5611/24/3/035020. MR2421974 (2009d:94027)

[5] Alan Edelman, *Eigenvalues and condition numbers of random matrices*, SIAM J. Matrix Anal. Appl. **9** (1988), no. 4, 543–560, DOI 10.1137/0609045. MR964668 (89j:15039)

[6] Alan Edelman, *Eigenvalues and condition numbers of random matrices*, Ph.D. thesis, Massachusetts Institute of Technology, 1989. PRoQuest LLC. MR2941174

[7] A. Fisher, *The Mathematical Theory of Probabilities and its Application to Frequency Curves and Statistical Methods*, vol. 1, The Macmillan Company, 1922.

[8] Simon Foucart and Ming-Jun Lai, *Sparsest solutions of underdetermined linear systems via l_q-minimization for $0 < q \leq 1$*, Appl. Comput. Harmon. Anal. **26** (2009), no. 3, 395–407, DOI 10.1016/j.acha.2008.09.001. MR2503311 (2011b:65045)

[9] Simon Foucart and Ming-Jun Lai, *Sparse recovery with pre-Gaussian random matrices*, Studia Math. **200** (2010), no. 1, 91–102, DOI 10.4064/sm200-1-6. MR2720209 (2011g:15061)

[10] Stuart Geman, *A limit theorem for the norm of random matrices*, Ann. Probab. **8** (1980), no. 2, 252–261. MR566592 (81m:60046)

[11] Gene H. Golub and Charles F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, 1996. MR1417720 (97g:65006)

[12] Gilles Pisier, *The Volume of Convex Bodies and Banach Space Geometry*, Cambridge Tracts in Mathematics, vol. 94, Cambridge University Press, Cambridge, 1989. MR1036275 (91d:52005)

[13] Herbert Robbins, *A remark on Stirling's formula*, Amer. Math. Monthly **62** (1955), 26–29. MR0069328 (16,1020e)

[14] Mark Rudelson and Roman Vershynin, *The least singular value of a random square matrix is $O(n^{-1/2})$* (English, with English and French summaries), C. R. Math. Acad. Sci. Paris **346** (2008), no. 15-16, 893–896, DOI 10.1016/j.crma.2008.07.009. MR2441928 (2009i:60104)

[15] Mark Rudelson and Roman Vershynin, *The Littlewood-Offord problem and invertibility of random matrices*, Adv. Math. **218** (2008), no. 2, 600–633, DOI 10.1016/j.aim.2008.01.010. MR2407948 (2010g:60048)

[16] Mark Rudelson, *Invertibility of random matrices: norm of the inverse*, Ann. of Math. (2) **168** (2008), no. 2, 575–600, DOI 10.4007/annals.2008.168.575. MR2434885 (2010f:46021)

[17] Mark Rudelson and Roman Vershynin, *Non-asymptotic Theory of Random Matrices: Extreme Singular Values*, Proceedings of the International Congress of Mathematicians. Volume III, Hindustan Book Agency, New Delhi, 2010, pp. 1576–1602. MR2827856 (2012g:60016)

[18] Jack W. Silverstein, *The smallest eigenvalue of a large-dimensional Wishart matrix*, Ann. Probab. **13** (1985), no. 4, 1364–1368. MR806232 (87b:60050)

[19] Steve Smale, *On the efficiency of algorithms of analysis*, Bull. Amer. Math. Soc. (N.S.) **13** (1985), no. 2, 87–121, DOI 10.1090/S0273-0979-1985-15391-1. MR799791 (86m:65061)

[20] Alexander Soshnikov, *A note on universality of the distribution of the largest eigenvalues in certain sample covariance matrices*, J. Statist. Phys. **108** (2002), no. 5-6, 1033–1056, DOI 10.1023/A:1019739414239. Dedicated to David Ruelle and Yasha Sinai on the occasion of their 65th birthdays. MR1933444 (2003h:62108)

[21] Daniel W. Stroock, *Probability Theory*, 2nd ed., Cambridge University Press, Cambridge, 2011. An analytic view. MR2760872 (2012a:60003)

[22] Stanisław J. Szarek, *Condition numbers of random matrices*, J. Complexity **7** (1991), no. 2, 131–149, DOI 10.1016/0885-064X(91)90002-F. MR1108773 (92i:65086)

[23] Terence Tao and Van Vu, *On the singularity probability of random Bernoulli matrices*, J. Amer. Math. Soc. **20** (2007), no. 3, 603–628, DOI 10.1090/S0894-0347-07-00555-3. MR2291914 (2008h:60027)

[24] Terence Tao and Van Vu, *Random matrices: the circular law*, Commun. Contemp. Math. **10** (2008), no. 2, 261–307, DOI 10.1142/S0219199708002788. MR2409368 (2009d:60091)

[25] Terence Tao and Van Vu, *On the permanent of random Bernoulli matrices*, Adv. Math. **220** (2009), no. 3, 657–669, DOI 10.1016/j.aim.2008.09.006. MR2483225 (2010b:15014)

[26] Terence Tao and Van Vu, *Random matrices: the distribution of the smallest singular values*, Geom. Funct. Anal. **20** (2010), no. 1, 260–297, DOI 10.1007/s00039-010-0057-8. MR2647142 (2011m:60020)

[27] Terence Tao and Van Vu, *Smooth analysis of the condition number and the least singular value*, Math. Comp. **79** (2010), no. 272, 2333–2352, DOI 10.1090/S0025-5718-2010-02396-8. MR2684367 (2011k:65065)

[28] John von Neumann and H. H. Goldstine, *Numerical inverting of matrices of high order*, Bull. Amer. Math. Soc. **53** (1947), 1021–1099. MR0024235 (9,471b)

[29] Eugene P. Wigner, *On the distribution of the roots of certain symmetric matrices*, Ann. of Math. (2) **67** (1958), 325–327. MR0095527 (20 #2029)

[30] J. Wishart, *The generalised product moment distribution in samples from a normal multivariate population*, Biometrika **20** (1928), no. 1/2, 32–52.

[31] Y. Q. Yin, Z. D. Bai, and P. R. Krishnaiah, *On the limit of the largest eigenvalue of the large-dimensional sample covariance matrix*, Probab. Theory Related Fields **78** (1988), no. 4, 509–521, DOI 10.1007/BF00353874. MR950344 (89g:60117)

Department of Mathematics, The University of Georgia, Athens, Georgia 30602
E-mail address: mjlai@math.uga.edu

Department of Mathematics, Michigan State University, East Lansing, Michigan 488244-1027
E-mail address: yliu@math.msu.edu

A New Approach for Analyzing Physiological Time Series

Dong Mao, Yang Wang, and Qiang Wu

July 28, 2010

1 Introduction

An understanding of physiological time series such as the heart-beat intervals is important to many areas, like heart-attack prediction, cardiovascular health, sport and exercise, etc. The study of time series can reveal underlying mechanisms of the physiological system, which usually contains both deterministic and stochastic components. Therefore the analysis of time series is very complicated because of the nonlinear and non-stationary characteristics of physiological time series data. Over the past years, time series analysis methods are applied to quantify physiological data for identification and classification (see [7, 12]). The application of physiological time series analysis commonly focus on measuring different aspects of time series data such as complexity, regularity, predictability, dimensionality, randomness, self similarity, etc. The tools used in these techniques include but not restrict to the mean, standard deviation, Fourier transform, wavelet, entropy, fractal

dimension, pattern detection (see [8, 13]).

Recently a new mathematical tool, empirical mode decomposition (EMD), was proposed by Norden Huang et al (see [5, 6]). It decomposes a time series into a finite sum of intrinsic mode functions (IMF) that generally admit well-behaved Hilbert transforms. This decomposition is based on the local characteristic time scale of the data, which makes EMD applicable to analyze nonlinear and non-stationary signals. EMD and Hilbert transform together, called the Hilbert-Huang transform (HHT), usually allow to construct meaningful time-frequency representations of signals using instantaneous frequency of the data. EMD and HHT have been applied with great success in many application areas such as biological and medical sciences, geology, astronomy, engineering, and others (see [5, 1, 3, 6, 11, 10]). Another interesting set of examples is the work of L.Yang, who has successfully applied EMD based techniques for texture analysis and Chinese handwriting recognition (see [16, 17, 15, 18]).

The main purpose of this paper is to develop a new approach for the analysis of physiological times series. Our approach is motivated by two intuitions and coupled with modern machine learning techniques. The first intuition comes from a belief that a physiological system should contain a deterministic part that reflects the basic mechanism for the system to survive and a stochastic part that represents the variability of resilience. Mathematically they can be represented by the low frequency and high frequency components of a physiological signal. This motivates the application of methods of decomposing signals into various components according to frequencies in the quantitative analysis of physiological time series. Examples include

the Fourier transform, wavelets, EMD. In our method we will use an iterative convolution filter which is an alternative of EMD. The second intuitions comes from a statistical perspective of irregularity. A lot of study has proved that normal physiological systems show irregularity due to the existence of stochastic components while the decrease of irregularity usually imply the abnormality. From statistical perspective, irregularity of a data set is represented by the "outliers". This motivates us to study the statistics of outliers in physiological time series. However, we must be careful in dong so. Practical physiological times series usually contains noise which may also appear as outliers. We have to guarantee the "outliers" we examined are not pure noise. This is possible because true outliers do not have informative structures and could be detected. The second intuition is the motivation for our feature construction in section 2.2.

These two intuitions enable us to decompose the physiological times series and construct features for our quantitative analysis. Combining with the well established feature selection techniques in machine learning we can remove the redundancy of the features and find relevant statistics for classification of physiological time series. SVM-RFE (Support Vector MachineRecursive Feature Elimination) is suggested in this paper for linear classification problems. The details of our approach will be described in Section 2.

We will use our approach to the study of congestive heart failure problems. The purposes is two-fold: The first is to build good classifier to enable good diagnosis. The second is to find what kind of irregularity is related to the heart health. The results and discussions are summarized in Section 3.

The novelty of our method is mainly the following two points. Firstly,

although we decompose the time series into components of different frequencies, we do not compare them from the frequency domain. Secondly, we proved that the outliers in a physiological time series are usually not true outliers but are informative instead.

2 Method

2.1 Signal decomposition

Let L be a low pass filter. Denote by T the weak limit of the the operator $(I-L)^n$ as $n \to \infty$, i.e., for a discrete signal X and time t

$$T(X)(t) = \lim_{n\to\infty} (I-L)^n(X)(t).$$

Using this operator iteratively, a signal X can be decomposed as follows: Let $F_1 = T(X)$ and for $k \geq 2$,

$$F_k = T\left(X - \sum_{i=1}^{k-1} F_i\right).$$

After m steps we get $F_1, \ldots, F_m$ which we call mode functions and the residual

$$R = X - \sum_{i=1}^{m} F_i.$$

Then we have

$$X = F_1 + F_2 + \ldots + F_m + R.$$

In this decomposition, roughly speaking the former mode functions are noise or high frequency components and the latter mode functions are low frequency components and R is the trend.

This procedure follows the spirit of the traditional EMD introduced in [5]. In the traditional EMD, the low pass filter L is chosen as the average of the upper envelope (the cubic spline connecting the local maxima) and the lower envelope (the cubic spline connecting the local minima). This method, although has been successfully used in many applications, is lack of theoretical foundation and has its limitations.

In [9] a new approach is proposed. In this new approach the low pass filter is a moving average generated by a mask $\mathbf{a} = (a_j)_{j=-N}^{N}$ that gives the $L(X)$ as the convolution of a and X, i.e.,

$$L(X)(t) = \sum_{j=-N}^{N} a_j X(j+t).$$

With this choice of L we call the operator T an iterative convolution filter. A rigorous mathematical foundation and convergence analysis is in [9, 14]. Note the mask $\mathbf{a}$ is finitely supported on $[-N, N]$ and N is called the window size. The flexibility to choose the window size is crucial in applications and forms a main advantage of this method.

Similar to decompositions by many other methods like Fourier transform and wavelets , the trend and low frequency components are usually assumed to characterize the profile of the signal and the high frequency components characterize the details. In different applications we need the features of difference components.

2.2 Feature extraction

After decomposing the signal into the mode functions and the trend, we need to extract statistics that can characterize the essential features of these

components. This step requires a priori knowledge of the problem under consideration. It could be rather weak. But without any priori knowledge, it is difficult to get proper statistics. Also, this step is strongly problem dependent. In the following let us use the heart-beat intervals as an example to illustrate how to construct the features.

For each mode function F_i, we first get its mean m_i and standard deviation σ_i. By the previous studies [2] the healthy heart beats more irregularly than the unhealthy heart. This motivates us to design the statistics to measure the irregularity. To this end, we consider the terms that are larger than $m + \sigma$ and find their mean and standard deviation. We also find the mean and standard deviation of the terms that are larger than $m + 2\sigma$. Symmetrically we also get the mean and standard deviation of those terms that are smaller than $m - \sigma$ and $m - 2\sigma$. Note all these terms are in some sense "outliers" and it is natural to use the statistics of the outliers as the characterization of the irregularity.

Next we consider the local maxima and local minima of F_i. These two series measure the local upper amplitude. For each series we consider the ten statistics as those for F_i.

Therefore for each component we get 30 statistics.

Unlike in [2], we use the whole 24-hour heart beat time series and assume we do not know the periods for different activities such as sleeping and walking. We think the statistics for different periods should be different and not all of them represent the difference between the healthy and unhealthy people. This motivates the idea of split the whole time series into subseries. Suppose we have K subseries for each patient. Then we get K subcompo-

nents for each mode function which will be denoted by $F_{ij},\ j = 1, \ldots, K$. For each subcomponent F_{ij} we also get the 30 statistics as for F_i. For the same statistics, we have K values from the K subcomponents. We compute the mean of all values, the mean of lower half and upper half, respectively. This gives 90 statistics as summary. So for each component we get 120 statistics.

For physiological signals, we believe the trend and low frequency components are determined by the fundamental mechanism while the individual differences should be reflected by the high frequency components. In case that we do not have much knowledge about the disease to be diagnosed we may assume the features may also comes from the trend. So the same 120 statistics are also computed for the trend component.

2.3 Feature subset selection

After the above two steps we have get many features for the data. Usually only a small part of them are related to the diagnosis and the physiological mechanism of the disease. The task of the third step is to find the relevant ones. This will be realized by eliminating the irrelevant ones step by step.

Firstly, if a statistic is almost constant, then it is useless in the diagnosis and should be eliminated. For example, the means of the mode functions m_i are all approximately zero and should be eliminated.

Next we use the SVM-RFE method [4] to rank the features. In this method, given a set of training samples, we first train linear SVM to get a classifier and then rank the features according to the weights. Because of large feature size and small training samples, the classifier might not be as good. Also, the high correlation between the features may result the

relevant features to have small weights. These reasons could lead the rank to be inaccurate. In order to refine the rank we eliminate the least important feature and repeat the process to re-rank the remained features. Running this process iteratively we finally get the refined rank of the features.

With this rank of features we can conclude which statistics are useful for the diagnosis and characterize the essence of the underlying physiological mechanism. Good classifiers can then be built to make accurate diagnosis.

3 Experiments and Results

In this section we apply our new method described in Section 2 to the hear beat interval times series and report our results and conclusions.

3.1 The data set

The data set includes the heart beat interval time series of 72 healthy people and 43 CHF patients. For each people the heart beat interval is measured for 24 hours under various activities. In our experiment we will assume the activity period is not known. The average ages of these two groups are both 55 years. The standard deviation of age of CHF patients is 11 years and which of healthy people is 16 years. If divide CHF patients into 4 degrees where the degree I is a slight CHF and the degree IV is a severe CHF, most CHF patients are of the degree III.

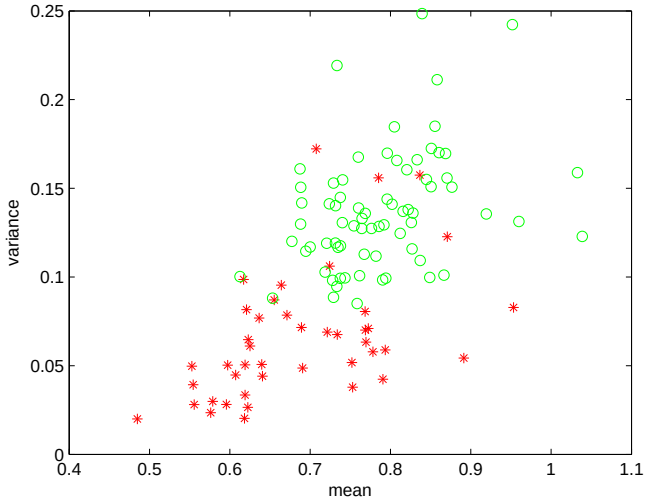


Figure 1: The mean and variance (in second) of the times series, 'o' for healthy people and '*' for CHF patients.

3.2 A primary study

Before using our new method, we study the classification ability of two simple statistics: mean and variance. In Figure 1 we plot the mean and variance of the heart beat intervals for the healthy people and CHF patients. We see that the healthy people and the CHF patients can be roughly separated. The average heart beat interval of healthy people is larger and so is the variance. It shows the heart of healthy people beats slower and more irregularly. This observation coincides with the previous study.

At the same time, we notice that several cases falling into the healthy people show to be severe CHF patients. So we conjecture that the mean and variance might not reflect the essence of the underlying mechanism, although

they have good separability.

3.3 Experiment: feature extraction

For each time series, we use the iterative convolution filter to realize the signal decomposition. In this step we need to specify the window size of the mask. It turns out it should be chosen between 50 and 100 to be stable. In our experiment it is chosen to be 50.

We then calculate the statistics proposed in Subsection 2.2. Here we need to specify the parameter K, the number of subseries. If a statistic really captures the essence of the data set, it should be stable and independent of the choice of K once it is chosen within a reasonable interval. Our experiments show that $K = 50$ is a good choice. Most heart beat signals were recorded for a little bit more than 24 hours. Thus when $K = 50$, each subseries is around 30 minutes of record.

Previous studies have shown that healthy heart beats irregularly. In statistics, irregularity could be measured by statistics of "outliers" that are not due to noise. This motivates us to consider the upper half mean and the lower half mean of the fluctuations. At the same time, from the study in Section 3.2 we find that a healthy heart beats slower than an unhealthy heart in average. These two intuitions enlighten us to conjecture that those larger heart beat intervals (i.e. slower heart beats) in the times series characterize the difference between the healthy people and CHF patients. To confirm this, we do a correlation analysis.

For the first two IMFs of the 50 components of each time series, we calculate and sort the mean and standard deviation of those terms larger

than mean plus standard deviation and those terms larger than mean plus two times standard deviation. For each statistic we compute its correlation to the CHF disease. The result is plotted in Figure 2 in red color. We compute the same indices for those items smaller than the mean minus one and two times standard deviation. The result is plotted in Figure 2 in blue color. From the comparison we see that, in average, correlations of the statistics associated to the larger fluctuations are larger and the upper half mean of these statistics are stable. This observation motives us to disregard the smaller fluctuations and the statistics for those.

3.4 Feature ranking and subset selection

To rank the features, we randomly split the data set into two subsets as the training set and the test set, respectively. In the training set we have 50 healthy subjects and 30 CHF subjects and in the test set there are 22 healthy and 13 CHF subjects. We use the training set to build the SVM classifier and use the test set to control the accuracy. Using the SVM-RFE methods described in Subsection 2.3 we rank the features. To guarantee the stability of the rank we repeat this procedure 1000 times and choose the statistics that appear most frequently in the model.

In all 1000 repeats, the classification error on the test data set is summarized in the following table:

number of errors	0	1	2	3	4	5
number of repeats	823	116	42	14	4	1

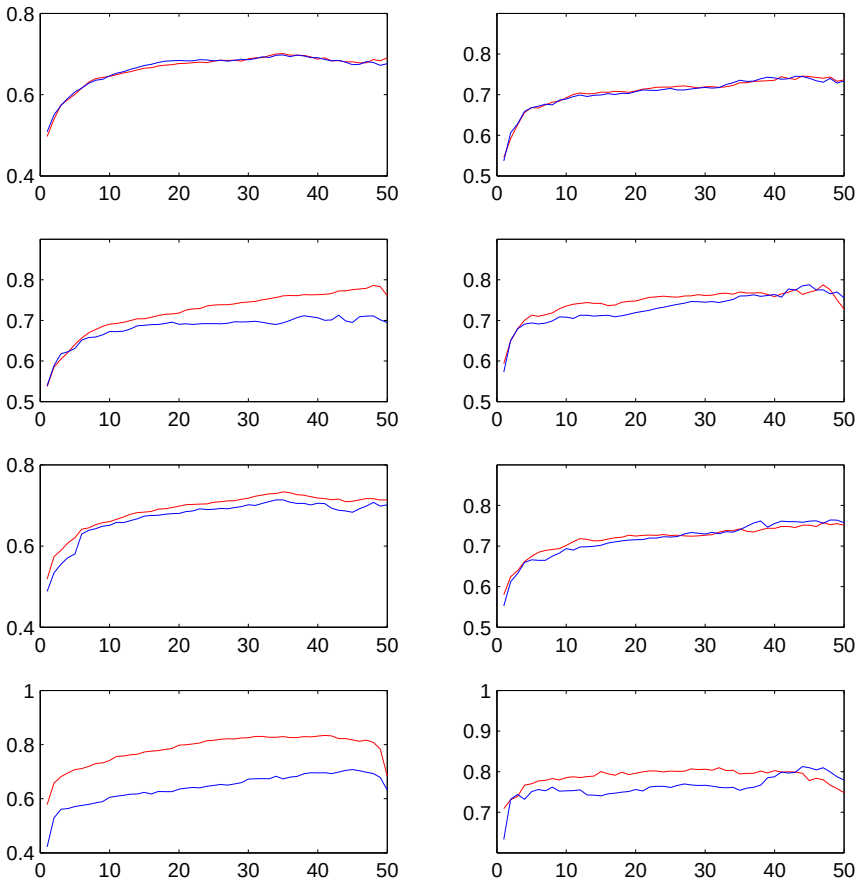


Figure 2: The correlations of various statistics to the CHF disease. The first column is for the first IMF and the second column is for the second IMF. The first line is for the mean of those items larger than the mean plus standard deviation (red line) and those items smaller than the mean minus the standard deviation (blue line). The second line is for the standard deviation of two types items. The third line is for the mean of those items larger than the mean plus 2 times standard deviation (red line) and those items smaller than the mean minus 2 times standard deviation (blue line). The forth line is for the standard deviation of two types of items.

We list the top 10 statistics selected by this procedure:

1. IMF 1: For the subseries consisting of local maxima, find all terms which are greater than the mean plus two times standard deviation, then compute the standard deviation.

2. IMF 1: For the subseries consisting of local maxima, find all terms which are less than the mean minus two times standard deviation, then compute the standard deviation.

3. IMF 1: Equally divide the series into K subseries, for each subseries find all terms which are less than the mean minus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations.

4. IMF 1: Equally divide the series into K subseries, find local maxima of each subseries, find all terms of local maxima which are greater than the mean plus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations.

5. IMF 1: Equally divide the series into K subseries, find local minima of each subseries, find all terms of local minima which are greater than the mean plus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations .

6. IMF 2: Find all terms which are greater than the mean plus two times standard deviation, then compute the standard deviation.

7. IMF 2: Equally divide the series into K subseries, for each subseries find all terms which are greater than the mean plus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations.

8. IMF 2: Equally divide the series into K subseries, find local maxima of each subseries, find all terms of local maxima which are greater than the mean plus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations.
9. IMF 2: Equally divide the series into K subseries, find local minima of each subseries, find all terms of local minima which are less than the mean minus two times standard deviation, compute the standard deviation, then take the mean of these K standard deviations.
10. Trend: Equally divide the series into K subseries, find local maxima of each subseries, find all terms of local minima which are greater than the mean plus standard deviation, compute the standard deviation, then take the mean of these K standard deviations.

These 10 statistics that appear most frequently in the model all measure the irregularity of the local amplitude. Take Statistics 1 and Statistics 7 as the example. They are obtained as the following. To get Statistics 1, for the first IMF F_1, find the local maxima u and compute the mean m and the standard deviation σ of u. Then we choose terms greater than $m + 2\sigma$ and find their standard deviation. To get Statistics 7, for the subseries of the second IMF $F_{2j}, j = 1, \ldots, K$, compute the mean m_{2j} and the standard deviation σ_{2j} of F_{2j}. Then we choose terms greater than $m_{2j} + 2\sigma_{2j}$ of F_{2j} and find their standard deviations. Then we compute the mean of K such standard deviations. In the following figure we show the distribution of the healthy people and CHF patients using these two statistics. From this figure it is easy to see that healthy people and CHF patients are well separated.

Observing these two statistics, we find that both of them measure the

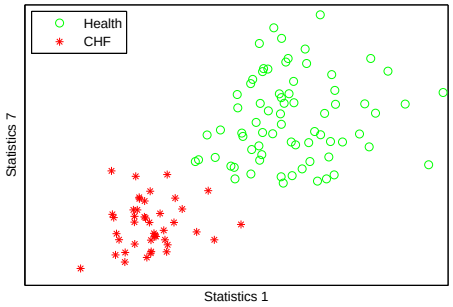


Figure 3: CHF * vs Healthy ∘. The x-axis is Statistics 1 and the y-axis is Statistics 7.

ability of the heart beat to become extremely slower than usual. Our result shows that the strong adaptability of extremely slower heart beat might be the irregularity that characterizes the healthy hearts.

3.4.1 Reliability of the top features

We have found that the most relevant features are statistics for the "outliers", i.e., those items larger than mean plus two times standard deviations, or items less than mean minus two times standard deviations for IMFs. A natural question arises: "Is this accidental?" This is equivalent to ask whether the outliers taken into account are noise or informative.

In order to answer this question we further analyze these outliers. Firstly we notice that the up and down fluctuations are not balanced for both healthy people and CHF patients. The percentage of items larger than mean plus two times standard deviation for healthy people is 2.84% and those items smaller than the mean minus stand deviation is only 2.35%. For CHF patients the

percentages are 2.49% and 2.17%, respectively. This observation is the first evidence that outliers are not due to noise because otherwise they should be balanced distributed. Moreover, recall for normal distribution the percentage of one-side outliers outside the two times standard deviation is 2.28%. We see the outliers for CHF is closer to it due to noise while those for healthy people are much more and probably due to not only noise and hence are informative.

To further confirm our conclusion, we do the following test: we calculate the statistics for the items larger than the mean plus v times standard deviation with the variable v changes from 0 to 2 and investigate their correlation to the CHF disease. Here we consider three quartile of the 50 standard deviations of these items in the 50 components. The correlation is plotted in Figure 4. From this analysis, we see the correlation increases with v. Such a trend appears also in other statistics. This clear trend implies that the relevancy between these statistics and the CHF disease is not accidental. Instead, we should consider the outliers informative and their properties characterize the essence difference between healthy people and CHF patients.

4 Conclusions and discussions

In this paper we developed a new approach for the analysis of the physiological times series. The motivation comes from that the physiological times series usually contains both deterministic and stochastic parts and they can be represented by the low and high frequency components of the times series. Our new method uses an iterative filter to realize the decomposition

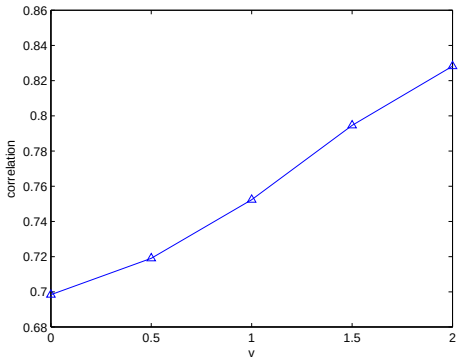


Figure 4: Corrections of the statistics described in Section 3.4.1 with v varying from 0 to 2.

of the times series into high and low frequency components and study their statistics. SVM-RFE is then used to select highly relevant features.

Our method is applied to analyze the heart beat interval time series for CHF disease. The top features are found to measure the ability of heart to beat extremely slowly. Healthy heart show strong ability which we conjecture are due to the strong resilience to the environment and human activities.

References

[1] Q. Chen, N. Huang, S. Riemenschneider, and Y. Xu. A B-spline approach for empirical mode decompositions. *Adv. Comput. Math.*, 24(1-4):171–195, 2006.

[2] M. Costa, A. L. Goldberger, and C.-K. Peng. Multiscalce entropy analysis of biological signals. *Physical Review, E*, 71:021906, 2005.

[3] J. Echeverria, J. Crowe, M. Woolfson, and B. Hayes-Gill. Application of empirical mode decomposition to heart rate variability analysis. *Medical and Biological Engineering and Computing*, 39:471–479, 2001.

[4] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422, 2002.

[5] N. Huang, S. Shen, Z.and Long, M. Wu, H. Shih, Q. Zheng, N. Yen, C. Tung, and L. H.H. Theempirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London, A*, 454:903–995, 1998.

[6] N. E. Huang, Z. Shen, and S. R. Long. A new view of nonlinear water waves: the Hilbert spectrum. In *Annual review of fluid mechanics, Vol. 31*, pages 417–457. Annual Reviews, Palo Alto, CA, 1999.

[7] H. Kantz, J. Kurths, and G. Mayer-Kress. *Nonlinear techniques in physiological time series analysis*. Springer series in synergetics. Springer, Heidelberg, 1998.

[8] H. Kantz and T. Schreiber. *Nonlinear time series analysis*, volume 7 of *Cambridge Nonlinear Science Series*. Cambridge University Press, Cambridge, 1997.

[9] L. Lin, Y. Wang, , and H. Zhou. A new approach to empirical mode decomposition. preprint, 2008.

[10] B. Liu, S. Riemenschneider, and X. Y. Gearbox fault diagnosis using emperical mode decomposition and hilbert spectrum. preprint.

[11] D. Pines and L. Salvino. Health monitoring of one dimensional structures using empirical mode decomposition and the hilbert-huang transform. In *Proceedings of SPIE*, volume 4701, pages 127 – 143, 2002.

[12] T. Schreiber. Interdisciplinary application of nonlinear time series methods. *Phys. Rep.*, 308(2), 1999.

[13] H. Tong. *Nonlinear Time Series Analysis*. Oxford University Press, Oxford, 1990.

[14] Y. Wang and Z. Zhou. Toeplitz operators in $\ell^\infty(F)$ and their applications to empirical mode decompositions. preprint, 2008.

[15] L. Yang, Z.and Yang and Y. Tang. Illumination-rotation-invariant feature extraction for texture classification based on hilbert-huang transform. preprint.

[16] Z. Yang, L. Yang, D. Qi, , and C. Suen. An emd-based recognition method for chinese fonts and styles. *Pattern Recognition Letter*, 27:1692–1701, 2006.

[17] Z. Yang, L. Yang, and D. Qi. Detection of spindles in sleep eegs using a novel algorithm based on the hilbert-huang transform. In T. Qian, M. I. Vai, and Y. Xu, editors, *Wavelet Analysis and Applications*, Applied and Numerical Harmonic Analysis, page 543C559. Birkhauser, 2006.

[18] T. Zheng, L. Xie, and L. Yang. Integrated extraction on handwritten numeral strings in form document. *Pattern Recognition and Artificial Intelligence*, 2009. in press.

Appl. Comput. Harmon. Anal. 38 (2015) 469–488

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Invertibility and robustness of phaseless reconstruction

Radu Balan [a,*], Yang Wang [b]

[a] Department of Mathematics, Center for Scientific Computation and Mathematical Modeling, University of Maryland, College Park, MD 20742, United States
[b] School of Mathematics, Michigan State University, East Lansing, MI 48824, United States

ARTICLE INFO

Article history:
Received 2 September 2013
Received in revised form 7 July 2014
Accepted 12 July 2014
Available online 17 July 2014
Communicated by Jared Tanner

Keywords:
Frames
Redundant representations
Phase retrieval
Phaseless reconstruction

ABSTRACT

This paper is concerned with the question of reconstructing a vector in a finite-dimensional real Hilbert space when only the magnitudes of the coefficients of the vector under a redundant linear map are known. We analyze various Lipschitz bounds of the nonlinear analysis map and we establish theoretical performance bounds of any reconstruction algorithm. The discussion of robustness is with respect to random noise and with respect to deterministic perturbations. We show that robust and uniformly stable reconstruction is not achievable with the minimum redundancy for phaseless reconstruction. Robust reconstruction schemes require additional redundancy than the critical threshold.

1. Introduction

This paper is concerned with the question of reconstructing a vector x in a finite-dimensional *real* Hilbert space H of dimension n when only the magnitudes of the coefficients of the vector under a redundant linear map are known.

Specifically our problem is to reconstruct $x \in H$ up to an overall change of sign from the magnitudes $\{|\langle x, f_k\rangle|,\ 1 \le k \le m\}$ where $\mathcal{F} = \{f_1, \ldots, f_m\}$ is a frame (complete system) for H.

A previous paper [6] described the importance of the phaseless reconstruction problem. One particular case is when the coefficients are obtained from an Undecimated Wavelet Transform. This case is relevant for instance in some audio and image signal processing applications, as well as in neural computations as performed by the auditory cortex [13].

While [6] presents some necessary and sufficient conditions for reconstruction, the general problem of finding fast/efficient algorithms is still open. In [3] we describe one solution in the case of STFT coefficients.

For vectors in real Hilbert spaces, the reconstruction problem is easily shown to be equivalent to a combinatorial problem. In [7] this problem is further proved to be equivalent to a (nonconvex) optimization problem.

* Corresponding author.
E-mail addresses: rvbalan@math.umd.edu (R. Balan), ywang@math.msu.edu (Y. Wang).

http://dx.doi.org/10.1016/j.acha.2014.07.003
1063-5203/

A different approach (which we called the *algebraic approach*) was proposed in [2]. While it applies to both real and complex cases, noiseless and noisy cases, the approach requires solving a linear system of size exponentially in the space dimension. This algebraic approach generalizes the approach in [8] where reconstruction is performed with complexity $O(n^2)$ (plus computation of the principal eigenvector for a matrix of size n). However this method requires $m = O(n^2)$ frame vectors.

Recently the authors of [10] developed a convex optimization algorithm (a SemiDefinite Program called *PhaseLift*) and proved its ability to perform exact reconstruction in the absence of noise, as well as its stability under noise conditions. In a separate paper [11], the authors further developed a similar algorithm in the case of windowed DFT transforms. Inspired by the PhaseLift and MaxCut algorithms, but operating in the coefficients space, the authors of [16] proposed a SemiDefinite Program called *PhaseCut*. They show the algorithm yields the exact solution in the absence of noise under similar conditions as PhaseLift.

The paper [4] presents an iterative regularized least-square algorithm for inverting the nonlinear map and compares its performance to a Cramer–Rao lower bound for this problem in the real case. The paper also presents some new injectivity results which are incorporated into this paper.

A different approach is proposed in [1]. There the authors use a 4-term polarization identity together with a family of spectral expander graphs to design a frame of bounded redundancy ($\frac{m}{n} \leq 236$) that yields an exact reconstruction algorithm in the absence of noise.

The authors of [14] study several robustness bounds to the phase recovery problem in the real case. However their approach is different from ours in several respects. First they consider a probabilistic setup of this problem, where data x and frame vectors f_j's are random vectors with probabilities from a class of subgaussian distributions. Additionally, their focus is on classes of k-sparse signals. In our paper we analyze stability bounds of reconstruction for a fixed frame using deterministic analytic tools. After that we present asymptotic behavior of these bounds for random frames.

Finally, the authors of [9] analyze the phaseless reconstruction problem for both the real and complex case. In the real case the authors obtain the exact upper Lipschitz constant for the nonlinear map $\alpha_{\mathcal{F}}$, namely $\sqrt{B}$ where B is the upper frame bound. For the lower Lipschitz constant, they give an estimate between two computable singular eigenvalues. Our results have overlaps with their results. However, in our paper we improve the lower Lipschitz constant by giving its exact value. There are some significant differences between this paper and [9]. In addition to studying of the Lipschitz property of the map $\alpha_{\mathcal{F}}$ we focus also on two related but different settings. First we study the robustness of the reconstruction given a fixed error allowance in measurements. Second we also consider the Lipschitz property of the map ${\alpha_{\mathcal{F}}}^2$. The authors of [9] point out that the map ${\alpha_{\mathcal{F}}}^2$ is not bi-Lipschitz. However in our paper we show ${\alpha_{\mathcal{F}}}^2$ becomes bi-Lipschitz for a different metric on the domain. With this metric (the one induced by the nuclear norm on the set of symmetric operators) the nonlinear map ${\alpha_{\mathcal{F}}}^2$ is bi-Lipschitz with constants indicated in Theorem 4.5. Furthermore the same conclusion holds true in the complex case, although this will be studied elsewhere.

The organization of the paper is as follows. Section 2 formally defines the problem and reviews existing inversion results in the real case. Section 3 establishes information theoretic performance bounds, namely the Cramer–Rao lower bound. Section 4 contains robustness measures of any reconstruction algorithm. Section 5 presents a stochastic analysis of these bounds. Section 6 presents a numerical example and is followed by references.

2. Background

Let us denote by $H = \mathbb{R}^n$ the n-dimensional real Hilbert space $\mathbb{R}^n$ with scalar product $\langle , \rangle$. Let $\mathcal{F} = \{f_1, \ldots, f_m\}$ be a spanning set of m vectors in H. In finite dimension (as it is the case here) such a set forms a *frame*. In the infinite dimensional case, the concept of frame involves a stronger property than completeness (see for instance [12]). We review additional terminology and properties which remain still

true in the infinite dimensional setting. The set $\mathcal{F}$ is a frame if and only if there are two positive constants $0 < A \leq B < \infty$ (called frame bounds) so that

$$A\|x\|^2 \leq \sum_{k=1}^{m} \big|\langle x, f_k\rangle\big|^2 \leq B\|x\|^2. \tag{2.1}$$

When we can choose $A = B$ the frame is said *tight*. For $A = B = 1$ the frame is called *Parseval*. The *frame matrix* corresponding to $\mathcal{F}$ is defined as $F = [f_1, f_2, \ldots, f_m]$ with the vectors $f_j \in \mathcal{F}$ as its columns. We shall frequently identify $\mathcal{F}$ with its corresponding frame matrix F. The largest A and smallest B in (2.1) are called the *lower frame bound* and *upper frame bound* of $\mathcal{F}$, and they are given by

$$A = \lambda_{\max}\big(FF^*\big) = \sigma_1^2(F), \qquad B = \lambda_{\min}\big(FF^*\big) = \sigma_n^2(F) \tag{2.2}$$

where $\lambda_{\max}, \lambda_{\min}$ denote the largest and smallest eigenvalues respectively, while σ_1, σ_n denote the first and n-th singular values respectively. A set of vectors $\mathcal{F}$ of the n-dimensional Hilbert space H is said to be *full spark* if any subset of n vectors is linearly independent.

For a vector $x \in H$, the collection of coefficients $\{\langle x, f_j\rangle : \ 1 \leq j \leq m\}$ represents the analysis map of vector x given by the frame $\mathcal{F}$, and from which x can be completely reconstructed. In the phaseless reconstruction problem, we ask the following question: Can x be reconstructed from $\{|\langle x, f_j\rangle| : \ 1 \leq j \leq m\}$? Consider the following equivalence relation $\sim$ on H: $x \sim y$ if and only if $y = cx$ for some unimodular constant c, $|c| = 1$. Since we focus on the real vector space $H = \mathbb{R}^n$, we have $x \sim y$ if and only if $x = \pm y$. Clearly the phaseless reconstruction problem cannot distinguish x and y if $x \sim y$, so we will be looking at reconstruction on $\hat{H} := H/\sim = \mathbb{R}^n/\sim$ whose elements are given by equivalent classes $\hat{x} = \{x, -x\}$ for $x \in \mathbb{R}^n$. The analogous analysis map for phaseless reconstruction is the following nonlinear map

$$\alpha_{\mathcal{F}} : \hat{H} \to \mathbb{R}_+^m, \qquad \alpha_{\mathcal{F}}(\hat{x}) = \big[\big|\langle x, f_1\rangle\big|, \big|\langle x, f_2\rangle\big|, \ldots, \big|\langle x, f_m\rangle\big|\big]^T. \tag{2.3}$$

Note that $\alpha_{\mathcal{F}}$ can also be viewed as a map from $\mathbb{R}^n$ to $\mathbb{R}_+^m$. Throughout the paper we will not make an explicit distinction unless such a distinction is necessary.

Thus the phaseless reconstruction problems aims to reconstruct $\hat{x} \in \hat{H}$ from the map $\alpha_{\mathcal{F}}(x)$. We say a frame $\mathcal{F}$ is *phase retrievable* if one can reconstruct $\hat{x} \in \hat{H}$ for all $\hat{x}$, or in other words, $\alpha_{\mathcal{F}}$ is injective on $\hat{H}$. The main objective of this paper is to analyze robustness and stability of the inversion map, and to give performance bounds of any reconstruction algorithm.

Before proceeding further we first review existing results on injectivity of the nonlinear map $\alpha_{\mathcal{F}}$. In general a subset Z of a topological space is said *generic* if its open interior is dense. However in the following statements, the term *generic* refers to Zarisky topology: a set $Z \subset \mathbb{K}^{n\times m} = \mathbb{K}^n \times \cdots \times \mathbb{K}^n$ is said *generic* if Z is dense in $\mathbb{K}^{n\times m}$ and its complement is a finite union of zero sets of polynomials in nm variables with coefficients in the field $\mathbb{K}$ (here $\mathbb{K} = \mathbb{R}$).

Theorem 2.1. *Let $\mathcal{F}$ be a frame in $H = \mathbb{R}^n$ with m elements. Then the following hold true:*

1. *The frame $\mathcal{F}$ is phase retrievable in $\hat{H}$ if and only if for any disjoint partition of the frame set $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2$, either $\mathcal{F}_1$ spans $\mathbb{R}^n$ or $\mathcal{F}_2$ spans $\mathbb{R}^n$.*
2. *If $\mathcal{F}$ is phase retrievable in $\hat{H}$ then $m \geq 2n-1$. Furthermore, for a generic $\mathcal{F}$ with $m \geq 2n-1$ the map $\alpha_{\mathcal{F}}$ is phase retrievable in $\hat{H}$.*
3. *Let $m = 2n-1$. Then $\mathcal{F}$ is phase retrievable in $\hat{H}$ if and only if $\mathcal{F}$ is full spark.*

4. *Let*

$$a_0 := \min_{\|x\|=\|y\|=1} \sum_{j=1}^{m} \big|\langle x, f_j\rangle\big|^2 \big|\langle y, f_j\rangle\big|^2 \geq 0, \tag{2.4}$$

so that

$$\sum_{k=1}^{m} \big|\langle x, f_k\rangle\big|^2 \big|\langle y, f_k\rangle\big|^2 \geq a_0 \|x\|^2 \|y\|^2. \tag{2.5}$$

Then $\mathcal{F}$ is phase retrievable on $\hat{H}$ if and only if $a_0 > 0$.

5. *For any $x \in \mathbb{R}^n$ define the matrix $R(x)$ by*

$$R(x) := \sum_{j=1}^{m} \big|\langle x, f_j\rangle\big|^2 f_j f_j^*. \tag{2.6}$$

Let $\lambda_{min}(R(x))$ denote the smallest eigenvalue of $R(x)$, and let $a_0 = \min_{\|x\|=1} \lambda_{\min}(R(x))$. Equivalently let a_0 be the largest constant so that $R(x) \geq a_0\|x\|^2 I$ for all $x \in H$, where I is the identity matrix.
Then $\mathcal{F}$ is phase retrievable on $\hat{H}$ if and only if $a_0 > 0$.
Additionally the constant a_0 introduced here is the same as the constant a_0 given by (2.4).

The results (1)–(3) are in [6], and (4)–(5) are in [4].

3. Information theoretic performance bounds

In this section we derive expressions for the *Fisher Information Matrix* and obtain performance bounds for reconstruction algorithms in the noisy case.

Consider the following noisy measurement process:

$$y_k = \big|\langle x, f_k\rangle\big|^2 + \nu_k, \quad \nu_k \sim \mathcal{N}\big(0, \sigma^2\big), \quad 1 \leq k \leq m \tag{3.1}$$

where the noise model is AWGN (additive white Gaussian noise): each random variable ν_k is independent and normally distributed with zero mean and σ^2 variance.

Consider the noiseless case first (that is $\nu_k = 0$). Obviously one cannot obtain the exact vector $x \in H$ due to the global sign ambiguity. Instead the best outcome is to identify (that is, to estimate) the class $\hat{x} = \{x, -x\}$ from $\alpha_{\mathcal{F}}(x)$. As such, we fix a disjoint partition of the punctured Hilbert space H, $\mathbb{R}^n \setminus \{0\} = \Omega_1 \cup \Omega_2$, such that $\Omega_2 = -\Omega_1$. We make the choice that the vector x belongs to Ω_1. Hence any estimator of x is a map $\omega : \mathbb{R}^m \longrightarrow \Omega_1 \cup \{0\}$. Denote by $\mathring{\Omega}_1$ its interior as a subset of $\mathbb{R}^n$. Such a decomposition is, for example

$$\Omega_1 = \bigcup_{k=1}^{n} \big\{x \in \mathbb{R}^n : \ x_k \geq 0, \ x_j = 0 \text{ for } j < k\big\}.$$

Note its interior is given by $\mathring{\Omega}_1 = \{x \in \mathbb{R}^n, \ x_1 > 0\}$.

Under these assumptions we compute the Fisher Information matrix (see [15]). This is given by

$$\big(\mathbb{I}(x)\big)_{k,j} = \mathbb{E}\big[\big(\nabla \log L(x)\big)\big(\nabla \log L(x)\big)^T\big] \tag{3.2}$$

where the likelihood function $L(x)$ is given by

$$L(x) = p(y|x) = \frac{1}{(2\pi)^{m/2}\sigma^m} \exp\left(-\frac{1}{2\sigma^2} \sum_{k=1}^{m} \big| y_k - \big|\langle x, f_k\rangle\big|^2 \big|^2 \right). \tag{3.3}$$

After some algebra (see [4]) we obtain

$$\mathbb{I}(x) = \frac{4}{\sigma^2} R(x), \qquad R(x) = \sum_{j=1}^{m} \big|\langle x, f_j\rangle\big|^2 f_j f_j^T. \tag{3.4}$$

Note the matrix $R(x)$ is exactly the same as the matrix introduced in (2.6). Thus we obtain the following results:

Theorem 3.1. *The frame $\mathcal{F}$ is phase retrievable if and only if the Fisher information matrix $\mathbb{I}(x)$ is invertible for any $x \neq 0$.*

When $\mathcal{F}$ is phase retrievable let a_0 be the positive constant introduced in (2.4). Then

$$\mathbb{I}(x) \geq \frac{4a_0}{\sigma^2} \|x\|^2 I \tag{3.5}$$

where I is the $n \times n$ identity operator.

This allows to state the following performance bound result (see [15] for details on the Cramer–Rao lower bound).

Theorem 3.2. *Assume $x \in \mathring{\Omega}_1$. Let $\omega : \mathbb{R}^m \to \Omega_1$ be any unbiased estimator for x. Then its covariance matrix is bounded below by the Cramer–Rao lower bound:*

$$\mathrm{Cov}\big[\omega(y)\big] \geq \big(\mathbb{I}(x)\big)^{-1} = \frac{\sigma^2}{4}\big(R(x)\big)^{-1}. \tag{3.6}$$

Furthermore, any efficient estimator (that is, any unbiased estimator ω that achieves the Cramer–Rao Lower Bound (3.6)) has the covariance matrix bounded from above by

$$\mathrm{Cov}\big[\omega(y)\big] \leq \frac{\sigma^2}{4a_0\|x\|^2} I \tag{3.7}$$

and Mean-Square error bounded above by

$$\mathrm{MSE}(\omega) = \mathbb{E}\big[\big\|\omega(y) - x\big\|^2\big] \leq \frac{n\sigma^2}{4a_0\|x\|^2}. \tag{3.8}$$

Remark 3.3. We point out the importance of the constant a_0 introduced in (2.4). On the one hand it represents a necessary and sufficient condition for phase retrievability as stated in Theorem 2.1. On the other hand the above results prove that a_0 provides also a bound for the Fisher Information matrix and hence a bound for any efficient estimator of $\hat{x}$. The larger this constant a_0, the smaller the variance of the efficient estimator. As we prove in the next section, the same constant a_0 represents the lower Lipschitz bound for the map $\alpha_{\mathcal{F}}{}^2$ (4.13) considered between $(\hat{H}, d_1)$ and the Euclidean space $(\mathbb{R}^m, \|\cdot\|)$ – see Theorem 4.5.

Additionally, similar expressions involving the bound a_0 occur in the complex case as well. Both the stochastic bound above and the bi-Lipschitz result in Theorem 4.5 can be extended to the complex case – see [5].

4. Robustness measures for reconstruction

In this section we analyze the robustness of deterministic phaseless reconstruction. Additionally we connect the constant a_0 introduced earlier in Theorem 2.1 and used in Theorem 3.1 to quantities directly computable from the frame $\mathcal{F}$.

Our approach is to analyze the stability in the worst case scenario, for which we consider the following measures. Denote $d(x,y) := \min(\|x-y\|, \|x+y\|)$. For any $x \in \mathbb{R}^n$ and $\varepsilon > 0$ define

$$Q_\varepsilon(x) = \max_{\{y:\|\alpha_{\mathcal{F}}(x)-\alpha_{\mathcal{F}}(y)\|\le\varepsilon\}} \frac{d(x,y)}{\varepsilon}. \tag{4.1}$$

The size of $Q_\varepsilon(x)$ measures the worst case stability of the reconstruction for the vector x, under the assumption that the total noise level is controlled by ε. We also study the global stability by analyzing the measures

$$q_\varepsilon := \max_{\|x\|=1} Q_\varepsilon(x), \qquad q_0 := \limsup_{\varepsilon\to 0} q_\varepsilon, \qquad q_\infty := \sup_{\varepsilon>0} q_\varepsilon. \tag{4.2}$$

Here $\|.\|$ denotes usual Euclidian norm. Note that $Q_\varepsilon(x)$ has the scaling property $Q_\varepsilon(x) = Q_{|c|\varepsilon}(cx)$ for any real $c \neq 0$. Thus it is natural to focus on unit vectors x.

We introduce now some quantities that play key roles in the estimation of these robustness measures. For the frame $\mathcal{F}$ let $F = [f_1, f_2, \cdots, f_m]$ be its frame matrix. Denote by $\mathcal{F}[S] = \{f_k,\ k \in S\}$ the subset of $\mathcal{F}$ indexed by a subset $S \subseteq \{1,2,\dots,m\}$, and by F_S the frame matrix corresponding to $\mathcal{F}[S]$ (which is the matrix with vectors in $\mathcal{F}[S]$ as its columns). Set

$$A[S] := \sigma_n^2(F_S) = \lambda_{\min}\big(F_S F_S^*\big), \tag{4.3}$$

where as usual σ_n and $\lambda_{\min}$ denote the n-th singular value and the minimal eigenvalue, respectively. Note that $A[S]$ is in fact the lower frame bound of $\mathcal{F}[S]$.

Let $\mathcal{S}$ denote the collection of subsets S of $\{1,2,\dots,m\}$ so that $\dim(\mathrm{span}(\mathcal{F}[S^c])) < n$, where $S^c = \{1,2,\dots,m\} \setminus S$ is the complement of S. In other words, $\mathrm{rank}(F_{S^c}) < n$. Denote by Δ and ω the following expressions:

$$\Delta = \min_S \sqrt{A[S] + A\big[S^c\big]} \tag{4.4}$$

$$\omega = \min_{S\in\mathcal{S}} \sigma_n(F_S). \tag{4.5}$$

All of them depend of course on $\mathcal{F}$. However since we fix $\mathcal{F}$ throughout the paper, we shall not explicitly reference $\mathcal{F}$ in the notation for simplicity as there will not be any confusion. Clearly

$$\Delta \le \omega. \tag{4.6}$$

Proposition 4.1. *Let $\varepsilon > 0$. Then the stability measurement function $Q_\varepsilon(x)$ is given by*

$$Q_\varepsilon(x) = \frac{1}{\varepsilon} \max_{(w_1,w_2)\in\Gamma} \min\big\{\|w_1\|, \|w_2\|\big\} \tag{4.7}$$

where the constraint set Υ is given by

$$\Upsilon = \left\{ (w_1, w_2) \mid \frac{1}{2}(w_1 + w_2) = x,\ \sum_{j=1}^{m} \min\big(|\langle f_j, w_1\rangle|^2, |\langle f_j, w_2\rangle|^2\big) = \|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2 \le \varepsilon^2 \right\}, \tag{4.8}$$

where $S := S(w_1, w_2) = \{j :\ |\langle f_j, w_1\rangle| \le |\langle f_j, w_2\rangle|\}$.

Proof. For any $x, y \in \mathbb{R}^n$ let $w_1 = x + y$ and $w_2 = x - y$. Then $x = \frac{1}{2}(w_1 + w_2)$ and $y = \frac{1}{2}(w_1 - w_2)$. It is easy to check that for $S = \{j :\ |\langle f_j, w_1\rangle| \le |\langle f_j, w_2\rangle|\}$ we have

$$\big|\langle f_j, x\rangle\big| - \big|\langle f_j, y\rangle\big| = \begin{cases} \pm\langle f_j, w_1\rangle & j \in S, \\ \pm\langle f_j, w_2\rangle & j \in S^c. \end{cases}$$

In other words,

$$\big|\langle f_j, x\rangle\big| - \big|\langle f_j, y\rangle\big| = \min\big(|\langle f_j, w_1\rangle|, |\langle f_j, w_2\rangle|\big). \tag{4.9}$$

Let F be the frame matrix of $\mathcal{F}$. We thus have

$$\big\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\big\|^2 = \sum_{j\in S} \big|\langle f_j, w_1\rangle\big|^2 + \sum_{j\in S^c} \big|\langle f_j, w_2\rangle\big|^2 = \|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2.$$

Note that $d(x, y) = \min(\|w_1\|, \|w_2\|)$. The proposition now follows. □

The above proposition allows us to establish the following stability result for the worst case scenario.

Theorem 4.2. *Assume that the frame $\mathcal{F}$ is phase retrievable. Let $A > 0$ be the lower frame bound for the frame $\mathcal{F}$ and let $\tau := \min\{\sigma_n(F_S) :\ S \subseteq \{1, \dots, m\},\ \operatorname{rank}(F_S) = n\}$.*

(A) *For any $\varepsilon > 0$ we have*

$$\min\left\{\frac{1}{\varepsilon}, \frac{1}{\omega}\right\} \le q_\varepsilon \le \frac{1}{\Delta}. \tag{4.10}$$

(B) *If $\varepsilon < \tau$ then $q_\varepsilon = \frac{1}{\omega}$. Consequently $q_0 = \frac{1}{\omega}$.*

(C) *For any nonzero $x \in \mathbb{R}^n$ and any $0 < \varepsilon < \Delta_x$ we have*

$$Q_\varepsilon(x) = \frac{1}{\sqrt{A}}, \tag{4.11}$$

where

$$\Delta_x := \frac{2\tau}{\max(\|f_j\|) + \tau} \min\big\{\big|\langle f_j, x\rangle\big| :\ \langle f_j, x\rangle \ne 0\big\}.$$

(D) *The upper bound q_∞ equals the reciprocal of Δ:*

$$q_\infty = \frac{1}{\Delta}. \tag{4.12}$$

Proof. To prove (A) we first establish the upper bound in (4.10). Let $x \in \mathbb{R}^n$. By Proposition 4.1 we have

$$Q_\varepsilon(x) = \frac{1}{\varepsilon} \max_{w_1,w_2} \min\{\|w_1\|, \|w_2\|\}$$

under the constraints $\frac{1}{2}(w_1 + w_2) = x$ and

$$\left\|F_S^* w_1\right\|^2 + \left\|F_{S^c}^* w_2\right\|^2 \le \varepsilon^2$$

for some S. Now assume without loss of generality that $\|w_1\| \le \|w_2\|$. Then

$$\begin{aligned}\frac{\varepsilon^2}{\|w_1\|^2} &\ge \frac{\|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2}{\|w_1\|^2} \\ &\ge \sigma_n^2(F_S) + \sigma_n^2(F_{S^c})\frac{\|w_2\|^2}{\|w_1\|^2} \\ &\ge \Delta.\end{aligned}$$

It follows that

$$\frac{1}{\varepsilon}\min\{\|w_1\|, \|w_2\|\} \le \frac{1}{\Delta}.$$

Thus $Q_\varepsilon(x) \le \frac{1}{\Delta}$.

To establish the lower bound in (4.10) we construct for any $\varepsilon > 0$ an $x \in \mathbb{R}^n$ and vectors w_1, w_2 satisfying the imposed constraints. Let S be a subset of $\{1, 2, \dots, m\}$ such that $\operatorname{rank}(F_{S^c}) < n$ and $\sigma_n(F_S) = \omega$. Choose $v_1, v_2 \in \mathbb{R}^n$ with the property $\|v_1\| = \|v_2\| = 1$ and

$$\left\|F_S^* v_1\right\| = \omega, \qquad F_{S^c}^* v_2 = 0.$$

Set

$$t = \min\left\{\frac{\varepsilon}{\omega}, 1\right\}, \quad \text{and} \quad w_1 = t v_1.$$

Hence $\|w_1\| = t \le 1$. Now we select an $s \in \mathbb{R}$ so that $\|w_1 + s v_2\| = 2$. This is always possible since $s \mapsto \|w_1 + s v_2\|$ is continuous and $\|w_1 + 0 v_2\| = t \le 1 \le 2 \le \|w_1 + 3 v_2\|$. Set $w_2 = s v_2$ so $\|w_1 + w_2\| = 2$. We have

$$|s| = \|s v_2\| \ge \|w_1 + s v_2\| - \|w_1\| = 2 - t \ge 1.$$

Thus $\|w_2\| \ge \|w_1\|$. Now let

$$x = \frac{1}{2}(w_1 + w_2) \quad \text{and} \quad y = \frac{1}{2}(w_1 - w_2).$$

We have then

$$\begin{aligned}\left\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\right\|^2 &= \sum_{j=1}^m \min\left(\left|\langle f_j, w_1\rangle\right|^2, \left|\langle f_j, w_2\rangle\right|^2\right) \\ &\le \sum_{j\in S}\left|\langle f_j, w_1\rangle\right|^2 + \sum_{j\in S^c}\left|\langle f_j, w_2\rangle\right|^2 \\ &= t^2\omega^2 \le \varepsilon^2.\end{aligned}$$

Furthermore

$$d(x,y) = \min(\|w_1\|, \|w_2\|) = \|w_1\| = t.$$

Hence for this x we have

$$Q_\varepsilon(x) \geq \frac{d(x,y)}{\varepsilon} = \min\left\{\frac{1}{\varepsilon}, \frac{1}{\omega}\right\}.$$

It follows that $q_\varepsilon \geq \min\{\frac{1}{\varepsilon}, \frac{1}{\omega}\}$. Now by taking $\varepsilon > 0$ sufficiently small we have $q_\varepsilon \geq \frac{1}{\omega}$.

We now prove (B). Assume that $\varepsilon \leq \min\{\sigma_n(F_S) : \ \operatorname{rank}(F_S) = n\}$. Then clearly we have $\varepsilon \leq \omega$. Thus by (4.10) we have $q_\varepsilon \geq \frac{1}{\omega}$. Again for each $x \in \mathbb{R}^n$ with $\|x\| = 1$ we consider w_1, w_2 for the estimation of $q_\varepsilon(x)$. The constraint $\|w_1 + w_2\| = 2$ implies either $\|w_1\| \geq 1$ or $\|w_2\| \geq 1$. Without loss of generality we assume that $\|w_1\| \geq 1$. For the constraint $\|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2 \leq \varepsilon^2$ for some S, assume that $\operatorname{rank}(F_S) = n$ then we have

$$\|F_S^* w_1\| \geq \sigma_n(F_S)\|w_1\| \geq \min\{\sigma_n(F_S):\ \operatorname{rank}(F_S) = n\} > \varepsilon.$$

This is a contradiction. So $\operatorname{rank}(F_S) < n$ and hence

$$\varepsilon^2 \geq \|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2 \geq \|F_{S^c}^* w_2\|^2 \geq \omega^2 \|w_2\|^2.$$

Thus $\|w_2\| \leq \frac{\varepsilon}{\omega}$. Proposition 4.1 now yields $q_\varepsilon = \frac{1}{\omega}$, proving part (B).

Now we prove (C). We go back to the formulation in Proposition 4.1.

$$Q_\varepsilon(x) = \frac{1}{\varepsilon} \max_{w_1, w_2} \min\{\|w_1\|, \|w_2\|\}$$

under the constraints $\frac{1}{2}(w_1 + w_2) = x$ and

$$\|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2 \leq \varepsilon^2$$

where $S := S(w_1, w_2) = \{j:\ |\langle f_j, w_1\rangle| \leq |\langle f_j, w_2\rangle|\}$. Since $\alpha_{\mathcal{F}}$ is injective, either $\operatorname{rank}(F_S) = n$ or $\operatorname{rank}(F_{S^c}) = n$ by Theorem 2.1 (1). Without loss of generality we assume $\operatorname{rank}(F_S) = n$. Thus $\varepsilon \geq \|F_S^* w_1\| \geq \tau\|w_1\|$. So $\|w_1\| \leq \varepsilon/\tau$. We show that for any $k \in S^c$ we must have $\langle f_k, x\rangle = 0$. Assume otherwise and write $w_2 = 2x - w_1$, $L_x := \min\{|\langle f_j, x\rangle| :\ \langle f_j, x\rangle \neq 0\}$. Then

$$|\langle f_k, w_2\rangle| \geq 2|\langle f_k, x\rangle| - |\langle f_k, w_1\rangle| \geq 2L_x - \max(\|f_j\|)\|w_1\| \geq 2L_x - \max(\|f_j\|)\frac{\varepsilon}{\tau} > \varepsilon.$$

This is a contradiction. Thus for $k \in S^c$ we have $\langle f_k, x\rangle = 0$ and

$$|\langle f_j, w_2\rangle| = |\langle f_j, 2x - w_1\rangle| = |\langle f_j, w_1\rangle|.$$

It follows that

$$\|F_S^* w_1\|^2 + \|F_{S^c}^* w_2\|^2 = \|F^* w_1\|^2 \leq \varepsilon^2.$$

Thus $\|w_1\| \leq \varepsilon/\sqrt{A}$ and hence $Q_\varepsilon(x) \leq \frac{1}{\sqrt{A}}$. Now we show the bound can be achieved. Let w_1 satisfy $\|F^* w_1\| = \sqrt{A}\|w_1\| = \varepsilon$. Such a w_1 always exists. Then clearly w_1 and $w_2 = 2x - w_1$ satisfy the required constraints, and it is easy to check that $\min(\|w_1\|, \|w_2\|) = \|w_1\| = \varepsilon/\sqrt{A}$.

Finally we prove (D). By the result at part (A), $q_\infty \leq \frac{1}{\Delta}$. It is therefore sufficient to show that $Q_\varepsilon(x) \geq \frac{1}{\Delta}$ for some x and ε. Let S_0 be the subset that achieves the minimum in (4.4). Let $u, v \in H$ be unit eigenvectors corresponding to the lowest eigenvalues of $F_{S_0}F_{S_0}^*$ and $F_{S_0^c}F_{S_0^c}^*$ respectively. Thus

$$\left\|F_{S_0}^* u\right\|^2 = A[S_0], \qquad \left\|F_{S_0^c}^* v\right\|^2 = A\left[S_0^c\right]$$

Let $x = (u+v)/2$ and $\varepsilon = \Delta$, and set $w_1 = u$, $w_2 = v$. Then by Proposition 4.1

$$Q_\varepsilon(x) \geq \frac{\min(\|w_1\|, \|w_2\|)}{\varepsilon} = \frac{1}{\Delta}$$

since

$$\sum_{j=1}^m \min\left(\left|\langle f_j, w_1\rangle\right|^2, \left|\langle f_j, w_2\rangle\right|^2\right) \leq \left\|F_{S_0}^* w_1\right\|^2 + \left\|F_{S_0^c}^* w_2\right\|^2 = \varepsilon^2$$

This concludes the proof. □

Remark. It may seem strange that $Q_\varepsilon(x) = \frac{1}{\sqrt{A}}$ for all $x \neq 0$ and sufficiently small ε while $q_0 = \frac{1}{\omega}$, where ω is typically much smaller than $\sqrt{A}$. The reason is that for $Q_\varepsilon(x) = \frac{1}{\sqrt{A}}$ to hold, ε depends on x. Thus we cannot exchange the order of $\limsup_{\varepsilon\to 0}$ and $\max_{\|x\|=1}$.

Related to the study of stability of phaseless reconstruction is the study of the Lipschitz property of the map $\alpha_{\mathcal{F}}$ on $\hat{H} := \mathbb{R}^n/\sim$. We analyze the bi-Lipschitz bounds of both $\alpha_{\mathcal{F}}$ and ${\alpha_{\mathcal{F}}}^2$, which is simply the map $\alpha_{\mathcal{F}}$ with all entries squared, i.e.

$${\alpha_{\mathcal{F}}}^2(x) := \left[\left|\langle f_j, x\rangle\right|^2, \ldots, \left|\langle f_m, x\rangle\right|^2\right]^T. \tag{4.13}$$

We shall consider two distance functions on $\hat{H} = \mathbb{R}^n/\sim$: the standard distance $d(x,y) := \min(\|x-y\|, \|x+y\|)$ and the distance $d_1(x,y) := \|xx^* - yy^*\|_1$ where $\|X\|_1$ denotes the *nuclear norm* of X, which is the sum of all singular values of X. Specifically we are interested in examining the local and global behavior of the following ratios

$$U(x,y) := \frac{\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\|}{d(x,y)}, \qquad V(x,y) := \frac{\|{\alpha_{\mathcal{F}}}^2(x) - {\alpha_{\mathcal{F}}}^2(y)\|}{d_1(x,y)}. \tag{4.14}$$

While all norms in finite dimensional spaces are equivalent, we choose to consider d_1, the nuclear norm induced distance on $\hat{H}$, because the Lipschitz lower and upper bounds are very much related to the matrix $R(x)$ introduced in Theorem 2.1.

We first investigate the bounds for $U(x,y)$. For this the upper bound is relatively straightforward. Let $w_1 = x - y$ and $w_2 = x + y$. We have already shown in the proof of Theorem 4.2 using (4.9) that

$$\begin{aligned}
\left\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\right\|^2 &= \sum_{j=1}^m \min\left(\left|\langle f_j, w_1\rangle\right|^2, \left|\langle f_j, w_2\rangle\right|^2\right) \\
&\leq \min\left\{\sum_{j=1}^m \left|\langle f_j, w_1\rangle\right|^2, \sum_{j=1}^m \left|\langle f_j, w_2\rangle\right|^2\right\} \\
&\leq B\min\left\{\|w_1\|^2, \|w_2\|^2\right\} = Bd^2(x,y),
\end{aligned}$$

where B is the upper frame bound of the frame $\mathcal{F}$. Thus $U(x,y)$ has an upper bound $U(x,y) \leq \sqrt{B}$. Furthermore, the bound is sharp. To see this, pick a unit vector $x \in \mathbb{R}^n$ such that $\sum_{j=1}^m |\langle f_j, w_1\rangle|^2 = B$ and set $y = 2x$. Then $U(x,y) = \sqrt{B}$.

To study the lower bound $U(x,y)$ we now consider the following quantities:

$$\begin{aligned}
\rho_\varepsilon(x) &:= \inf_{\{y: d(x,y)\leq \varepsilon\}} U(x,y),\\
\rho(x) &:= \liminf_{\{y: d(x,y)\to 0\}} U(x,y) = \liminf_{\varepsilon\to 0} \rho_\varepsilon(x),\\
\rho_0 &:= \inf_x \rho(x),\\
\rho_\infty &:= \inf_{d(x,y)>0} U(x,y).
\end{aligned}$$

We apply the equality

$$U^2(x,y) = \frac{\sum_{j=1}^m \min(|\langle f_j, w_1\rangle|^2, |\langle f_j, w_2\rangle|^2)}{\min(\|w_1\|^2, \|w_2\|^2)}$$

where again $w_1 = x - y$ and $w_2 = x + y$. Now fix x and let $d(x,y) < \varepsilon$. Without loss of generality we may assume $\|y - x\| < \varepsilon$. Thus $\|w_1\| < \varepsilon$ and $\|w_2 - 2x\| = \|w_1\| < \varepsilon$. Let $S = \{j,\ \langle f_j, x\rangle \neq 0\}$ and set

$$\varepsilon_0(x) := \frac{\min_{k\in S} |\langle f_k, x\rangle|}{\max_{k\in S} \|f_k\|}. \tag{4.15}$$

Note for any w_1 with $\|w_1\| < \varepsilon_0$ and $k \in S$ we have

$$\big|\langle f_k, w_2\rangle\big| = \big|2\langle f_k, x\rangle - \langle f_k, w_1\rangle\big| \geq 2\big|\langle f_k, x\rangle\big| - \big|\langle f_k, w_1\rangle\big| \geq 2\varepsilon_0(x)\|f_k\| - \|w_1\|\|f_k\| \geq \big|\langle f_k, w_1\rangle\big|,$$

whereas for $k \in S^c$ we have

$$\big|\langle f_k, w_2\rangle\big| = \big|\langle f_k, w_1\rangle\big|.$$

Hence $\min(|\langle f_j, w_1\rangle|^2, |\langle f_j, w_2\rangle|^2) = |\langle f_j, w_1\rangle|^2$ for all j whenever $\varepsilon < \varepsilon_0(x)$. It follows that

$$U^2(x,y) = \frac{\sum_{j=1}^m |\langle f_j, w_1\rangle|^2}{\|w_1\|^2} = \sum_{j=1}^m \left|\left\langle \frac{w_1}{\|w_1\|}, f_j\right\rangle\right|^2.$$

Thus $U^2(x,y) \geq A$ where A is the lower frame bound for the frame $\mathcal{F}$. Furthermore this lower bound is achieved whenever $w_1 = x - y$ is an eigenvector corresponding to the smallest eigenvalue of FF^*. This implies that

$$\rho_\varepsilon(x) = \sqrt{A}$$

whenever $\varepsilon < \varepsilon_0(x)$. Consequently $\rho(x) = \sqrt{A}$. We have the following theorem:

Theorem 4.3. *Assume that the frame $\mathcal{F}$ is phase retrievable. Let A, B be the lower and upper frame bounds for the frame $\mathcal{F}$, respectively and for each $x \in \mathbb{R}^n$, let $\varepsilon_0(x)$ be given in (4.15). Then*

(1) *$U(x,y) \leq \sqrt{B}$ for any $x, y \in \mathbb{R}^n$ with $d(x,y) > 0$.*
(2) *Assume that $\varepsilon < \varepsilon_0(x)$. Then $\rho_\varepsilon(x) = \sqrt{A}$. Consequently $\rho(x) = \rho_0 = \sqrt{A}$.*

(3) $\Delta = \rho_\infty \leq \omega \leq \rho_0 = \rho(x) = \sqrt{A}$.
(4) *The map $\alpha_{\mathcal{F}}$ is bi-Lipschitz with (optimal) upper Lipschitz bound $\sqrt{B}$ and lower Lipschitz bound ρ_∞:*

$$\rho_\infty d(x,y) \leq \big\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\big\| \leq \sqrt{B} d(x,y), \quad \forall x, y \in \hat{H}$$

Proof. We have already proved (1) and (2) of the theorem in the discussion. It remains only to prove (3) since (4) is just a restatement of (1) and (3). Note that

$$\rho_\infty^2 = \inf_{d(x,y)>0} U^2(x,y) = \inf_{w_1, w_2 \neq 0} \frac{\sum_{j=1}^m \min(|\langle f_j, w_1\rangle|^2, |\langle f_j, w_2\rangle|^2)}{\min(\|w_1\|^2, \|w_2\|^2)}.$$

For any w_1, w_2, assume without loss of generality that $0 < \|w_1\| \leq \|w_2\|$. Let $S = \{j : |\langle f_j, w_1\rangle| \leq |\langle f_j, w_2\rangle|\}$. Set $v_1 = w_1/\|w_1\|$, $v_2 = w_2/\|w_2\|$ and $t = \|w_2\|/\|w_1\| \geq 1$. Then

$$\begin{aligned}
\frac{\sum_{j=1}^m \min(|\langle f_j, w_1\rangle|^2, |\langle f_j, w_2\rangle|^2)}{\min(\|w_1\|^2, \|w_2\|^2)} &= \sum_{j\in S} \big|\langle f_j, v_1\rangle\big|^2 + t^2 \sum_{j\in S^c} \big|\langle f_j, v_2\rangle\big|^2 \\
&\geq \sum_{j\in S} \big|\langle f_j, v_1\rangle\big|^2 + \sum_{j\in S^c} \big|\langle f_j, v_2\rangle\big|^2 \\
&\geq \Delta^2.
\end{aligned}$$

Hence $\rho_\infty \geq \Delta$.

Let S and $u, v \in H$ be normalized (eigen) vectors that achieve the bound Δ, that is:

$$\|u\| = \|v\| = 1, \qquad \sum_{k\in S} \big|\langle u, f_k\rangle\big|^2 + \sum_{k\in S^c} \big|\langle v, f_k\rangle\big|^2 = \Delta^2.$$

Set $x = u + v$ and $y = u - v$. Then, following [9]

$$\begin{aligned}
\big\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\big\|^2 &= \sum_{k\in S} \big|\big|\langle u+v, f_k\rangle\big| - \big|\langle u-v, f_k\rangle\big|\big|^2 + \sum_{k\in S^c} \big|\big|\langle u+v, f_k\rangle\big| - \big|\langle u-v, f_k\rangle\big|\big|^2 \\
&\leq 4\Big(\sum_{k\in S} \big|\langle u, f_k\rangle\big|^2 + \sum_{k\in S^c} \big|\langle v, f_k\rangle\big|^2\Big) = 4\Delta^2.
\end{aligned}$$

On the other hand

$$d(x,y) = \min\big(\|x-y\|, \|x+y\|\big) = 2.$$

Thus we obtain

$$\frac{\|\alpha_{\mathcal{F}}(x) - \alpha_{\mathcal{F}}(y)\|}{d(x,y)} \leq \Delta.$$

The theorem is now proved. □

Remark. The two quantities, ρ_∞ and q_∞ satisfy $\rho_\infty = \frac{1}{q_\infty}$. However there are subtle differences between $Q_\varepsilon(x)$ and $\rho_\varepsilon(x)$ so that the simple relationship $\rho_\varepsilon(x) = 1/Q_\varepsilon(x)$ does not usually hold. One such difference is due to the significance of ε for the two bounds. See the numerical example presented in the last section.

Remark. The upper Lipschitz bound $\sqrt{B}$ has been obtained independently in [9]. The lower Lipschitz bound we obtained here strenghtens the estimates given in [9]. Specifically their estimate for ρ_∞ reads $\boldsymbol{\sigma} \leq \rho_\infty \leq \sqrt{2}\boldsymbol{\sigma}$ where

$$\boldsymbol{\sigma} = \min_S \max\big(\sigma_n(F_S), \sigma_n(F_{S^c})\big) \tag{4.16}$$

Clearly $\boldsymbol{\sigma} \leq \Delta \leq \sqrt{2}\boldsymbol{\sigma}$.

We conclude this section by turning our attention to the analysis of $V(x,y)$. A motivation for studying it is that in practical problems the noise is often added directly to ${\alpha_{\mathcal{F}}}^2$ as in (3.1) rather than to $\alpha_{\mathcal{F}}$. Such noise model is used in many studies of phaseless reconstruction, e.g. in the Phaselift algorithm [10], or in the IRLS algorithm in [4].

Let $\mathrm{Sym}_n(\mathbb{R})$ denote the set of $n \times n$ symmetric matrices over $\mathbb{R}$. It is a Hilbert space with the standard inner product given by $\langle X, Y\rangle := \mathrm{tr}(XY^T) = \mathrm{tr}(XY)$. The nonlinear map ${\alpha_{\mathcal{F}}}^2$ actually induces a linear map on $\mathrm{Sym}_n(\mathbb{R})$. Write $X = xx^T$ for any $x \in \mathbb{R}^n$. Then the entries of ${\alpha_{\mathcal{F}}}^2(x)$ are

$$\big({\alpha_{\mathcal{F}}}^2(x)\big)_j = \big|\langle f_j, x\rangle\big|^2 = x^T f_j f_j^T x = \mathrm{tr}(F_j X) = \langle F_j, X\rangle, \tag{4.17}$$

where $F_j := f_j f_j^T$. Now we denote by $\mathcal{A}$ the linear operator $\mathcal{A}:\ \mathrm{Sym}_n(\mathbb{R}) \longrightarrow \mathbb{R}^m$ with entries

$$\big(\mathcal{A}(X)\big)_j = \langle F_j, X\rangle = \mathrm{tr}(F_j X).$$

Let $S_n^{p,q}$ be the set of $n \times n$ real symmetric matrices that have at most p positive and q negative eigenvalues. Thus $S_n^{1,0}$ denotes the set of $n \times n$ real symmetric non-negative definite matrices of rank at most one. Note that spectral decomposition easily shows that $X \in S_n^{1,0}$ if and only if $X = xx^T$ for some $x \in \mathbb{R}^n$.

The following lemma will be useful in this analysis

Lemma 4.4. *The following are equivalent.*

(A) $X \in S_n^{1,1}$.
(B) $X = xx^T - yy^T$ *for some* $x, y \in \mathbb{R}^n$.
(C) $X = \frac{1}{2}(w_1 w_2^T + w_2 w_1^T)$ *for some* $w_1, w_2 \in \mathbb{R}^n$.

Furthermore, for $X = \frac{1}{2}(w_1 w_2^T + w_2 w_1^T)$ *its nuclear norm is* $\|X\|_1 = \|w_1\|\|w_2\|$.

Proof. (A) $\Rightarrow$ (B) is a direct result of spectral decomposition, which yields $X = \beta_1 u_1 u_1^T - \beta_2 u_2 u_2^T$ for some $u_1, u_2 \in \mathbb{R}^n$ and $\beta_1, \beta_2 \geq 0$. Thus $X = xx^T - yy^T$ where $x := \sqrt{\beta_1}u_1$ and $y := \sqrt{\beta_2}u_2$.

(B) $\Rightarrow$ (C) is proved directly by setting $w_1 = x - y$ and $w_2 = x + y$.

We now prove (C) $\Rightarrow$ (A) by computing the eigenvalues of $X = \frac{1}{2}(w_1 w_2^T + w_2 w_1^T)$. Obviously $\mathrm{rank}(X) \leq 2$. Let λ_1, λ_2 be the two (possibly) nonzero eigenvalues of X. Then

$$\lambda_1 + \lambda_2 = \mathrm{tr}\{X\} = \langle w_1, w_2\rangle,$$
$$\lambda_1^2 + \lambda_2^2 = \mathrm{tr}\{X^2\} = \big(\|w_1\|^2\|w_2\|^2 + \big|\langle w_1, w_2\rangle\big|^2\big)/2.$$

Solving for eigenvalues we obtain

$$\lambda_1 = \frac{1}{2}\big(\langle w_1, w_2\rangle + \|w_1\|\|w_2\|\big),$$
$$\lambda_2 = \frac{1}{2}\big(\langle w_1, w_2\rangle - \|w_1\|\|w_2\|\big).$$

Hence, by Cauchy–Schwarz inequality, $\lambda_1 \geq 0 \geq \lambda_2$ which proves $X \in S_n^{1,1}$. Furthermore, it also shows that the nuclear norm of X is $\|X\|_1 = |\lambda_1| + |\lambda_2| = \|w_1\|\|w_2\|$. $\square$

Now we analyze $V(x,y)$. Parallel to the study of $U(x,y)$ we consider the following quantities:

$$\begin{aligned}
\mu_\varepsilon(x) &:= \inf_{\{y:d(x,y)\leq\varepsilon\}} V(x,y),\\
\mu(x) &:= \liminf_{\{y:d(x,y)\to 0\}} V(x,y) = \liminf_{\varepsilon\to 0} \mu_\varepsilon(x),\\
\mu_0 &:= \inf_x \mu(x),\\
\mu_\infty &:= \inf_{d(x,y)>0} V(x,y),
\end{aligned}$$

as well as the upper bound $\sup_{d_1(x,y)>0} V(x,y)$. By (4.17) we have $|\langle f_j, x\rangle|^2 - |\langle f_j, y\rangle|^2 = \langle F_j, X\rangle$ where $F_j = f_j f_j^T$ and $X = xx^T - yy^T$. Hence

$$V^2(x,y) = \frac{\sum_{j=1}^m |\langle F_j, X\rangle|^2}{\|X\|_1^2}.$$

Set $w_1 = x - y$ and $w_2 = x + y$ and apply Lemma 4.4 we obtain

$$V^2(x,y) = \frac{\sum_{j=1}^m |\langle f_j, w_1\rangle|^2 |\langle f_j, w_2\rangle|^2}{\|w_1\|^2\|w_2\|^2}. \tag{4.18}$$

We can immediately obtain the upper bound:

$$V(x,y) \leq \left(\sup_{\|e_1\|=1,\|e_2\|=1} \sum_{j=1}^m \big|\langle f_j, e_1\rangle\big|^2 \big|\langle f_j, e_2\rangle\big|^2\right)^{1/2} = \left(\max_{\|e\|=1} \sum_{j=1}^m \big|\langle f_j, e\rangle\big|^4\right)^{1/2} =: \Lambda_{\mathcal{F}}{}^2$$

where $\Lambda_{\mathcal{F}}$ denotes the operator norm of the linear analysis operator $T : H \to \mathbb{R}^m$, $T(x) = (\langle x, f_k\rangle)_{k=1}^m$ defined between the Euclidian space $H = \mathbb{R}^n$ and the Banach space $\mathbb{R}^m$ endowed with the l^4-norm:

$$\Lambda_{\mathcal{F}} = \left(\max_{\|x\|=1} \sum_{k=1}^m \big|\langle x, f_k\rangle\big|^4\right)^{1/4} \tag{4.19}$$

Note also that

$$\Lambda_{\mathcal{F}}{}^2 = \max_{\|x\|=1} \lambda_{\max}\big(R(x)\big)$$

where $R(x)$ was defined in (2.6). An immediate bound is $\Lambda_{\mathcal{F}} \leq \sqrt{B}\max\|f_k\|$ with B the upper frame bound of $\mathcal{F}$.

Fix $x \neq 0$ and let $d(x,y) \to 0$. Then either $y \to x$ or $y \to -x$. Without loss of generality we assume that $x \to y$. Thus $w_1 = x - y \to 0$ and $w_2 = x + y \to 2x$. However $w_1/\|w_1\|$ can be any unit vector. Thus

$$\mu^2(x) = \frac{1}{\|x\|^2} \inf_{\|u\|=1} \sum_{j=1}^m \big|\langle f_j, x\rangle\big|^2 \big|\langle f_j, u\rangle\big|^2 = \frac{1}{\|x\|^2} \inf_{\|u\|=1} \big\langle R(x)u, u\big\rangle = \frac{1}{\|x\|^2}\lambda_{\min}\big(R(x)\big)$$

where $R(x)$ was introduced in (2.6). Thus we obtain

$$\mu^2(x) = \frac{1}{\|x\|^2}\lambda_{\min}\big(R(x)\big), \qquad \mu_0^2 = \min_{\|u\|=1} \lambda_{\min}\big(R(u)\big).$$

On the other hand note

$$\inf_{d(x,y)>0} V^2(x,y) = \inf_{w_1,w_2\neq 0} \frac{\sum_{j=1}^m |\langle f_j, w_1\rangle|^2 |\langle f_j, w_2\rangle|^2}{\|w_1\|^2\|w_2\|^2} = \min_{\|u\|=1} \lambda_{\min}\big(R(u)\big) = a_0^2,$$

where a_0 was introduced in (2.4). Thus we proved:

Theorem 4.5. *Assume the frame $\mathcal{F}$ is phase retrievable. Then*

$$\mu(x) = \frac{1}{\|x\|}\sqrt{\lambda_{\min}\big(R(x)\big)}, \tag{4.20}$$

$$\mu_\infty = \mu_0 = \min_{u:\|u\|=1} \sqrt{\lambda_{\min}\big(R(u)\big)} = \sqrt{a_0}. \tag{4.21}$$

Furthermore ${\alpha_{\mathcal{F}}}^2$ is bi-Lipschitz with upper Lipschitz bound ${\Lambda_{\mathcal{F}}}^2$ and lower Lipschitz bound $\sqrt{a_0}$:

$$\sqrt{a_0}d_1(x,y) \leq \big\|{\alpha_{\mathcal{F}}}^2(x) - {\alpha_{\mathcal{F}}}^2(y)\big\| \leq {\Lambda_{\mathcal{F}}}^2 d_1(x,y)$$

where a_0 is the same positive constant used in Theorems 2.1 and 3.1, and $\Lambda_{\mathcal{F}}$ is the norm of the analysis operator defined between the Euclidian space H and $l^4(\{1,2,\dots,m\})$.

Remark. Note that the distance $d(.,.)$ is not equivalent to $d_1(.,.)$. Theorem 4.5 now also implies that ${\alpha_{\mathcal{F}}}^2$ is not bi-Lipschitz with respect to the distance $d(.,.)$ on $\hat{H}$. This fact was pointed out in [9].

5. Robustness and size of redundancy

Previous sections establish results on the robustness of phaseless reconstruction for the worst case scenario. A natural question is to ask: can "reasonable" robustness be achieved for a given frame, and in particular with small number of samples? We shall examine how q_∞ scales as the dimension n increases.

Consider the case where $m = 2n-1$. This is the minimal redundancy required for phaseless reconstruction. In this case any frame $\mathcal{F}$ would have $\Delta = \omega$. Hence we have $\min\{1/\omega, 1/\varepsilon\} \leq q_\varepsilon = 1/\omega$. The stability of the reconstruction is thus mostly controlled by the size of $1/\omega$. The question is: how big is ω, especially as n increases?

Assume that the frame elements of $\mathcal{F}$ are all bounded by L, $\|f_j\| \leq L$ for all $f_j \in \mathcal{F}$. Consider the $n+1$ elements $\{f_j : j = 1,\dots,n+1\}$. They are linearly dependent so we can find $c_j \in \mathbb{R}$ such that $\sum_{j=1}^{n+1} c_j f_j = 0$. Without loss of generality we may assume $|c_{n+1}| = \min\{|c_j|\}$. Set $v = [c_1, c_2, \dots, c_n]^T$. Let $G = [f_1, \dots, f_n]$. Then $Gv = \sum_{j=1}^n c_j f_j = -c_{n+1} f_{n+1}$. Now all $|c_j| \geq |c_{n+1}|$ so $\|v\| \geq \sqrt{n}|c_{n+1}|$. Thus

$$\|Gv\| = |c_{n+1}|\|f_{n+1}\| \leq \frac{L}{\sqrt{n}}\|v\|.$$

It follows that $\sigma_n(G) \leq \frac{L}{\sqrt{n}}$, and hence

$$\omega \leq \frac{L}{\sqrt{n}}. \tag{5.1}$$

Note that here we have considered only the first $n+1$ vectors of the frame $\mathcal{F}$. The actual value of ω will likely decay much faster as n increases. In a preliminary work we are able to establish the bound $\omega \leq CL/\sqrt{n^3}$ where C is independent of n [18]. But even this estimate is likely far from optimal.

Conjecture 5.1. *Let $m = 2n-1$ and $\|f_j\| \leq L$ for all $f_j \in \mathcal{F}$. Then there exist constants $C > 0$ and $0 < \beta < 1$ independent of n such that*

$$\omega \leq CL\beta^n.$$

A related problem is as follows: Consider an $n \times (n+k)$ matrix $F = [g_1, g_2, \ldots, g_{n+k}]$. Let $\tau = \min\{\sigma_n(F_S) : \ S \subset \{1, \ldots, n+k\}, |S| = n\}$. Assume that all $\|g_j\| \leq 1$. How large can τ be? For $k = 1$ we have already seen that it is bounded from above by $C/\sqrt{n}$. The preliminary work [18] shows that for $k = 1$ it is bounded from above by $C/n^{\frac{3}{2}}$.

Conjecture 5.2. *There exists a constant $C = C(k, n)$ such that*

$$\tau \leq \frac{C}{n^{k-\frac{1}{2}}},$$

where $C(k, n) = O_k(\log^{q_k} n)$ for some $q_k > 0$. Here O_k denotes the dependence on k.

Thus in the minimal setting with $m = 2n-1$ it is impossible to achieve scale independent stability for phaseless reconstruction. The same arguments can be used to show that even when $m = 2n + k_0$ for some fixed k_0 scale independent stability is not possible. A natural question is whether scale independent stability is possible when we increase the redundancy of the frame. As it turns out this is possible via a recent work by Wang [17]. More precisely, the following result follows from the main results in [17]:

Theorem 5.3. *Let $r_0 > 2$ and let $F = \frac{1}{\sqrt{n}} G$ where G is an $n \times m$ random matrix whose elements are i.i.d. normal $N(0, 1)$ random variables such that $m/n = r_0$. Then there exist constants $0 < \Delta_0 \leq \omega_0$ dependent only on r_0 and not on n such that with high probability we have*

$$\Delta \geq \Delta_0, \qquad \omega \geq \omega_0.$$

Proof. Theorem 1.1 and Theorem 3.1 of [17] proves the following result: Let $\lambda > \Delta > 1$ be fixed. Assume that $A = \frac{1}{\sqrt{n}} B$ where B is an $n \times N$ random Gaussian matrix with i.i.d. $N(0, 1)$ entries such that $N/n = \lambda$. Then there exists a constant $c > 0$ depending only on τ_0, λ and Δ such that

$$\min_{S \subseteq \{1, \ldots, N\}, |S| \geq \Delta n} \sigma_n(A_S) \geq c$$

with probability at least $1 - 3e^{-\tau_0 n}$. The value c was explicitly estimated in terms of τ_0, λ and Δ in the proof of Theorem 3.1 in [17].

The theorem now readily follows. Observe that because $r_0 > 2$, in the expression for Δ we may choose $\lambda = r_0$ $\Delta = \frac{r_0}{2} > 1$ and clearly we have

$$\Delta \geq \min_{S \subseteq \{1, \ldots, N\}, |S| \geq \Delta n} \sigma_n(F_S) \geq \Delta_0,$$

for some $\Delta_0 > 0$ independent of n. For ω we may choose $\lambda = r_0$ and $\Delta = r_0 - 1 > 1$. Again the theorem of [17] implies that

$$\omega \geq \min_{S \subseteq \{1, \ldots, N\}, |S| \geq \Delta n} \sigma_n(F_S) \geq \omega_0. \qquad \square$$

In the theorem the values Δ_0 and ω_0 can be estimated explicitly using the estimates in [17]. Here with high probability is in the standard sense that the probability is at least $1 - c_0 e^{-\beta n}$ for some $c_0, \beta > 0$. Thus

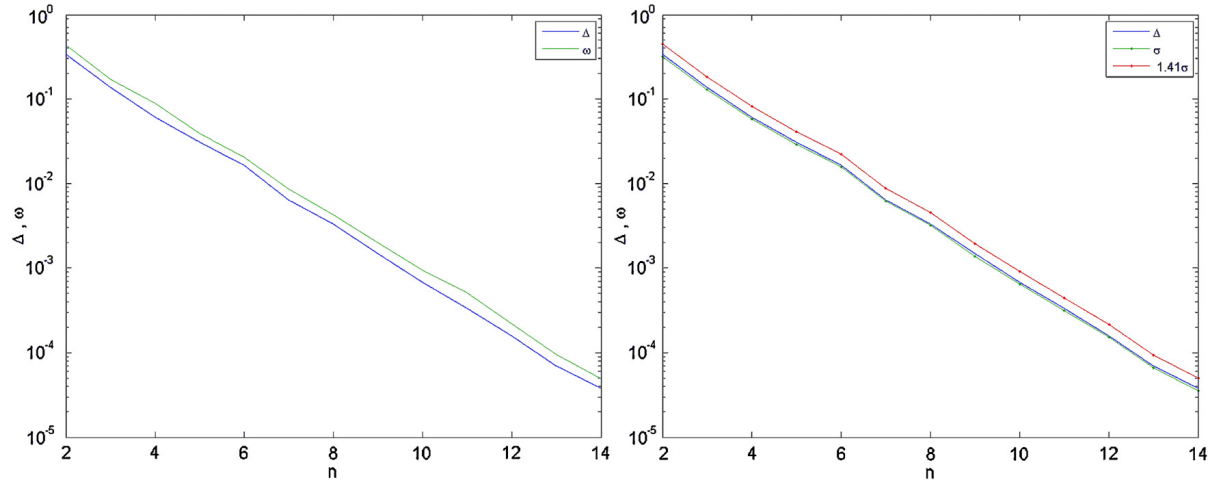


Fig. 1. Plots of sample medians of Δ and ω (left plot) and Δ and σ, $\sqrt{2}\sigma$ (right plot) for randomly generated frames of size $m = 2n$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

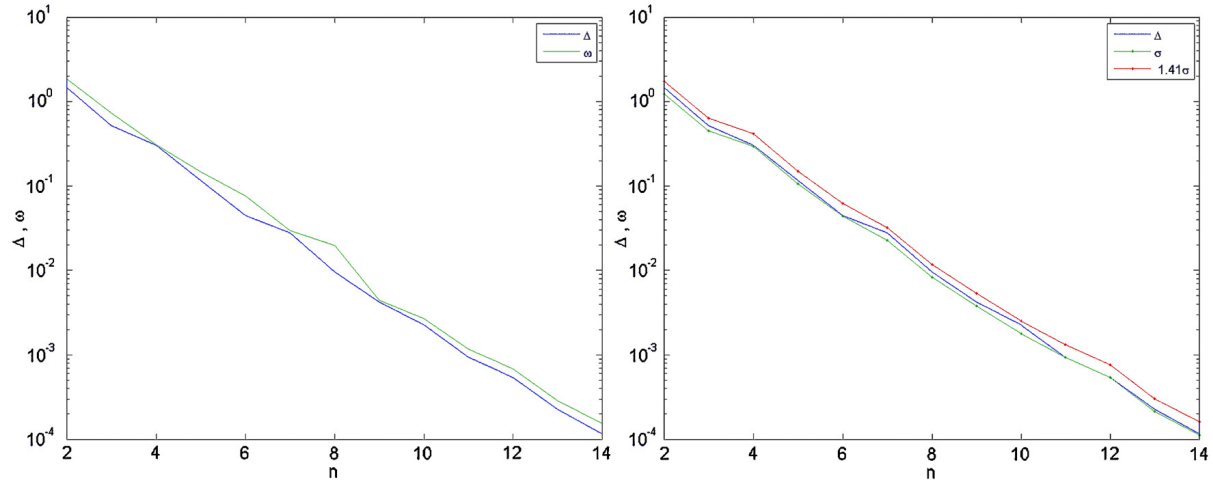


Fig. 2. Plots of largest sample value of Δ and ω (left plot) and Δ and σ, $\sqrt{2}\sigma$ (right plot) for randomly generated frames of size $m = 2n$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

scale independent stable phaseless reconstruction is possible whenever the redundancy is greater than $2+\Delta$, $\Delta > 0$, at least for random Gaussian matrices.

6. Numerical examples

In this section we present two numerical studies of the stability bounds derived earlier.

1. First consider the following setup. For each n between 2 and 14 we generate 100 realizations of random frames of $m = 2n$ vectors where each entry is i.i.d. normally distributed with zero mean and unit variance. For each realization we compute Δ, ω and σ. For each fixed n we compute the sample median, the largest sample value and the smallest sample value of these random variables.

Fig. 1 contains the plots of sample medians of Δ, ω and σ's for each value of n. The left plot contains only Δ (the lower Lipschitz constant) and ω (the lower Lipschitz constant for small perturbations); the right plot contains Δ and the two bounds σ and $\sqrt{2}\sigma$ as obtained in [9]. Similar statistics are plotted in Fig. 2 where sample medians are replaced by the largest sample values, and in Fig. 3 where sample medians are replaced by smallest sample values.

Note there is about 1–2 orders of magnitude spread between the largest and the smallest sample value of these random variables.

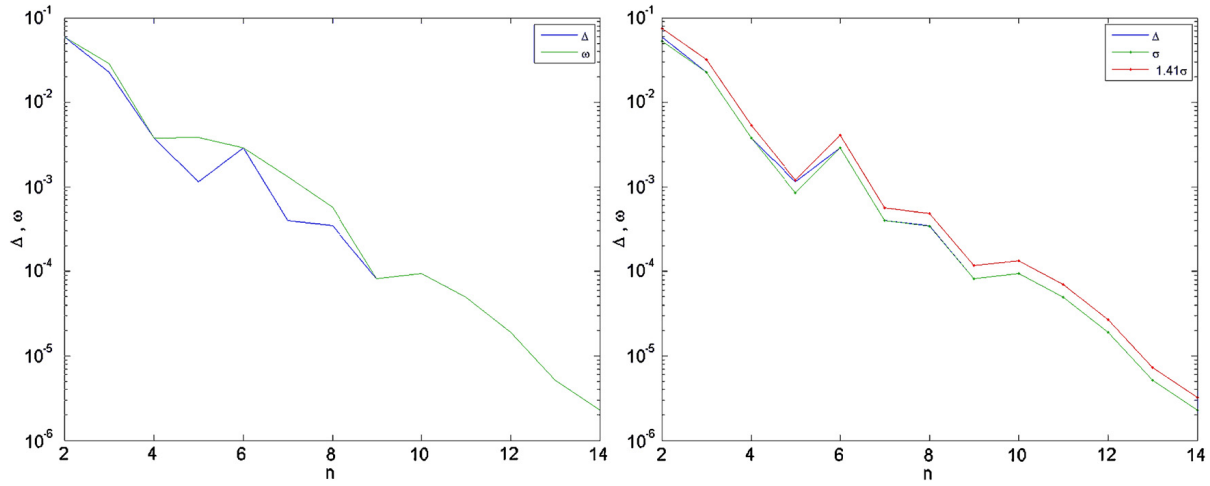


Fig. 3. Plots of largest sample value of Δ and ω (left plot) and Δ and $\sigma, \sqrt{2}\sigma$ (right plot) for randomly generated frames of size $m = 2n$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

2. Next we consider the following specific example. $H = \mathbb{R}^2$, $m = 4$ and the frame containing

$$f_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad f_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad f_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad f_4 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

which is a tight frame of bounds $A = B = 3$. The frame is full spark hence phase retrievable. The bounds Δ and ω defined by (4.4) and (4.5) are given by

$$\Delta = \sqrt{3 - \sqrt{5}} = 0.874032, \quad \omega = 1$$

which corresponds to choices $S = \{1, 3\}$ and $S = \{1, 2, 3\}$, respectively. The parameters $\boldsymbol{\sigma}$ introduced in (4.16) is given by

$$\boldsymbol{\sigma} = \sqrt{\frac{3 - \sqrt{5}}{2}} = 0.618034$$

and corresponds to $S = \{1, 3\}$. The parameter τ introduced in the statement of Theorem 4.2 is given by the same expression, $\tau = \boldsymbol{\sigma} = \sqrt{\frac{3-\sqrt{5}}{2}} = 0.618034$ and corresponds to the same selection $S = \{1, 3\}$.

Tedious algebra can provide closed form expressions for $\rho_\varepsilon(x)$ as function of radius ε. Because of scaling relation $\rho_{c\varepsilon}(cx) = \rho_\varepsilon(x)$ for all $c > 0$ it follows that only the direction of x describes this function. For instance for $x^{(1)} = (1, 0)$ we obtain the following expression:

$$\rho_\varepsilon\big(x^{(1)}\big) = \begin{cases} \sqrt{3}, & \varepsilon < \frac{1}{\sqrt{2}} \\ \sqrt{3 - \frac{4\sqrt{2}}{\varepsilon} + \frac{4}{\varepsilon^2}}, & \frac{1}{\sqrt{2}} \le \varepsilon < \sqrt{2} \\ 1, & \sqrt{2} \le \varepsilon \end{cases}$$

For $x^{(2)} = (1, 1)$ we obtain:

$$\rho_\varepsilon\big(x^{(2)}\big) = \begin{cases} \sqrt{3}, & \varepsilon < 1 \\ \sqrt{3 - \frac{4}{\varepsilon} + \frac{4}{\varepsilon^2}}, & 1 \le \varepsilon < 2 \\ \sqrt{2}, & 2 \le \varepsilon \end{cases}$$

The plots of these two functions are depicted in Fig. 9.

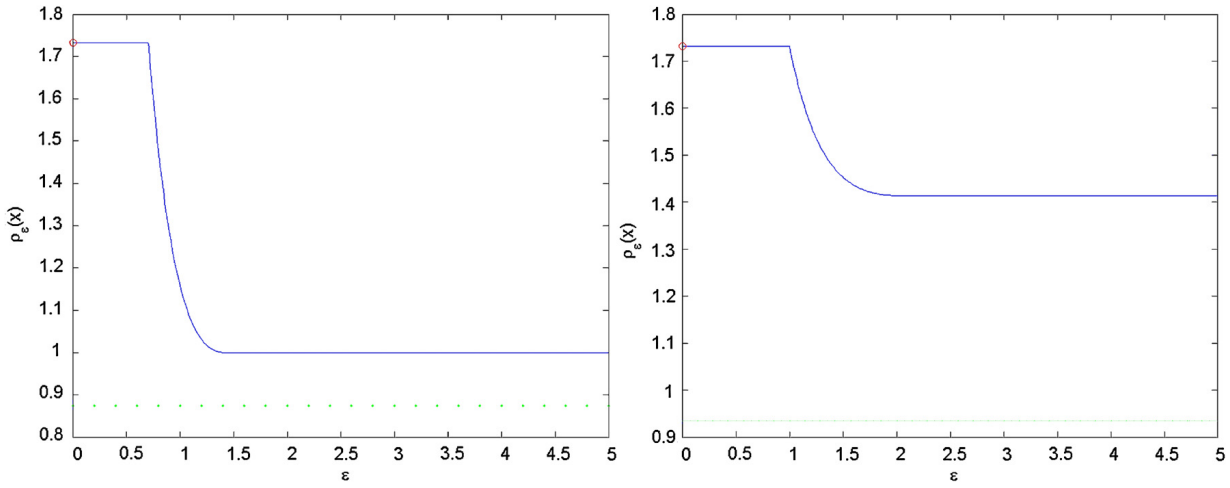


Fig. 4. Plots of $\rho_\varepsilon(x^{(1)})$ (left plot) and $\rho_\varepsilon(x^{(2)})$ (right plot) as function of radius ε. The red circle is at $\sqrt{A} = \sqrt{3}$. The horizontal dotted line is the lower bound $\Delta = 0.874$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

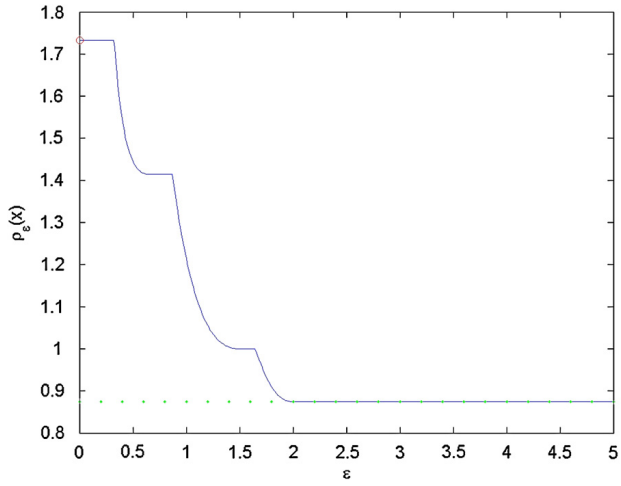


Fig. 5. Plots of $\rho_\varepsilon(x^{(3)})$ as function of radius ε. The red circle is at $\sqrt{A} = \sqrt{3}$. The horizontal dotted line is the lower bound $\Delta = 0.874$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Following the proof of Theorem 4.3 it follows the critical direction that achieves the lower bound $\sqrt{\Delta}$ is given by $x = u + v$ where u and v are the two normalized eigenvectors associated to the lowest eigenvalue (i.e. the lower frame bound) for $\{f_1, f_3\}$ and $\{f_2, f_4\}$ respectively. The lowest eigenvalue is given by $\frac{3-\sqrt{5}}{2}$ and the eigenvectors are

$$u = \begin{bmatrix} -\sqrt{\frac{2}{5+\sqrt{5}}} \\ \frac{1+\sqrt{5}}{\sqrt{2(5+\sqrt{5})}} \end{bmatrix}, \qquad v = \begin{bmatrix} -\sqrt{\frac{2}{5-\sqrt{5}}} \\ \frac{1-\sqrt{5}}{\sqrt{2(5-\sqrt{5})}} \end{bmatrix}$$

and thus the critical vector is

$$x^{(3)} = u + v = \begin{bmatrix} -\sqrt{\frac{2}{5+\sqrt{5}}} - \sqrt{\frac{2}{5-\sqrt{5}}} \\ \frac{1+\sqrt{5}}{\sqrt{2(5+\sqrt{5})}} \frac{1-\sqrt{5}}{\sqrt{2(5-\sqrt{5})}} \end{bmatrix} = \begin{bmatrix} -1.3764 \\ 0.3249 \end{bmatrix}$$

The function $\rho_\varepsilon(x^{(3)})$ is computed numerically and is plotted in Fig. 5. For reference we pictured a circle at $\sqrt{A} = \sqrt{3}$ and we plotted a dotted line at $\sqrt{\Delta} = 0.874$. We remark in all three cases, the limit

$\lim_{\varepsilon\to 0}\rho_\varepsilon(x) = \sqrt{A} = \rho_0$ as predicted by Theorem 4.3. Furthermore, $\min_{\varepsilon>0,x}\rho_\varepsilon(x) = \Delta = \rho_\infty$ as proved in same Theorem 4.3.

Acknowledgments

The authors would like to thank Matt Fickus, Dustin Mixon and Jeffrey Schenker for very helpful discussions.

R. Balan was supported in part by NSF DMS-1109498. Y. Wang was supported in part by NSF DMS-1043032, and by AFOSR FA9550-12-1-0455.

References

[1] B. Alexeev, A.S. Bandeira, M. Fickus, D.G. Mixon, Phase retrieval with polarization, arXiv:1210.7752v1 [cs.IT], 2012.
[2] R. Balan, A nonlinear reconstruction algorithm from absolute value of frame coefficients for low redundancy frames, in: Proceedings of SampTA Conference, Marseille, France, May 2009.
[3] R. Balan, On signal reconstruction from its spectrogram, in: Proceedings of the CISS Conference, Princeton, NJ, May 2010.
[4] R. Balan, Reconstruction of signals from magnitudes of redundant representations, arXiv:1207.1134v1 [math.FA], 2012.
[5] R. Balan, Reconstruction of signals from magnitudes of redundant representations: the complex case, Available online arXiv:1304.1839v1 [math.FA], 2013.
[6] R. Balan, P. Casazza, D. Edidin, On signal reconstruction without phase, Appl. Comput. Harmon. Anal. 20 (2006) 345–356.
[7] R. Balan, P. Casazza, D. Edidin, Equivalence of reconstruction from the absolute value of the frame coefficients to a sparse representation problem, IEEE Signal Process. Lett. 14 (5) (2007) 341–343.
[8] R. Balan, B. Bodmann, P. Casazza, D. Edidin, Painless reconstruction from magnitudes of frame coefficients, J. Fourier Anal. Appl. 15 (4) (2009) 488–501.
[9] A.S. Bandeira, J. Cahill, D.G. Mixon, A.A. Nelson, Saving phase: injectivity and stability for phase retrieval, arXiv:1302.4618v2 [math.FA], 2013.
[10] E. Candés, T. Strohmer, V. Voroninski, PhaseLift: exact and stable signal recovery from magnitude measurements via convex programming, Comm. Pure Appl. Math. 66 (8) (2013) 1241–1274.
[11] E. Candés, Y. Eldar, T. Strohmer, V. Voroninski, Phase retrieval via matrix completion problem, SIAM J. Imag. Sci. 6 (1) (2013) 199–225.
[12] P. Casazza, The art of frame theory, Taiwanese J. Math. (2) 4 (2000) 129–202.
[13] D.A. Depireux, M. Elhilali, Handbook of Modern Techniques in Auditory Cortex (Otolaryngology Research Advances), Nova Science Publishers, 2014.
[14] Y.C. Eldar, S. Mendelson, Phase retrieval: stability and recovery guarantees, Available online arXiv:1211.0872.
[15] S.M. Kay, Fundamentals of Statistical Signal Processing. I. Estimation Theory, Prentice Hall PTR, 2010, 18th printing.
[16] I. Waldspurger, A. d'Aspremont, S. Mallat, Phase recovery, MaxCut and complex semidefinite programming, Available online, arXiv:1206.0102.
[17] Y. Wang, Random matrices and erasure robust frames, E-print http://arxiv.org/abs/1403.5969.
[18] Y. Wang, Worst case condition number for matrices with erasure, preliminary report.

MULTIPLE AUTHORS DETECTION: A QUANTITATIVE ANALYSIS OF DREAM OF THE RED CHAMBER

XIANFENG HU, YANG WANG, AND QIANG WU

Abstract. Inspired by the authorship controversy of *Dream of the Red Chamber* and the application of machine learning in the study of literary stylometry, we develop a rigorous new method for the mathematical analysis of authorship by testing for a so-called chrono-divide in writing styles. Our method incorporates some of the latest advances in the study of authorship attribution, particularly techniques from support vector machines. By introducing the notion of relative frequency as a feature ranking metric our method proves to be highly effective and robust.

Applying our method to the Cheng-Gao version of *Dream of the Red Chamber* has led to convincing if not irrefutable evidence that the first 80 chapters and the last 40 chapters of the book were written by two different authors. Furthermore, our analysis has unexpectedly provided strong support to the hypothesis that Chapter 67 was not the work of Cao Xueqin either.

We have also tested our method to the other three Great Classical Novels in Chinese. As expected no chrono-divides have been found. This provides further evidence of the robustness of our method.

1. Introduction

Dream of the Red Chamber (红楼梦) by Cao Xueqin (曹雪芹) is one of China's Four Great Classical Novels. For more than one and a half centuries it has been widely acknowledged as the greatest literary masterpiece ever written in the history of Chinese literature. The novel is remarkable for its vividly detailed descriptions of life in the 18th century China during the Qing Dynasty and the psychological affairs of its large cast of characters. There is a vast literature in *Redology*, a term devoted exclusively to the study of *Dream of the Red Chamber*, that touches upon virtually all aspects of the book one can imagine, from the analysis of even minor characters in the book to in-depth literary study of the book. Much of the scope of Redology is outside the focus of this paper.

The original manuscript of *Dream of the Red Chamber* began to circulate in the year 1759. The problems concerning the text and authorship of the novel are extremely complex and have remained very controversial even today, and they remain an important part of Redology studies. Cao, who

Key words and phrases. Dream of the Red Chamber, Cao Xueqing, Redology, machine learning, support vector machine (SVM), recursive feature elimination (RFE), literary stylometry, authorship authentication, chrono-divide.

The second author is supported in part by the National Science Foundation grant DMS-0813750 and DMS-1043034.

died in 1763-4, did not live to publish his novel. Only hand-copied manuscripts – some 80 chapters – had been circulating. It was not until 1791 the first printed version was published, which was put together by Cheng Weiyuan (程伟元) and Gao E (高鄂) and was known as the *Cheng-Gao version*. The Cheng-Gao version has 120 chapters, 40 more than the various hand-copied versions that were circulating at the time. Cheng and Gao claimed that this "complete version" was based on previously unknown working papers of Cao, which they obtained through different channels. It was these last 40 chapters that were the subject of intense debate and scrutiny. Most modern scholars believe that these 40 chapters were not written by Cao. Many view those late additions as the work of Gao E. Some critics, such as the renowned scholar Hu Shi (胡适), called them forgeries perpetrated by Gao, while others believe that Gao was duped into taking someone else's forgery as an original work. There is, however, a minority of critics who view the last 40 chapters as genuine.

The analysis of the authenticity of the last 40 chapters has largely been based on the examination of plots and prose style by Redology scholars and connoisseurs. For example, many scholars consider the plotting and prose of the last 40 chapters to be inferior to the first 80 chapters. Others have argued that the fates of many characters in the end were inconsistent with what earlier chapters have been foreshadowing. A natural question is whether a mathematical stylometry analysis of the book can shed some light on this authenticity debate.

The problem of style quantification and authorship attribution in literature goes at least as far back as 1854 by the English mathematician Augustus De Morgan [7], who in a letter to a clergyman on the subject of Gospel authorship, suggested that the lengths of words might be used to differentiate authors. In 1897 the term *stylometry* was coined by the historian of philosophy, Wincenty Lutaslowski, as a catch-all for a collection of statistical techniques applied to questions of authorship and evolution of style in the literary arts (see e.g. [12]). Today, literary stylometry is a well developed and highly interdisciplinary research area that draws extensively from a number of disciplines such as mathematics and statistics, literature and linguistics, computer science, information theory and others. It is a central area of research in statistical learning (see e.g. [9]). A popular classic technique for stylometric analysis of authorship involves comparing frequencies of the so-called *function words*, a class of words that in general have little content meaning, but instead serve to express grammatical relationships with other words within a sentence. Although this technique is still widely used today, the field of literary stylometry has seen impressive advances in recent years, with more and more new and sophisticated mathematical techniques as well

as softwares being developed. We shall not focus on these advances here. Instead we refer all interested readers to the excellent survey articles by Juola [10] and Stamatatos [14] for a comprehensive discussion of the latest advances in the field.

Although there is a vast Redology literature going back over 100 years, the number of studies of the book based on mathematical and statistical techniques is surprisingly small, particularly in view of the fact that such techniques have been used widely in the West for settling authorship questions. Among the notable efforts, Cao [3] meticulously broke down a number of function characters and words into classes according to their functions. By analyzing their frequencies in the first 40 chapters, the middle 40 chapters and the last 40 chapters, Cao concluded that the first 80 chapters and the last 40 chapters were written by different authors. Zhang & Liu [2] examined the occurrence of 240 characters in the book that are outside the GB2312 encoding system, and found that 210 of them have appeared exclusively in the first 80 chapters while only 20 of them have appeared exclusively in the last 40 chapters. This led to the same conclusion by the authors. Yue [1] studied the authorship by combining both historical knowledge and statistical tools. In the study Yue tested two hypotheses, that the last 40 chapters were not written by the same author or they were written by the same author. His study focused on the frequencies of 5 particular function characters, the proportion of texts to poems in each chapter, and a few other stylometric peculiarities such as the number of sentences ended with the character "Ma" (吗). Using various statistical techniques the comparisons led the paper to draw the conclusion that it is unlikely that the first 80 chapters and the last 40 chapters were written by the same author. At the same time, using historic knowledge about the book and the original author Cao Xueqin, the paper also speculated that it was not likely that the last 40 chapters were created entirely by a single different author such as Gao E. In the opposite direction, the studies of Chan [6] and Li & Li [11] concluded that the entire book was likely written by a single author. The study [11] focused on the usage of functional characters while [6] examined the usage of some eighty thousand characters. Both studies tabulated the frequencies of the selected characters, which led to a frequency vector for each of the first 40 chapters, the middle 40 chapters and the last 40 chapters. The correlations of these frequency vectors were computed. In [11] the correlations were found to be large enough for the authors to conclude that the entire 120 chapters of the book were written by the same author. In [6] a fourth frequency vector using parts of the book *The Gallant Ones* (儿女英雄传) was added for comparison. The author found significantly higher correlations among the first three frequency vectors from chapters of *Dream of the Red Chamber* than the correlations between the

fourth frequency vector and the first three. This fact formed the basis of the conclusion by the author that all 120 chapters were written by a single author. A different conclusion was reached by Li [4]. By analyzing the frequencies of 47 functional characters and applying several statistical tests the author conjectured that the last 40 chapters were put together by Gao E using unedited and unfinished manuscripts by Cao Xueqin and his family members.

Although some of these aforementioned studies are impressive in their scopes, missing conspicuously from the Redology literature are studies based on the latest advances in literary stylometry, particularly some of the new and powerful methods from machine learning theory. While comparing the frequencies of function characters and words is clearly a viable way to analyze the authorship question, care needs to be taken to account for random fluctuations of these frequencies, especially when some of the function characters and words used for comparison have limited occurrences overall in the book and some times not at all in some chapters. None of the aforementioned studies employed cross validation to address random fluctuations. We have substantial reservations about drawing conclusions from correlations alone as in the studies of Chan [6] and Li & Li [11], because the differentiating power of any single variable such as correlation is rather limited. It would be interesting to see a more comprehensive study of correlations on a large corpus of texts in Chinese to determine its effectiveness as a metric for authorship attribution, something the authors failed to do in both studies. The use of the book *The Gallant Ones* in [6] for benchmark comparison is curious to us in particular, especially considering that the author did not limit to just function characters. The two books are of two different genres and are different in their respective background settings. Considering these differences *and* the fact that *The Gallant Ones* is known *not* to be written by Cao Xueqin, it would be a shock if the correlation between the last 40 chapters of *Dream of the Red Chamber* and the first 80 chapters is *not* higher than the correlation between the last 40 chapters and *The Gallant Ones*. It is possible that the correlation computed in [6] tells more about the genre than the authorship of the books. Again, without extensive evidence that using the same technique the correlation between two bodies of texts written by different authors is generally low even when the plots are closely related, the argument made in [6] is unconvincing at best. The objective of this paper is to present a rigorous stylometric analysis of *Dream of the Red Chamber* that incorporates some of the latest advances in the study of authorship attribution, particularly techniques from the theory of machine learning. To minimize the impact of random fluctuations we have meticulously followed well established protocols in selecting significant features by proper randomization of training and testing samples.

We shall detail our methodology in the next section, including feature construction and selection techniques in machine learning. In Section 3, we use our approach to the study of authorship of *Dream of the Red Chamber* and show the experimental results. In Section 4 we use our approach to analyze the other three Great Classical Novels. Finally in Section 5 we present some additional comments and our conclusions.

2. Chrono-divide and Methodology

The main idea behind statistically or computationally-supported authorship attribution is that by measuring some textual features we can distinguish between texts written by different authors. Nearly a thousand different measures including sentence length, word length, word frequencies, character frequencies, and vocabulary richness functions had been proposed thus far [13] over the years. Some of these measures, such as frequencies of function words, have proven effective while others, such as length of words, have proven less effective [10]. The field of literary stylometry has seen impressive advances over the years, and has become an increasingly important research field in the digital age with the explosion of texts online.

This paper focuses on a particular class of authorship controversies, in which there is a suspected change of authorship at some point of a book. In other words, one suspects that the first X chapters of a book were written by one author while the remaining Y chapters were written by another. Clearly, the authorship controversy for *Dream of the Red Chamber* falls into this category. Since no two authors have exactly the same writing style, no matter how similar they might be, a book written in such a fashion would have a stylistic discontinuity going from Chapter X to Chapter $X+1$. If we can quantify the styles of the two authors by a stylometric function $S(n)$ (a classifier) where n denotes chapters, or chronologically ordered samples, of the book in question, this stylistic discontinuity will appear as a dividing point in the stylometric function $S(n)$ going from $n = X$ to $n = X + 1$. Because the samples are ordered by time, we shall call this divide in the stylometric function $S(n)$ a *chrono-divide in style*, or simply a chrono-divide. This paper develops a technique for verifying and detecting chrono-divides in books or other body of texts. Knowing X and Y, as it is the case with *Dream of the Red Chamber*, can help validating the conclusion but is not always necessary for our method. Our method does not apply to any body of texts where two authors share the writing in an interwoven way without a chrono-divide.

The underlying principle of our study is that if a book is in fact written by two authors A and B, then there should exist a group of features that characterize the difference of their respective styles. These features will lead to a stylometric function that separate the book into two different classes. In the rest of the paper we shall use the more conventional term *classifier* for such a stylometric function. The foundational principle for literary stylometry is built around finding such classifiers. Suppose that a chrono-divide in style exists. Then an effective classifier will show a break point somewhere in the middle of the book, before and after which the classifier gives positive values and negative values, respectively. Thus in analyzing a book suspected to be written by two authors with a chrono-divide, one can look for a classifier that gives rise to such a break point. The existence of such a classifier will provide strong support for the two-author hypothesis. Conversely, if such a classifier cannot be found then we can confidently reject the two-author with a chrono-divide hypothesis.

We use function characters and words to build and select a group of stylometric features having the highest discriminative power, and from which we construct our classifier. We shall detail our method in the following subsections.

2.1. **Initial stylometric feature extraction.** Suppose the book in question is suspected to be written by two authors. For simplicity we shall call the part written by author A *Part A* and the part written by author B *Part B*. In many cases, such as with *Dream of the Red Chamber*, both Part A and part B are known. In some cases, they are not precisely known. However, for books suspected to have a chrono-divide from authorship change, there is usually a good estimate for where the divide is. Typically the first few chapters can be confidently attributed to A and the last few chapters to B.

We begin by choosing a feature set consisting of the kinds of features that might be used consistently by a single author over a variety of writings. Typically, these features include the frequencies of words (or characters for books in Chinese), phrases, mean and variation of sentence length, and frequencies of direct speeches and exclamations, and others. In our analysis, to get a better understanding of an author's writing style, we first find the most frequently used characters and words in the book, e.g. we would find the 500 most frequently used characters in the whole book, from which we pick out only, say, n function characters. We choose m words (combinations of characters) among the 300 most frequently used words in the same way. An important point is that by selecting only function characters and words we obtain a selection of characters and words

that are *content independent*. This leads to an initial set of features consisting of the frequencies of the n characters and the m words, plus the mean and variance of sentence length as well as the frequencies of direct speeches and exclamations. These features will be computed over given sample texts of the book (e.g. chapters). We normalize each sample text in the following way: set the median of the mean and variation of sentence length and the frequencies of direct speeches, exclamations, n characters and m words in each work of A and B to be 1. For each sample, we now get $n + m + 4$ features.

2.2. **Data preparation.** Having constructed the appropriate feature vectors, we build a distinguishing model through a machine learning algorithm. To do so requires careful data preparation. Since we usually have in hand only limited samples while the number of features will be very large, building a model directly on the entire book will easily lead to over-fitting. To overcome the over-fitting problem, we use the standard technique of separating the whole data into samples consisting of training data and test data. Our model will be established based only on the training data while its performance is tested over the independent test data. If we know Part A and Part B already then a subset of each can be designated as training data. For books suspected to have a chrono-divide in style, the training data will consist of the first few chapters and the last few chapters. The rest of the book will be used as test data.

In order to obtain more training sets and testing sets we shall chunk the book in question into smaller pieces of sample texts of relatively uniform size and style. In all the books we have studied, we have kept the sample texts to be at least 1000 characters long. In the case of *Dream of the Red Chamber* each sample text is a chapter.

2.3. **Feature subset selection.** When we build authorship analysis the model using the training data only, we do not use all the features ($n + m + 4$ features). Instead we start out with all of them, but eventually select a subset of features that achieves the highest discriminative powers. Feature subset selection has been well understood for high dimensional data analysis in the machine learning context. First, the number of discriminative features may be small because the number of features an author uses in a consistently different way from others is usually not very big. Moreover, the classifier can perform very poorly if too many irrelevant features are included into the model. In this paper we will use Support Vector Machines Recursive Feature Elimination (SVM-RFE) introduced in [8] to realize feature selection.

SVM-RFE is a feature ranking method. Given the a set of samples we can use linear SVM to build a linear classifier. It ranks the importance of the features according to their weights. As mentioned above, because of large feature size and small sample size, the classifier might not be robust. In addition, the high correlation between the features may result in small weights for relevant features. Thus the ranking by SVM classifier directly may be inaccurate. In order to refine the ranking, the least important feature is removed and the linear SVM classifier is retrained. This new classifier provides a refined ranking for the remaining features. The process is then repeated until the ranking of all features are refined. This is the SVM-RFE method introduced [8]. The idea underlying SVM-RFE is that in each repeat, although the overall ranking may be poor, the least important feature is seldom a relevant one. By iteratively eliminating the least important features the new classifiers will become more and more reliable and hence will provide better and better ranking. In the application of gene expression data analysis SVM-RFE has been proven to be substantially superior to the SVM direct ranking without RFE.

However in general SVM-RFE is not stable under the perturbation of samples. A small change in samples may result in very different feature ranking. There are two possible reasons. One is that the highly correlated variables are too sensitive and may be ranked in different orders by different classifiers. Another is that, due to the randomness, some subset of samples might be singular in the sense that they are less representative for the whole data structure. As a result the SVM classifiers are over-fitting and the feature ranking by SVM-RFE is therefore unreliable. The first situation is less harmful for classification performance while the second is vital. To overcome this phenomenon and guarantee the stability of the ranking, we use a pseudo-aggregation technique. We randomly choose a subset of training samples to run SVM-RFE to select the top important features. This process is repeated tens or hundreds times and only those features that appear important very frequently are deemed as truly important ones. This removes the randomness and results in a much more reliable ranking.

With this ranking of features, we can conclude which statistics are useful for quantifying the writing style. We use cross validation to select the number of features included in the final classification model. This group of features is a stable and most discriminative subset of features. A final classifier is built to classify the test data.

2.4. **Data analysis.** The classifier we have built is used to analyze the authorship question. We examine the discriminative power of the classifier on the training data. If it cannot even reliably

classify the training data we can convincingly reject the two-author hypothesis. Even if it can the telling story will be whether it can classify, or detect a chrono-divide, from the test data. If it fails then again we should reject the two-author hypothesis. On the other hand, if the the classifier classifies the training data, and it can also classify the test data accurately or detect a clear chrono-divide, we can then convincingly conclude that the book does contain two different writing styles and can therefore be confidently attributed to two different authors. Moreover, the feature subset and the classifier describe the difference of the two authors' writing styles.

2.5. **The algorithm.** In the following we summarize the process of our algorithm:

(1) Initialize the data (the book), which contains parts A and B suspected to be written by two different authors.
(2) Split part A and part B into many sections and extract the features for each section as described in section 2.1. This forms the whole data set D, containing D_A and D_B.
(3) Choose a portion (e.g. 20%-30%) of D_A and D_B respectively to form the test data set and leave the remaining as the training data set. The test data will not be used until the final model is built.
(4) Randomly choose a subset from the training data as modeling data and the rest (again 20%-30%) as the validation data. Run SVM-RFE on the modeling data and using the validation data to determine all the parameters used. This provides a ranking of all the $n + m + 4$ features extracted in step 2.
(5) For d range from 1 to $n+m+4$, build a classifier using only the top d features and evaluate their performance on the validation data. The best model is the one with minimal validation error and minimal number of top features. The feature subset of this best model is recorded.
(6) Repeat T times step 4 and step 5 to obtain T best models and T subsets of corresponding important features. We recommend T to be larger than 50. Rank all the features in these subsets according to their appearance frequency. Denote N as the total number of features included.
(7) For $d = 1, ..., N$, using cross validation to select the number of features that should be included in the final classifier. Denote it by d_*. Note we require both the cross validation error and the number of features to be as small.
(8) Retrain the model using the whole training set based on this top d_* important features.

(9) Using the classifier to classifying the test data. Draw the conclusion according to the performance.

Since our ranking process involves aggregation of large number of models that are trained using SVM-RFE based on different subsets of the same data source, we refer to our approach as pseudo-aggregation SVM-RFE method.

3. Analysis of *Dream of the Red Chamber*

Having established a rigorous protocol for the study of authorship of a body of texts, we apply this protocol to investigate the authorship controversy of the Cheng-Gao version of *Dream of the Red Chamber*. In particular we investigate the existence of a chrono-divide at Chapter 80.

The book is first divided into samples. To balance the number of samples, we generate one sample for each of the first 80 chapters while using the conventional practice of duplicating each of the last 40 chapters into two chapters to obtain 80 samples. From those samples we extract the features by calculating the statistics proposed in subsection 2.1. These features are then normalized for fair comparison. In total we have 196 variables. They are the 144 characters and 48 words, the normalized mean and variation of sentence length, and the frequencies of direct speeches and exclamations.

To investigate the authorship controversy we perform three separate tests. First we build a classifier for the whole book and look for the existence of a chrono-divide at Chapter 80. For added robustness and reliability we also perform the same tests only on the first 80 chapters and the last 40 chapters.

3.1. **Separability of the chapters by Cao and Gao.** In the first experiment we apply our method to the whole Chen-Gao version of *Dream of the Red Chamber*. Samples from the first 60 chapters are designated as training samples for one class while samples from the last 30 chapters are designated as training samples for another class. The remaining samples, from Chapter 61 to 90, are held out as test samples. The training samples are further randomly split into modeling data of 80 samples and validation data of 40 samples. The SVM-RFE is repeated 100 times and d_* is chosen using 50 cross validation runs. We have the following observations.

Instability of SVM-RFE. The randomness of the modeling set has resulted in very substantial fluctuations in the number of features selected as well as feature rankings. The resulted

Modeling set	Features Selected	Validation Error
1	去，得，就，回，知，到，时，呢，倒，别，作	5/40
2	回，方，没，好些	1/40

Table 1. The features and validation errors of the classifiers obtained from two randomly selected modeling subsets.

classifier may also perform quite differently. Table 3.1 lists the features selected using two different modeling data sets. One selects 11 features and the other selects only 4, with only one feature in common. The classifiers also perform differently. The experiments clearly establish the instability of SVM-REF.

Given such instability one cannot reliably draw any conclusions from any single run. For example, if a modeling data set separates the training data well it might be due to over-fitting. Conversely if it separates poorly it might be due to under-fitting. This problem is overcome with our Pseudo Aggregate SVM-RFE method.

Stability of Pseudo Aggregate SVM-RFE. Our pseudo aggregate SVM-RFE approach repeats SVM-RFE 100 times using randomized data sets. The data set from each repeat is used to select a set of features, from which a classifier is being built. For simplicity we shall refer to the data set, features and the resulting classifier together from a repeat as a *model*. To counter random fluctuations we consider important features to be those that appear frequently among the 100 classifiers. This reduced the instability caused by randomness. In fact, our belief is as follows: if the two classes are well separated, there should exist a set of features that help to build a good classifier. Most modeling subsets should be able to select these features out and only a limited number of modeling sets might be singular and miss them. Conversely, if the two classes cannot be well separated, no consistently discriminative features exist. Different modeling set may lead to totaly different feature subset. As a result, no feature appears with high frequency in all 100 models. This philosophy, however, is only partially true. When the two classes cannot be separated, the modeling process sometimes can overfit the data by selecting a lot of variables which results in high absolute frequencies for some less important or irrelevant features. Such a phenomenon is usually accompanied by large number of variables and low validation accuracy. To improve the process we propose a more appropriate metric,

which we call *relative frequency*. In relative frequency we weight the frequency by two criteria. In the first criteria a variable appearing in short models is weighted more than the variables appearing in long models. This leads to a weight of $h(n_j)$ for a variable in the j-th model, with n_j being the number of variables in the j-th model. In the second criteria a variable in a model with high predictive accuracy is weighted more than a variable with poor predictive accuracy. This provides another weight $g(A_j)$ for a variable in the j-th model, where A_j denotes the accuracy of the j-th model computed from the validation process. Mathematically the relative frequency for a variable x_i in a test run of M repeats is defined as

$$rf(x_i) = \frac{1}{M}\sum_{j=1}^{M} g(A_j)h(n_j)\mathbf{1}(x_i \text{ appears in model } j). \tag{3.1}$$

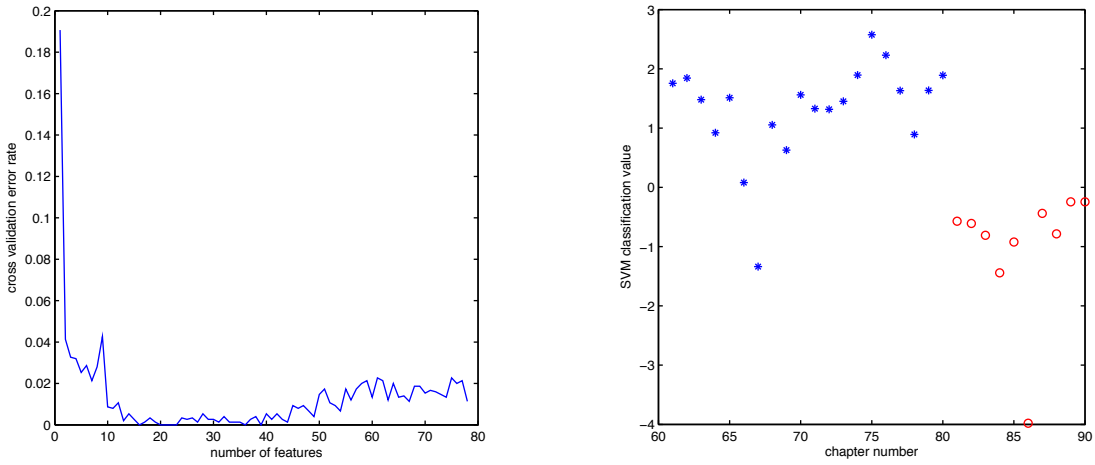
In our study we always set $M = 100$. Also, we set $g(A_j) = \exp(\frac{A_j-1}{[2A_j-1]_+})$ where $[t]_+ = \max\{0, t\}$ and $h(n_j) = [1 - cn_j]_+$ for some constant c. For $g(A_j)$ the idea is that if the weight should decay fast if the accuracy is close to 50% or less because it indicates that the classifier is simply not effective. For $h(n_j)$ we put in a penalty for the number of variables used in a model. In our experiments we have chosen $c = 1/30$, which seems to work well.

Our experiments show that features yielded from relative frequency rankings are very stable and consistent. We have performed runs of 100 repeats using different random seeds in MATLAB, and the results are always similar. An additional benefit of using relative frequency instead of absolute frequency is that the existence of an effective classifier is typically accompanied by high relative frequencies for the top features, while low relative frequencies for the top features usually imply poor separability. Hence we can use relative frequency as a simple guide on the separability of the samples. We will show some examples in the next section.

Results and conclusion. In Experiment 1 we have performed a run of 100 repeats on the entire Cheng-Gao version of *Dream of the Red Chamber*. Altogether 70 features have appeared in at least one model. However, of those only a small number of them have appeared with high enough frequency to be viewed as being important. We apply cross validation to select the number of features, and the mean cross validation error rate against different number of features is plotted in Figure 1 (a). The figure tells us that 10 to 50 features are enough to tell the style difference between the two parts. Using less characters and words is insufficient, while using more degrades the performance also by bringing in too much noise. The small cross validation error rate is encouraging, and it is already hinting a strong possibility that

the two training sample sets have significant stylistic differences to support the two-author hypothesis.

To settle the two-author hypothesis more definitively we apply our classifier on the test data, which until now has never been used during the feature selection and classifier modeling process. In particular we investigate the existence of a chrono-divide in the values obtained through classifier. Figure 1 (b), which plots these values, clearly shows a chrono-divide at Chapter 80: For Chapter 81-90 the classifier yields all negative values while for Chapters 61-80 the classifier yields all positive values with the exception of Chapter 67. Allowing some statistical abberations to occur, our results provide an extremely convincing if not irrefutable evidence that there exist clear stylometric differences between the writings of the first 80 chapters and the last 40 chapters. This difference strongly supports the two-author hypothesis for *Dream of the Red Chamber*. We also note that our investigation did not need to assume that the knowledge that the stylistic change should be at Chapter 80. The fact that the chrono-divide we have detected is indeed at Chapter 80 lends even stronger support to the two-author hypothesis.

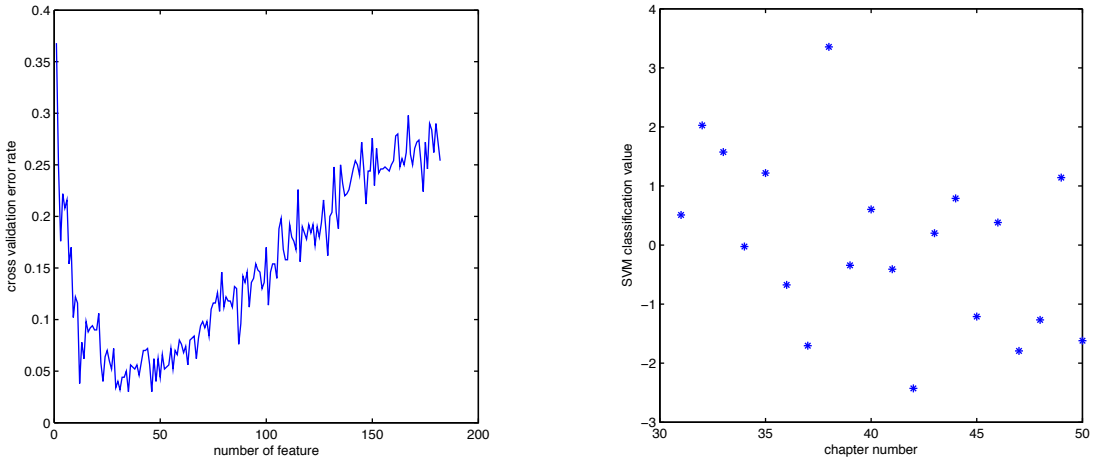


(a) (b)

Figure 1. Experiment 1: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 60-90.

Interestingly, the fact that Chapter 67 appeared as an "outlier" in our classification serves as further evidence to the validity of our analysis. It was only after the tests we realized that the authorship of Chapter 67 itself is one of the controversies in Redology. Unlike the main controversy about the authorship of the first 80 chapters and the last 40 chapters, experts are less unified in their positions here. Again, our results strongly suggests that Chapter 67 is stylistically different from the rest of the first 80 chapters, and it may not be written by Cao. Our finding is consistent with the conclusion of [5].

3.2. **Non-separability of the first 80 chapters.** To further validate our method we apply the same tests to the first 80 chapters of *Dream of the Red Chamber* to see whether we can get a chrono-divide (Experiment 2). We use the first 30 and last 30 chapters as the training data and leave chapters 31-50 as the test data. Figure 2 shows the mean cross validation error and the values of SVM classifier on the test data chapters 31-50. The experiment shows many more features have been selected in the 100 repeats, implying the difficulty of find a consistent subset of discriminative features. The large errors on the training data also indicate the difficulty for separation. When the classifier is applied to the test data, there is clearly no chrono-divide. This suggests that our method yields a conclusion that is completely consistent with what is known.



(a) (b)

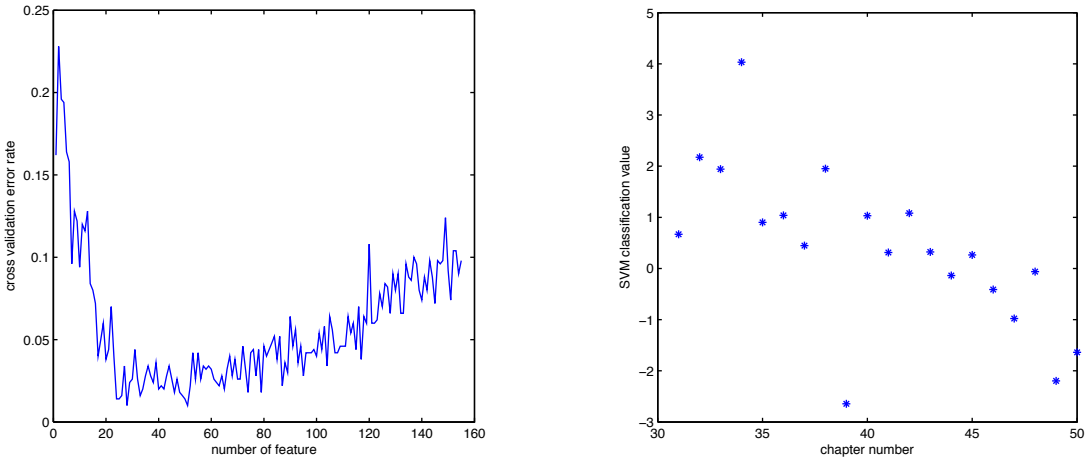
Figure 2. Experiment 2: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 31-50. Note there is no chrono-divide.

3.3. **Analysis of chapters 81-120: style change over time.** We next apply our method to the last 40 chapters (Experiment 3). Our first experiment has already confirmed that they are unlikely to be written by Cao. However, there are still debates on whether these were written entirely by one author (most likely Gao himself), or by more than one author. Our mathematical analysis may offer some insight here.

We split the 40 chapters into two subsets as before. The training data include Chapters 81-95 as one class and Chapters 106-120 as another. The test data are the middle 10 chapters. Because of the relatively small number of samples we have subdivided each chapter into 2 sections to increase the sample size. As a result we now have 60 samples in the training data and 20 in test data, with 2 samples corresponding to one chapter. The mean cross validation error of the final classifier and its classification values on the test samples are shown in Figures 3 (a) and (b) respectively.

In this experiment we observe that the performance in terms of both the classifier and feature ranking is noticeably worse than that in Experiment 1 but substantially better than that in Experiment 2. Furthermore, unlike the results from the first two experiments, the values from the classifier show an interesting trend. Compared with Figure 2 (b) where the values appeared to lack any order, the values here exhibit a clear gradual downward shift. On the other hand, compared to Figure 1 (b) the values plotted in 3 (b) do not show a clear sharp chrono-divide, even though the values change gradually from being positive to being negative. What it tells us is that the writing style of the last 80 chapters had undergone a graduate change, but this change is unlikely to be due to change of authorship.

Our results here could be subject to several interpretations. One plausible interpretation is that Gao might indeed obtained some incomplete set of manuscripts by Cao, and tried to complete the novel based on what he had obtained. The style change is a result of the lack of genuine work by Cao as the story developed. A more plausible interpretation is that the last 40 chapters were written by someone such as Gao trying to imitate Cao's style, and over time the author became sloppier and returned more and more to his own style.



(a) (b)

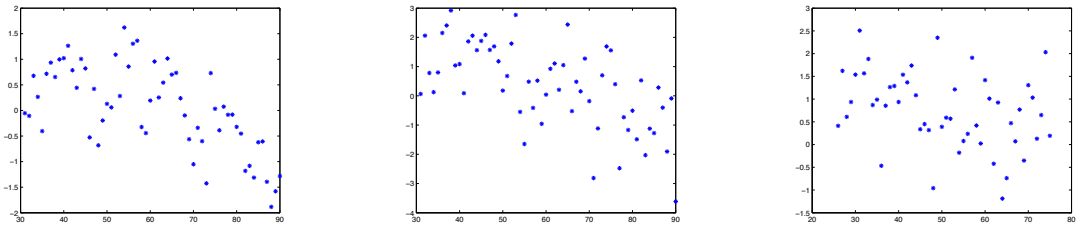
Figure 3. Experiment 3: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 96-105, which correspond to the samples 31-50 in all 80 samples. Note two samples come from one chapter in this experiment.

4. Analysis of the other three Great Classical Novels

To further bolster the credibility of our approach we test our method on the other three Great Classical Novels in Chinese literature, *Romance of the Three Kingdoms* (三国演义), *Water Margin* (水浒传), and *Journey to the West* (西游记). Unlike *Dread of the Red Chamber*, there is no authorship controversy for these other three novels. Thus if our method is indeed robust we should expect negative answers for the two-author hypotheses for all of them by finding no chrono-divides.

As with *Dream of the Red Chamber*, we split each of the three novels into training samples and test samples. Both *Romance of the Three Kingdoms* and *Water Margin* have 120 chapters. In both cases we designate the first 30 chapters and the last 30 chapters as the two classes of training data, and the middle 60 chapters as test data. For *Journey to the West* the two classes of training data are the first and last 25 chapters respectively, with the middle 50 chapters as test data.

We use the same procedure to test for chrono-divides on the three novels. Compared to *Dream of the Red Chamber*, the selected features show much lower relative frequencies, indicating difficulty in differentiating between the writing styles. Table 2 show the relative frequencies (with $c = 1/30$) of the top 8 features for each of the four Great Classical Novels. Also of note is that in the case of



(a) (b) (c)

Figure 4. Classification results from the test sampels of the other three classical novels: (a) *Romance of the Three Kingdoms*; (b) *Water Margin*; (c) *Journey to the West.*

Water Margin, 51 features are used to build a classifier from the 60 training data, which is clearly another strong indication of the difficulty.

Novel	Relative frequencies of top 8 features							
Dream of the Red Chamber	0.57	0.46	0.43	0.36	0.31	0.30	0.29	0.19
Romance of the Three Kingdoms	0.31	0.27	0.26	0.25	0.23	0.22	0.17	0.15
Water Margin	0.18	0.17	0.16	0.16	0.14	0.11	0.11	0.10
Journey to the West	0.03	0.03	0.02	0.02	0.02	0.02	0.02	0.02

Table 2. Relative frequencies of the top ranked 8 features in each of the four Great Classical Novels.

Figure 4 plots the values from the classifiers for all three novels. In all cases the values fluctuate in such a way that it is quite clear that no chrono-divides exist, as expected.

This analysis shows that our approach can reliably reject the two-author hypothesis when it is false, lending further support to the effectiveness and robustness of our method.

5. Conclusions

Inspired by authorship controversy of *Dream of the Red Chamber* and the application of SVM in the study of literary stylometry, we have developed a mathematically rigorous new method for the analysis of authorship by testing for a chrono-divide in writing styles. We have shown that the method is highly effective and robust. Applying our method to the Cheng-Gao version of *Dream of the Red Chamber* has led to convincing if not irrefutable evidence that the first 80 chapters and the last 40 chapters of the book were written by two different authors. Furthermore, our analysis

has unexpectedly provided strong support to the hypothesis that Chapter 67 was not the work of Cao Xueqin either.

The methodology in this paper is rather effective in selecting the most important features for classification through a new ranking system based on relative frequency. A series of future experiments should be included in the application of this methodology to wider range of works.

It is worth mentioning that there are several other attempts to complete *Dream of the Red Chamber* from the its first 80 chapters, among them is *Continued Dream of the Red Chamber* (续红楼梦) by Qi Zichen (秦子忱). Using the same features for building the classifier in Experiment 1, we can compute the Euclidean distances between all chapters and their distances of chapters from *Continued Dream of the Red Chamber*, see Figure 5. Surprisingly, although these features are obtained in favor of the differences between Cao and Cheng-Gao, they lead to even larger distance between the first 80 chapters and those chapters of *Continued Dream of the Red Chamber*. It obviously implies that the style of the 40 chapters by Cheng-Gao are more similar to the 80 chapters by Cao compared to *Continued dream of the Red Chamber*. Maybe that's why the Cheng-Gao version is more popular than other versions.

Acknowledgement. Throughout this project, Haiyi Jiang has enthusiastically participated in the discussions on the project, and his knowledge of *Dream of the Red Chamber* has been valuable to the study. The authors would like to thank him in particular for his support.

References

[1] 余清祥. 統計在紅樓夢的應用(註). *http://csyue.nccu.edu.tw/ch/1998RedChamber.pdf*.
[2] 张卫东and 刘丽川. 《红楼梦》前八十回与后四十回语言风格初探. 深圳大学学报, 1, 1986.
[3] 曹清富. 《红楼梦》后四十回决非曹雪芹所作——前八十回与后四十回虚词、词组及回目之比较. 红楼梦学刊, 01:288–319, 1985.
[4] 李贤平. 红楼梦成书新说. 《复旦学报》社会科学版, 5, 1987.
[5] 严安政. One Piece of Evidence that Chapters 64 and 67 Are Not the Original Version. 咸阳师范学院学报, 24(3), 2009.
[6] B.C. Chan. Authorship of The Dream of the Red Chamber: A Computerized Statistical Study of Its Vocabulary. *Dissertation Abstracts International Part A: Humanities and[DISS. ABST. INT. PT. A- HUM. & SOC. SCI.],*, 42(2):1981, 1981.
[7] A. de Morgan. Letter to rev. heald 18/08/1851, 1851.
[8] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.
[9] D.I. Holmes and J. Kardos. Who was the author? an introduction to stylometry. *CHANCE-BERLIN THEN NEW YORK-*, 16(2):5–8, 2003.
[10] P. Juola. Authorship attribution. *Foundations and Trends in information Retrieval*, 1(3):233–334, 2006.
[11] Guo-qiang Li and Rui-fang Li. Study Based on Statistics of word Frequency Research on Only Author of the" Dream of the Red Chamber"[J]. *Journal of Shenyang Institute of Chemical Technology*, 4, 2006.

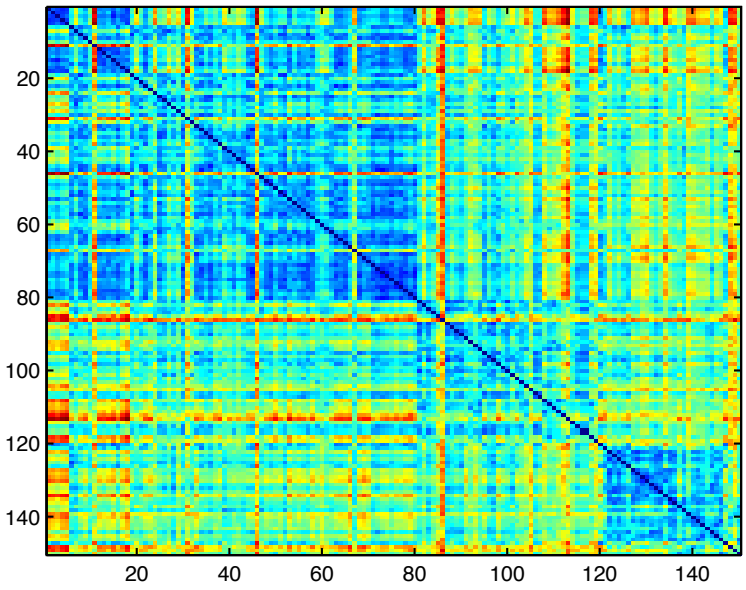


Figure 5. Distances between the first 80 chapters of the Cheng-Gao version, the last 40 chapters of the Cheng-Gao version, and 30 chapters of *Continued Dream of the Red Chamber* .

[12] A. Pawlowski. Wincenty lutoslawski-a forgotten father of stylometry. *Glottometrics*, 8:83–89, 2004.
[13] J. Rudman. The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities*, 31(4):351–365, 1997.
[14] E. Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556, 2009.

Department of Mathematics, Michigan State University, East Lanisng, MI 48824, USA.

E-mail address: hxf0204@gmail.com

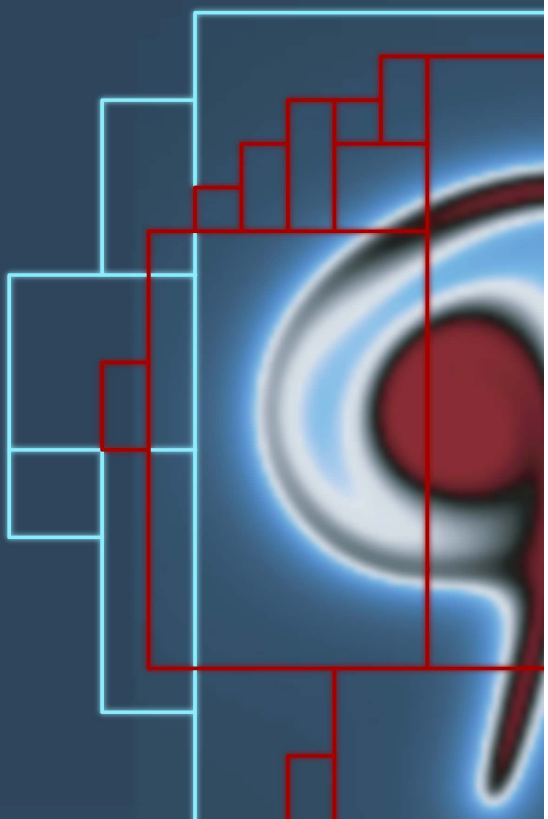
Department of Mathematics, Michigan State University, East Lanisng, MI 48824, USA.

E-mail address: ywang@math.msu.edu

Department of Mathematical Sciences, Middle Tennessee State University, Murfreesboro, TN 37132, USA.

E-mail address: qwu@mtsu.edu

Communications in Applied Mathematics and Computational Science



REVISIONIST INTEGRAL DEFERRED CORRECTION WITH ADAPTIVE STEP-SIZE CONTROL

ANDREW J. CHRISTLIEB, COLIN B. MACDONALD, BENJAMIN W. ONG AND RAYMOND J. SPITERI

vol. 10 no. 1 2015

msp

281

COMM. APP. MATH. AND COMP. SCI.
Vol. 10, No. 1, 2015
dx.doi.org/10.2140/camcos.2015.10.1

msp

REVISIONIST INTEGRAL DEFERRED CORRECTION WITH ADAPTIVE STEP-SIZE CONTROL

ANDREW J. CHRISTLIEB, COLIN B. MACDONALD,
BENJAMIN W. ONG AND RAYMOND J. SPITERI

Adaptive step-size control is a critical feature for the robust and efficient numerical solution of initial-value problems in ordinary differential equations. In this paper, we show that adaptive step-size control can be incorporated within a family of parallel time integrators known as revisionist integral deferred correction (RIDC) methods. The RIDC framework allows for various strategies to implement step-size control, and we report results from exploring a few of them.

1. Introduction

The purpose of this paper is to show that local error estimation and adaptive step-size control can be incorporated in an effective manner within a family of parallel time integrators based on revisionist integral deferred correction (RIDC). RIDC methods, introduced in [10], are "parallel-across-the-step" integrators that can be efficiently implemented with multicore [10; 6], multi-GPGPU [4], and multinode [9] architectures. The "revisionist" terminology was adopted to highlight that (1) RIDC is a revision of the standard integral defect correction (IDC) formulation [12], and (2) successive corrections, running in parallel but (slightly) lagging in time, revise and improve the approximation to the solution.

RIDC methods have been shown to be effective parallel time-integration methods. They can typically produce a high-order solution in essentially the same amount of wall-clock time as the constituent lower-order methods. In general, for a given amount of wall-clock time, RIDC methods are able to produce a more accurate solution than conventional methods. These results have thus far been demonstrated with constant time steps. It has long been accepted that local error estimation and adaptive step-size control form a critical part of a robust and efficient strategy for solving initial-value problems in ordinary differential equations (ODEs), in particular problems with multiple timescales; see [15], for example. Accordingly, in order to assess the practical viability of RIDC methods, it is important to establish

MSC2010: 65H10, 65L05, 65Y05.
Keywords: initial-value problems, revisionist integral deferred correction, parallel time integrators, local error estimation, adaptive step-size control.

whether they can operate effectively with variable step sizes. It turns out that there are subtleties associated with modifying the RIDC framework to incorporate functionality for local error estimation and adaptive step-size control: there are a number of different implementation options, and some of them are more effective than others.

The remainder of this paper is organized as follows. In Section 2, we review the ideas behind RIDC as well as strategies for local error estimation and step-size control. We then combine these ideas to propose various strategies for RIDC methods with error and step-size control. In Section 3, we describe the implementation of these strategies within the RIDC framework and suggest avenues that can be explored for a production-level code. In Section 4, we demonstrate that the use of local error estimation and adaptive step-size control inside RIDC is computationally advantageous. Finally, in Section 5, we summarize the conclusions reached from this investigation and comment on some potential directions for future research.

2. Review of relevant background

We are interested in numerical solutions to initial-value problems (IVPs) of the form

$$\begin{cases} y'(t) = f(t, y(t)), \quad t \in [a, b], \\ y(a) = y_a. \end{cases} \tag{1}$$

where $y(t) : \mathbb{R} \to \mathbb{R}^m$, $y_a \in \mathbb{R}^m$, and $f : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$. We first review RIDC methods, a family of parallel time integrators that can be applied to solve (1). Then, we review strategies for local error estimation and adaptive step-size control for IVP solvers.

2.1. ***RIDC.*** RIDC methods [10; 6; 4] are a class of time integrators based on integral deferred correction [12] that can be implemented in parallel via pipelining. RIDC methods first compute an initial (or *provisional*) solution, typically using a standard low-order scheme, followed by one or more corrections. Each correction revises the current solution and increases its formal order of accuracy. After initial startup costs, the predictor and all the correctors can be executed in parallel. It has been shown that parallel RIDC methods with uniform step-sizes give almost perfect parallel speedups [10]. In this section, we review RIDC algorithms, generalizing the overall framework slightly to allow for nonuniform step-sizes on the different correction levels.

We denote the nodes for correction level ℓ by

$$a = t_0^{[\ell]} < t_1^{[\ell]} < \cdots < t_{N^{[\ell]}}^{[\ell]} = b,$$

where $N^{[\ell]}$ denotes the number of time steps on level ℓ. In practice, the nodes on each level are obtained dynamically by the step-size controller.

2.1.1. *The predictor.* To generate a provisional solution, a low-order integrator is applied to solve the IVP (1). For example, a first-order forward Euler integrator applied to (1) gives

$$\eta_n^{[0]} = \eta_{n-1}^{[0]} + \big(t_n^{[0]} - t_{n-1}^{[0]}\big) f(t_{n-1}^{[0]}, \eta_{n-1}^{[0]}), \tag{2}$$

for $n = 1, 2, \ldots, N^{[0]}$, with $\eta_0^{[0]} = y_a$, and where we have indexed the prediction level as level 0. We denote $\eta^{[\ell]}(t)$ as a continuous extension [15] of the numerical solution at level ℓ, i.e., a piecewise polynomial $\eta^{[0]}(t)$ that satisfies

$$\eta^{[0]}(t_n^{[0]}) = \eta_n^{[0]}.$$

The continuous extension of a numerical solution is often of the same order of accuracy as the underlying discrete solution [15]. Indeed, for the purposes of this study, we assume $\eta^{[\ell]}(t)$ is of the same order as $\eta_n^{[\ell]}$.

2.1.2. *The correctors.* Suppose an approximate solution $\eta(t)$ to IVP (1) is computed. Denote the exact solution by $y(t)$. Then, the error of the approximate solution is $e(t) = y(t) - \eta(t)$. If we define the defect as $\delta(t) = f(t, \eta(t)) - \eta'(t)$, then

$$e'(t) = y'(t) - \eta'(t) = f(t, \eta(t) + e(t)) - f(t, \eta(t)) + \delta(t).$$

The error equation can be written in the form

$$\left[e(t) - \int_a^t \delta(\tau)\, d\tau \right]' = f(t, \eta(t) + e(t)) - f(t, \eta(t)), \tag{3}$$

subject to the initial condition $e(a) = 0$. In RIDC, the corrector at level ℓ solves for the error $e^{[\ell-1]}(t)$ of the solution $\eta^{[\ell-1]}(t)$ at the previous level to generate the corrected solution $\eta^{[\ell]}(t)$,

$$\eta^{[\ell]}(t) = \eta^{[\ell-1]}(t) + e^{[\ell-1]}(t).$$

For example, a corrector at level ℓ that corrects $\eta^{[\ell-1]}(t)$ by applying a first-order forward Euler integrator to the error equation (3) takes the form

$$e^{[\ell-1]}(t_n^{[\ell]}) - e^{[\ell-1]}(t_{n-1}^{[\ell]}) - \int_{t_{n-1}^{[\ell]}}^{t_n^{[\ell]}} \delta^{[\ell-1]}(\tau)\, d\tau = \Delta t_n^{[\ell]} \Big[f\big(t_{n-1}^{[\ell]}, \eta^{[\ell-1]}(t_{n-1}^{[\ell]}) + e^{[\ell-1]}(t_{n-1}^{[\ell]})\big) - f\big(t_{n-1}^{[\ell]}, \eta^{[\ell-1]}(t_{n-1}^{[\ell]})\big) \Big],$$

where $\Delta t_n^{[\ell]} = t_n^{[\ell]} - t_{n-1}^{[\ell]}$. After some algebraic manipulation, one obtains

$$\eta_n^{[\ell]} = \eta_{n-1}^{[\ell]} + \Delta t_n^{[\ell]} \Big[f\big(t_{n-1}^{[\ell]}, \eta^{[\ell]}(t_{n-1}^{[\ell]})\big) - f\big(t_{n-1}^{[\ell]}, \eta^{[\ell-1]}(t_{n-1}^{[\ell]})\big) \Big] + \int_{t_{n-1}^{[\ell]}}^{t_n^{[\ell]}} f\big(\tau, \eta^{[\ell-1]}(\tau)\big)\, d\tau. \tag{4}$$

The integral in (4) is approximated using quadrature,

$$\int_{t_{n-1}^{[\ell]}}^{t_n^{[\ell]}} f\big(\tau, \eta^{[\ell-1]}(\tau)\big)\, d\tau \approx \sum_{i=1}^{|\vec{\mathcal{T}}_n^{[\ell]}|} \alpha_{n,i}^{[\ell-1]} f(\tau_i, \eta^{[\ell-1]}(\tau_i)), \quad \tau_i \in \vec{\mathcal{T}}_n^{[\ell]}, \tag{5}$$

where the set of quadrature nodes, $\vec{\mathcal{T}}_n^{[\ell]}$, for a first-order corrector satisfies

1. $|\vec{\mathcal{T}}_n^{[\ell]}| = \ell + 1$,
2. $\vec{\mathcal{T}}_n^{[\ell]} \subseteq \{t_n^{[\ell-1]}\}_{n=0}^{N^{[\ell-1]}}$,
3. $\min(\vec{\mathcal{T}}_n^{[\ell]}) \le t_{n-1}^{[\ell]}$,
4. $\max(\vec{\mathcal{T}}_n^{[\ell]}) \ge t_n^{[\ell]}$.

The quadrature weights, $\alpha_{n,i}^{[\ell-1]}$, are found by integrating the interpolating Lagrange polynomials exactly,

$$\alpha_{n,i}^{[\ell-1]} = \prod_{j=1, j\neq i}^{|\vec{\mathcal{T}}_n^{[\ell]}|} \int_{t_{n-1}^{[\ell]}}^{t_n^{[\ell]}} \frac{(t-\tau_j)}{(\tau_i-\tau_j)}\, dt, \quad \tau_i \in \vec{\mathcal{T}}_n^{[\ell]}. \tag{6}$$

The term $f\big(t_{n-1}^{[\ell]}, \eta^{[\ell-1]}(t_{n-1}^{[\ell]})\big)$ in (4) is approximated using Lagrange interpolation,

$$f\big(t_{n-1}^{[\ell]}, \eta^{[\ell-1]}(t_{n-1}^{[\ell]})\big) \approx \sum_{i=1}^{|\vec{\mathcal{T}}_n^{[\ell]}|} \gamma_{n,i}^{[\ell-1]} f\big(\tau_i, \eta^{[\ell-1]}(\tau_i)\big), \quad \tau_i \in \vec{\mathcal{T}}_n^{[\ell]}, \tag{7}$$

where the same set of nodes, $\vec{\mathcal{T}}_n^{[\ell]}$, for the quadrature is used for the interpolation. The interpolation weights are given by

$$\gamma_{n,i}^{[\ell-1]} = \prod_{j=1, j\neq i}^{|\vec{\mathcal{T}}_n^{[\ell]}|} \frac{(t_{n-1}^{[\ell]}-\tau_j)}{(\tau_i-\tau_j)}, \quad \tau_i \in \vec{\mathcal{T}}_n^{[\ell]}. \tag{8}$$

2.2. ***Adaptive step-size control.*** Adaptive step-size control is typically used to achieve a user-specified error tolerance with minimal computational effort by varying the step-sizes used by an IVP integrator. This is commonly done based on a local error estimate. It is also generally desirable that the step-size vary smoothly over the course of the integration. We review common techniques for estimating the local error, followed by algorithms for optimal step-size selection.

2.2.1. *Error estimators.* Two common approaches for estimating the local truncation error of a single-step IVP solver are through the use of Richardson extrapolation (commonly used within a step-size selection framework known as step doubling) and embedded Runge–Kutta pairs [15]. Step doubling is perhaps the more intuitive technique. The solution after each step is estimated twice: once as a full step and

once as two half steps. The difference between the two numerical estimates gives an estimate of the truncation error. For example, denoting the exact solution to IVP (1) at time $t_n + \Delta t$ as $y(t_n + \Delta t)$, the forward Euler step starting from the exact solution at time t_n and using a step-size of size Δt is

$$\eta_{1,n+1} = y(t_n) + \Delta t\ f(t_n, y_n),$$

and the forward Euler step using two steps of size $\Delta t/2$ is

$$\eta_{2,n+1} = \left(y(t_n) + \frac{\Delta t}{2}\ f(t_n, y_n)\right) + \frac{\Delta t}{2} f\left(t_n + \frac{\Delta t}{2}, y(t_n) + \frac{\Delta t}{2}\ f(t_n, y_n)\right).$$

Because forward Euler is a first-order method (and thus has a local truncation error of $\mathbb{O}(\Delta t^2)$), the two numerical approximations satisfy

$$\begin{aligned} y(t_n + \Delta t) &= \eta_{1,n+1} + (\Delta t)^2 \phi + \mathbb{O}(\Delta t^3) + \cdots, \\ y(t_n + \Delta t) &= \eta_{2,n+1} + 2\left(\frac{\Delta t}{2}\right)^2 \phi + \mathbb{O}(\Delta t^3) + \cdots, \end{aligned}$$

where a Taylor series expansion gives that ϕ is a constant proportional to $y''(t_n)$. The difference between the two numerical approximations gives an estimate for the local truncation error of $\eta_{2,n+1}$,

$$e_{n+1} = \eta_{2,n+1} - \eta_{1,n+1} = \frac{\Delta t^2}{2}\phi + \mathbb{O}(\Delta t^3).$$

An alternative approach to estimating the local truncation error is to use embedded RK pairs [11]. An s-stage Runge–Kutta method is a single-step method that takes the form

$$\eta_{n+1} = \eta_n + \Delta t \sum_{i=1}^{s} b_i k_i,$$

where

$$k_i = f\left(t_i + c_i h, \eta_n + \Delta t \sum_{j=1}^{s} a_{ij} k_j\right), \quad i = 1, 2, \ldots, s.$$

The idea is to find two single-step RK methods, typically one with order p and the other with order $p-1$, that share most (if not all) of their stages but have different quadrature weights. This is represented compactly in the extended Butcher tableau

$$\begin{array}{c|c} c & A \\ \hline & b \\ & \hat{b} \end{array}$$

Denoting the solution from the order-p method as

$$\eta^*_{n+1} = \eta_n + \Delta t \sum_{i=1}^{s} \hat{b}_i k_i, \tag{9a}$$

and the solution from the order-$(p-1)$ method as

$$\eta_{n+1} = \eta_n + \Delta t \sum_{i=1}^{s} b_i k_i, \tag{9b}$$

the error estimate is

$$e_{n+1} = \eta_{n+1} - \eta^*_{n+1} = \Delta t \sum_{i=1}^{s} (b_i - \hat{b}_i) k_i, \tag{9c}$$

which is $\mathbb{O}(\Delta t^p)$.

A third approach for approximating the local truncation error is possible within the deferred correction framework. We observe that in solving the error equation (3), one is in fact obtaining an approximation to the error. As discussed in Section 3.3, it can be shown that the approximate error after ℓ first-order corrections satisfies $o(\Delta t^{p_0+\ell+1})$. We shall see in Section 3.3 that this error estimate proves to be a poor choice for optimal step-size selection because in our formulation the time step selection for level ℓ does not allow for the refinement of time steps at earlier levels.

2.2.2. *Optimal step-size selection.* Given an error estimate from Section 2.2.1 for a step Δt, one would like to either accept or reject the step based on the error estimate and then estimate an optimal step-size for the next time step or retry the current step. Following [16], Algorithm 1 outlines optimal step-size selection given an estimate of the local truncation error. In lines 1–4, one computes a scaled error estimate. In line 5, an optimal time step is computed by scaling the current time step. In lines 6–10, a new time step is suggested; a more conservative step-size is suggested if the previous step was rejected.

3. RIDC with adaptive step-size control

There are numerous adaptive step-size control strategies that can be implemented within the RIDC framework. We consider three of them in this paper as well as discuss other strategies that are possible.

3.1. ***Adaptive step-size control: prediction level only.*** One simple approach to step-size control with RIDC is to perform adaptive step-size control on the prediction level only, e.g., using step doubling or embedded RK pairs as error estimators for the step-size control strategy. The subsequent correctors then use this grid unchanged (i.e., without performing further step-size control). With this strategy, corrector ℓ is

Input:
y_n: approximate solution at time t_n;
y_{n+1}: approximate solution at time t_{n+1};
e_{n+1}: error estimate for y_{n+1};
p: order of integrator;
m: number of ODEs;
`atol`, `rtol`: user specified tolerances;
`prev_rej`: flag that indicates whether the previous step was rejected;
$\alpha < 1$: safety factor;
$\beta > 1$: allowable change in step-size.

Output:
`accept_flag`: flag to accept or reject this step;
Δt_{new}: optimal time step

Set $a(i) = \max\{|y_n(i)|, |y_{n+1}(i)|\},\ i = 1, 2, \ldots, m.$
Compute $\tau(i) = \texttt{atol} + \texttt{rtol} * a(i),\ i = 1, 2, \ldots, m.$
Compute $\epsilon = \sqrt{\frac{\sum_{i=1}^{m} (e(i)/\tau(i))^2}{m}}.$
Compute $\Delta t_{opt} = \Delta t\ (\frac{1}{\epsilon})^{1/(p+1)}.$
if `prev_rej` **then**
    $\Delta t_{new} = \alpha \min\{\Delta t, \max\{\Delta t_{opt}, \Delta t/\beta\}\}$
else
    $\Delta t_{new} = \alpha \min\{\beta \Delta t, \max\{\Delta t_{opt}, \Delta t/\beta\}\}$
end
if $\epsilon > 1$ **then**
    `accept_flag` $= 1$
else
    `accept_flag` $= 0$
end

Algorithm 1: Optimal step-size selection algorithm. The approximate solution, the error estimate, and its order are provided as inputs. For the numerical experiments in Section 4, we fix $\alpha = 0.9$, $\beta = 10$.

lagged behind corrector $\ell - 1$ so that each node simultaneously computes an update on its level (after an initial startup period). This is illustrated graphically in Figure 1. In principle, near optimal parallel speedup is maintained with this approach provided the computational overhead for the RIDC method (i.e., the interpolation, quadrature, and linear combination of solutions) is small compared to the advance of predictor

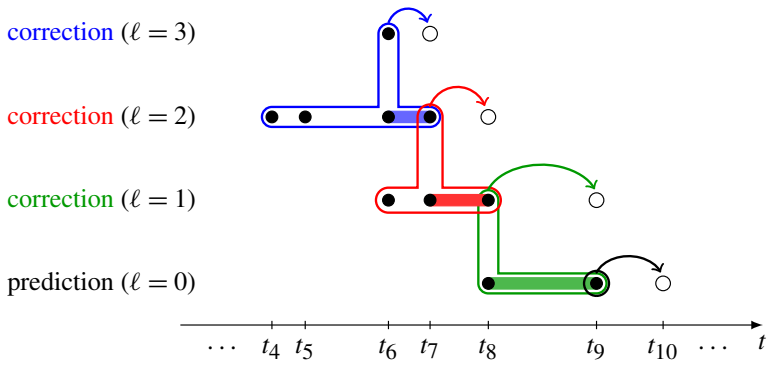


Figure 1. Schematic diagram of step-size control on the prediction level only. The filled circles denote previously computed and stored solution values at particular times. The corrections are run in parallel (but lagging in time) and the open circles indicate which values are being simultaneously computed. The stencil of points required by each level is shown by the "bubbles" surrounding certain grid points; the thick horizontal shading indicates the integrals needed in (4).

from t_n to t_{n+1}; in this implementation, a small memory footprint similar to [10] can be used. Additionally, an interpolation step is circumvented because the nodes are the same on each level. There are however a few potential drawbacks to this approach. First, it is not clear how to distribute the user-defined tolerance among the levels. Clearly, satisfying the user-specified tolerance on the prediction level defeats the purpose of the deferred correction approach. Estimating a reduced tolerance criterion may be possible a priori, but such an estimate would at present be ad hoc. Second, there is no reason to expect the corrector (4) should take the same steps to satisfy an error tolerance when computing a numerical approximation to the error equation (3).

3.2. ***Adaptive step-size control: all levels.*** A generalization of the above formulation is to utilize adaptive step-size control to solve the error equations (3) as well. The variant we consider is step doubling on all levels, where each predictor and corrector performs Algorithm 1; embedded RK pairs can also be used to estimate the error for step-size adaptivity on all levels. Intuitively, step-size control on every level gives more opportunity to detect and adapt to error than simply adapting using the (lowest-order) predictor. For example, this allows the corrector take a smaller step if necessary to satisfy an error tolerance when solving the error equation. Some drawbacks are: (i) an interpolation step is necessary because the nodes are generally no longer in the same locations on each level, (ii) more memory registers are required, and (iii) there is a potential loss of parallel efficiency because a corrector may be stalled waiting for an adequate stencil to become available to compute a quadrature approximation to the integral in (4). Another issue — both a potential benefit and a potential drawback — is the number of parameters that can be tuned

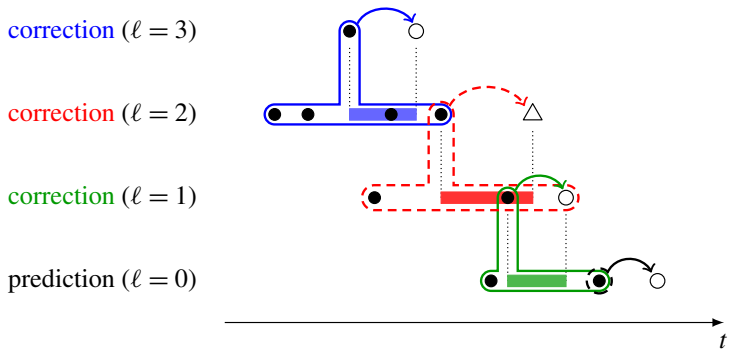


Figure 2. Schematic diagram of a scenario when step-size control is applied on all levels. Unlike in Figure 1, here each level has its own grid in time. Solid circles indicate particular times and levels where the solution is known. In this particular diagram, levels $\ell = 0, 1, 3$ are all able to advance simultaneously to the open circles. However, correction level $\ell = 2$ is unable to advance to the time indicated by the triangle symbol because correction level $\ell = 1$ has not yet computed far enough. The stencil of points required by each level is shown by the "bubbles" surrounding certain grid points; the thick horizontal shading indicates the integrals needed in (4). Note in particular that the dashed stencil includes a open circle at level $\ell = 1$ that is not yet computed.

for each problem. A discussion on the effect of tolerance choices for each level is provided in Section 4. One can in practice also tune step-size control parameters α, β, `atol`, and `rtol` for Algorithm 1 separately on each level. Figure 2 highlights that some nodes might not be able to compute an updated solution on their current level if an adequate stencil is not available to approximate the integral in (4) using quadrature. In this example, the level $\ell = 2$ correction is unable to proceed because it would require interpolated solution values not yet available from level $\ell = 1$, whereas the prediction level $\ell = 0$ and corrections $\ell = 1$ and $\ell = 3$ are all able to advance the solution by one step.

3.3. ***Adaptive step-size control: using the error equation.*** A third strategy one might consider is adaptive step-size control for the error equation (3) using the solution to the error equation itself as the error estimate. (One still uses step doubling or embedded RK pairs to obtain an error estimate for step-size control on the predictor equation (1).) At first glance, this looks promising provided the order of the integrator can be established because it is used to determine an optimal step-size. One would expect computational savings from utilizing available error information, as opposed to estimating it via step doubling or an embedded RK pair.

If first-order predictor and first-order correctors are used to construct the RIDC method, the analysis in [17] can be easily extended to the proposed RIDC methods with adaptive step-size control. We note that the numerical quadrature approximation given in (5) and the numerical interpolation given in (7) are accurate to the order $\mathcal{O}(\Delta t_n^{\ell+2})$; this is sufficient for the inductive proof in [17] to hold. Hence, one

can show that the method has a formal order of accuracy $\mathcal{O}(\Delta t^{\ell+2})$, where $\Delta t = \max_{n,\ell}(t_n^{[\ell]} - t_{n-1}^{[\ell]})$.

Although the formal order of accuracy can be established, using the error estimate from successive levels is a poor choice for optimal step-size selection. Consider step-size selection for level ℓ, time step $t_n^{[\ell]}$, using $\eta_n^{[\ell]} - \eta^{[\ell-1]}(t_n^{[\ell]})$ as the error estimator in Algorithm 1. The optimal step-size is chosen to control the local error estimate via the step-size $\Delta t_n^{[\ell]} = t_n^{[\ell]} - t_{n-1}^{[\ell]}$. However, the local error for the correctors generally contains contributions from the solutions at all the previous levels. The validity of the asymptotic local error expansion of the RIDC method in terms of $\Delta t_n^{[\ell]}$ requires that $\Delta t = \max_{n,\ell}(t_n^{[\ell]} - t_{n-1}^{[\ell]})$ be sufficiently small, and it is not normally possible to guarantee this in the context of an IVP solver. In other words, the step-size controller for a corrector at a given level cannot control the entire local error, and hence standard step-control strategies, which are predicated on the validity of error expansions in terms of only the step-size to be taken, cannot be expected to perform well. We present some numerical tests in Section 4.2.4 to illustrate the difficulties with using successive errors as the basis for step-size control.

3.4. ***Further discussion.*** There are many other strategies/implementation choices that affect the overall performance of the adaptive RIDC algorithm. Some have already been discussed in the previous section. We summarize some of the implementation choices that must be made:

- The choice of how to estimate the error of the discretization must be made. Three possibilities have already been mentioned: step doubling, embedded RK pairs, and solutions to the error equation (3). A combination of all three is also possible.
- If an IVP method with adaptive step-size control is used to solve (3), choices must be made as to how the tolerances and step-size control parameters, α and β, are to be chosen for each correction level.

We also list a few implementation details that should be considered when designing adaptive RIDC schemes.

- If adaptive step-size control is implemented on all levels, some correction levels may sit idle because the information required to perform the quadrature and interpolation in (4) is not available. This idle time adversely affects the parallel efficiency of the algorithm. One possibility to decrease this idle time is instead of taking an "optimal step" (as suggested by the step-size control routine), one could take a smaller step for which the quadrature and interpolation stencil is available. There is some flexibility in choosing exactly which points are used in the quadrature stencil, and it might also be possible to choose a stencil to minimize the time that correction levels are sitting idle.

- Because values are needed from lower-order correction levels, the storage required by a RIDC scheme depends on when values can be overwritten (see, e.g., the stencils in Figures 1 and 2). Thus to avoid increasing the storage requirements, the prediction level and each correction level should not be allowed to get too far ahead of higher correction levels. Although this is also the case for the nonadaptive RIDC schemes [10; 6], if adaptive step-size control is implemented on all levels (Figure 2), the memory footprint is likely to increase. Some consideration should thus be given to a potential trade-off between parallel efficiency and the overall memory footprint of the scheme.
- It is important to reduce round-off error when computing the quadrature weights (6) and the interpolation weights (8). This can be done by through careful scaling and control of the order of the floating-point operations [3].
- If one wishes to use higher-order correctors and predictors to construct RIDC integrators, we note that the convergence analysis in [7; 8; 5] only holds for uniform steps. A nonuniform mesh introduces discrete "roughness" (see [8]); hence, an increase of only one order per correction level is guaranteed even though a high-order method is used to solve (3).
- RIDC methods necessarily incur computational overhead costs, for example, quadrature evaluation (5), interpolation evaluation (7), and the combination of these components in (4). Parallel speedup can only be expected if the computational overhead is small compared to the advance of predictor from t_n to t_{n+1}.

Additionally, the RIDC framework, by construction, solves a series of error equations to generate a successively more accurate solution. This framework can be potentially be exploited to generate *order-adaptive* RIDC methods. For example, one might control the number of corrector levels adaptively based on an error estimate.

4. Numerical examples

We focus on the solutions to three nonlinear IVPs. The first is presented in [1]; we refer to it as the Auzinger IVP:

$$\begin{cases} y_1' = -y_2 + y_1(1 - y_1^2 - y_2^2), \\ y_2' = y_1 + 3y_2(1 - y_1^2 - y_2^2), \\ y(0) = (1, 0)^T, \quad t \in [0, 10], \end{cases} \tag{AUZ}$$

that has the analytic solution $y(t) = (\cos t, \sin t)^T$.

The second is the IVP associated with the Lorenz attractor:

$$\begin{cases} y_1' = \sigma(y_2 - y_1), \\ y_2' = \rho y_1 - y_2 - y_1 y_3, \\ y_3' = y_1 y_2 - \beta y_3, \\ y(0) = (1, 1, 1)^T, \quad t \in [0, 1]. \end{cases} \tag{LORENZ}$$

For the parameter settings $\sigma = 10$, $\rho = 28$, $\beta = 8/3$, this system is highly sensitive to perturbations, and an IVP integrator with adaptive step-size control may be advantageous.

The third is the restricted three-body problem from [15]; we refer to it as the Orbit IVP:

$$\begin{cases} y_1'' = y_1 + 2y_2' - \mu' \dfrac{y_1 + \mu}{D_1} - \mu \dfrac{y_1 - \mu'}{D_2}, \\ y_2'' = y_2 - 2y_1' - \mu' \dfrac{y_2}{D_1} - \mu \dfrac{y_2}{D_2}, \\ D_1 = ((y_1 + \mu)^2 + y_2^2)^{3/2}, \ D_2 = ((y_1 - \mu')^2 + y_2^2)^{3/2}, \\ \mu = 0.012277471, \ \mu' = 1 - \mu. \end{cases} \tag{ORBIT}$$

Choosing the initial conditions

$$y_1(0) = 0.994, \quad y_1'(0) = 0, \quad y_2(0) = 0,$$
$$y_2'(0) = -2.00158510637908252240537862224,$$

gives a periodic solution with period $t_{\text{end}} = 17.065216560159625588917206249$.

We now present numerical evidence to demonstrate that:

1. RIDC integrators with nonuniform step-sizes converge and achieve their designed orders of accuracy.
2. RIDC methods with adaptive step-size constructed using step doubling (on the prediction level only) and embedded RK error estimators (on the prediction level only) converge.
3. RIDC methods with adaptive step-size control based on step doubling to estimate the local error on the prediction and correction levels converge; however, the step-sizes selected are poor (many rejected steps), even for the smooth Auzinger problem.
4. RIDC methods with adaptive step-size control based on step doubling to estimate the local error on the prediction level but using the solution to the error equation for step-size control results is problematic.

The numerical examples chosen are canonical problems designed to illustrate the step-size adaptivity properties of the RIDC methods. Because the computational

overhead is significant compared to the advance of an Euler solution from time t_n to t_{n+1}, a runtime analysis does not reveal parallel speedup for any of these examples. Whereas the number of function evaluations is an effective parameter for comparing algorithms, we need a different metric to compare a parallel algorithm to a sequential algorithm. Where appropriate, we tabulate the number of *sets of concurrent function evaluations* as a proxy for measuring parallel speedup when the function evaluation costs dominate. A set of concurrent function evaluations consists of function evaluations that can be evaluated in parallel.

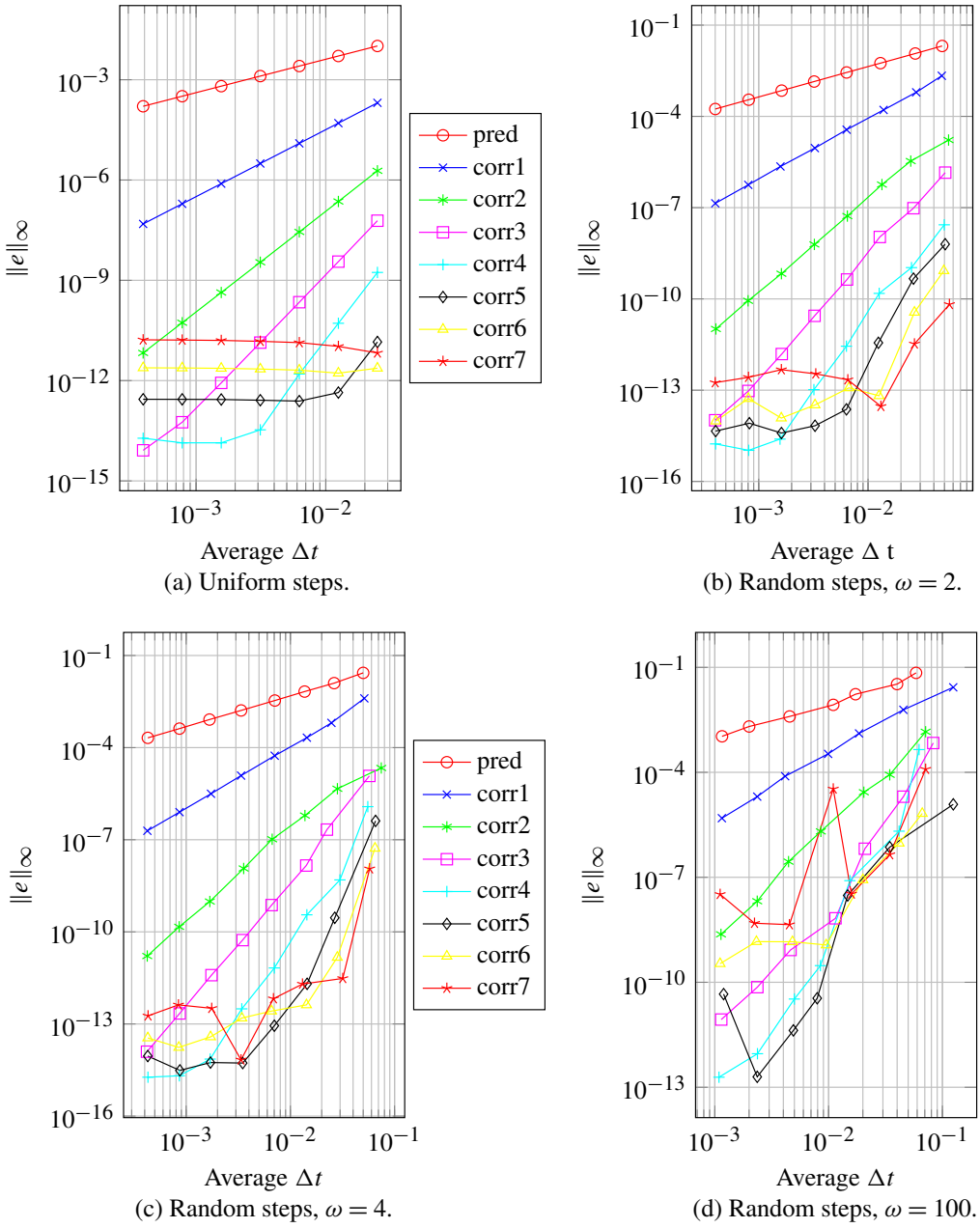
4.1. ***RIDC with nonuniform step-sizes.*** For our first numerical experiment, we demonstrate that RIDC integrators with nonuniform step-sizes converge and achieve their design orders of accuracy. Figure 3 shows the classical convergence study (error as a function of mean step-size) for the RIDC integrator applied to (AUZ). Figure 3(a) shows the convergence of RIDC integrators with uniform step-sizes; Figure 3(b)–(d) shows the convergence of RIDC integrators when random step-sizes are chosen. The random step-sizes are chosen so that

$$\Delta t_n^{[\ell]} \in \left[\frac{1}{\omega}\Delta t_{n-1}^{[\ell]}, \omega \Delta t_{n-1}^{[\ell]}\right], \quad \omega \in \mathbb{R},$$

where ω controls how rapidly a step-size is allowed to change. The figures show that RIDC integrators with nonuniform step-sizes achieve their designed order of accuracy (each additional correction improves the order of accuracy by one), at least up to order 6. In Figure 3 (corresponding to RIDC with uniform step-sizes), we observe that the error stagnates at a value significantly larger than machine precision. This is likely due to numerical issues associated with quadrature on equispaced nodes [14]. We note that $\omega = 1$ gives the uniformly distributed case. We also observe that as the ratio of the largest to the smallest cell increases, the performance of higher-order RIDC methods degrades, likely due to round-off error associated with calculating the quadrature and interpolation weights.

Figure 4 shows the convergence study (error as a function of mean step-size) for (LORENZ). The reference solution is computed using an RK-45 integrator with a fine time step. Similar observations can be made that RIDC methods with nonuniform step-sizes converge with their designed orders of accuracy (at least up to order 6).

4.2. ***Adaptive RIDC.*** We study four different variants of RIDC methods with adaptive step-size control: (i) step doubling is used for adaptive step-size control on the prediction level only (Section 4.2.1); (ii) an embedded RK pair is used for adaptive step-size control on the prediction level only (Section 4.2.2); (iii) step doubling is used for adaptive step-size control on the prediction and correction levels (Section 4.2.3); and (iv) step doubling is used for adaptive step-size control

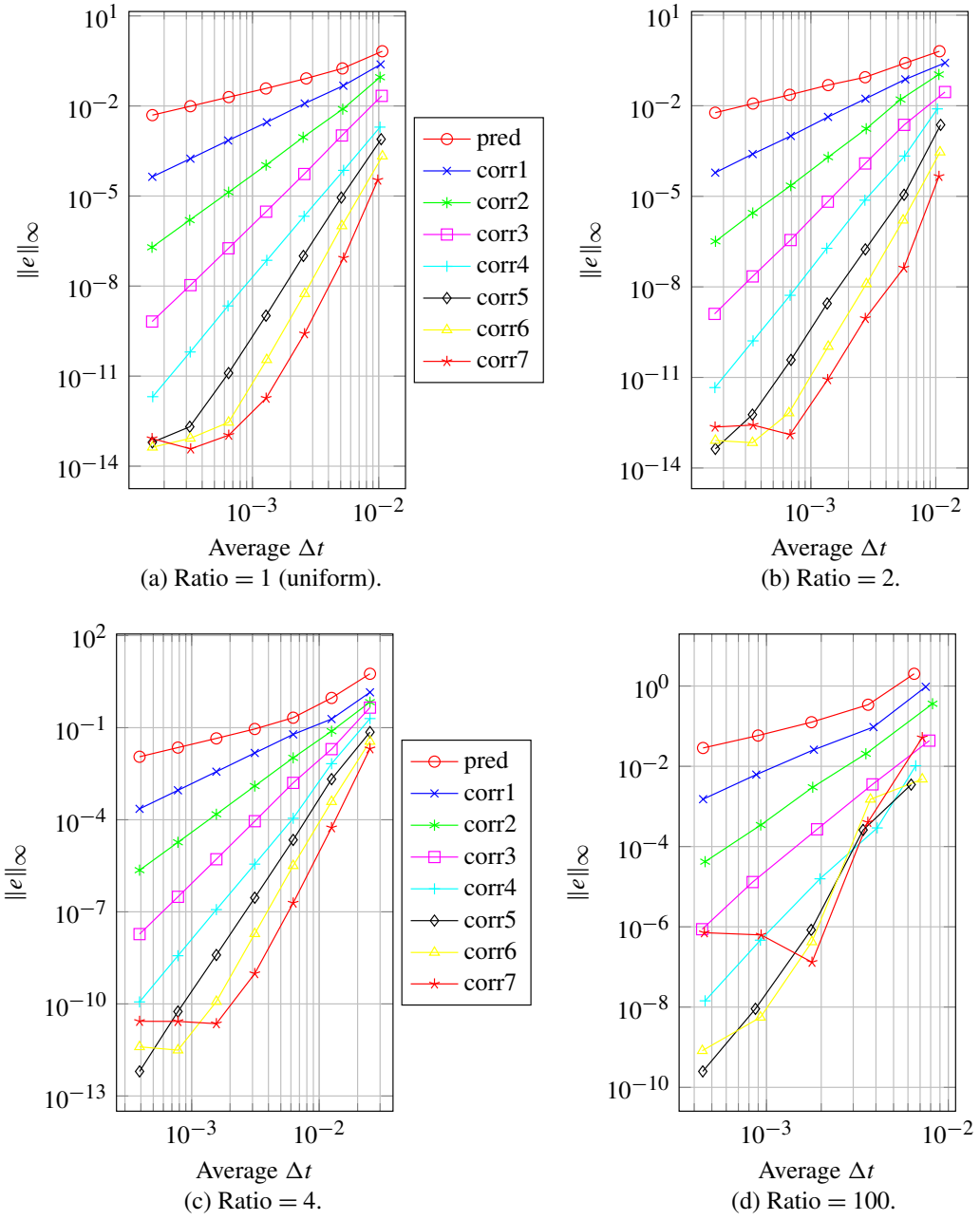


(a) Uniform steps.
(b) Random steps, $\omega = 2$.
(c) Random steps, $\omega = 4$.
(d) Random steps, $\omega = 100$.

Figure 3. Auzinger IVP: The design order is illustrated for the RIDC methods.

on the prediction level, and the computed errors from the error equation (3) are used for adaptive step-size control on the correction levels.

4.2.1. *Step doubling on the prediction level only.* In this numerical experiment, we solve the orbit problem (ORBIT) using a fourth-order RIDC method (constructed using forward Euler integrators), and adaptive step-size control on the prediction level only, where step doubling is used to provide the error estimate. As shown in

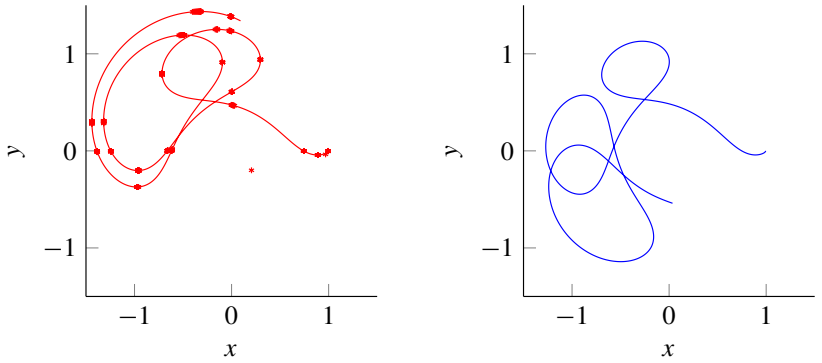


(a) Ratio = 1 (uniform). (b) Ratio = 2.

(c) Ratio = 4. (d) Ratio = 100.

Figure 4. Lorenz IVP: the design order is illustrated for the RIDC methods.

Figure 5, successive correction loops are able to reduce the error in the solution and recover the desired orbit. The red circles in Figure 5(a) indicate rejected steps. Figure 6(a) shows that RIDC with step doubling only on the prediction level converges as the tolerance is reduced. In this experiment, the RIDC integrator is *reset* after every 100 accepted steps. By "reset" [10], we mean that the highest-order solution after every 100 steps is used as an initial condition to reinitialize the



(a) Prediction. (b) First correction.

(c) Second correction. (d) Third correction.

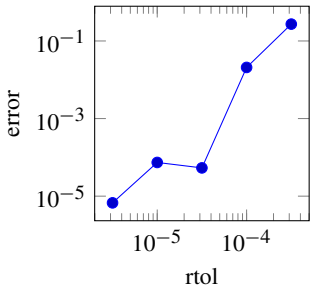
Figure 5. Orbit problem: although the prediction level gives a highly inaccurate solution, successive correction loops are able to reduce the error and produce the desired orbit. The red circles on the prediction level (a) indicate rejected steps.

provisional solution; e.g., instead of solving (1), one solves a sequence of problems

$$\begin{cases} y'(t) = f(t, y), \quad t \in [t_{100(i-1)}, \min(b, t_{100i})], \\ y(t_{100(i-1)}) = \eta^{[P-1]}_{100(i-1)}, \end{cases}$$

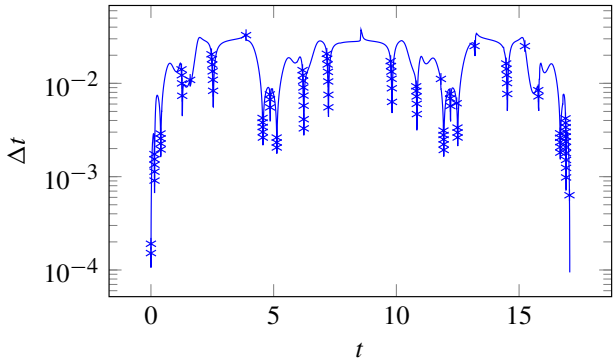
if $(L-1)$ correctors are applied and $\eta_0^{[L-1]} = y_a$. The time steps chosen by the RIDC integrator with resets performed every 100 and 400 steps are shown in Figure 6(b) and (c).

In Figure 6(b), $\Delta t_{\min} = 1.06 \times 10^{-4}$. If a nonadaptive fourth-order RIDC method was used with $\Delta t_{\min}$, 160814 uniform time steps would have been required. By adaptively selecting the time steps for this example and tolerance, the adaptive RIDC method required approximately one one-hundredth of the functional evaluations, corresponding to a one hundred-fold speedup. The effective parallel speedup can be computed by taking the ratio of the total number of function evaluations required and the number of sets of concurrent function evaluations required. For the computation in Figure 6(b) where a reset is performed after every 100 steps, the parallel speedup

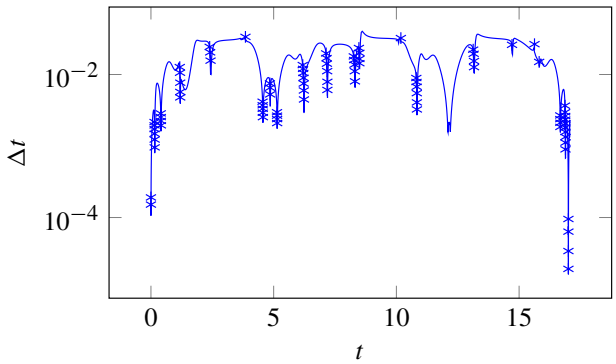


rtol	atol	error	naccept	nreject
$10^{-3.5}$	$10^{-6.5}$	$2.72\cdot10^{-1}$	1456	99
$10^{-4.0}$	$10^{-7.0}$	$2.08\cdot10^{-2}$	2650	81
$10^{-4.5}$	$10^{-7.5}$	$5.35\cdot10^{-5}$	4730	68
$10^{-5.0}$	$10^{-8.0}$	$7.39\cdot10^{-5}$	8436	42
$10^{-5.5}$	$10^{-8.5}$	$6.72\cdot10^{-6}$	15031	10

(a) Convergence study.



(b) Adaptive step-sizes selected (reset every 100 steps).



(c) Adaptive step-sizes selected (reset every 400 steps).

Figure 6. Orbit problem: (a) convergence of a fourth-order RIDC method constructed with forward Euler integrators and adaptive step-size control on the prediction level (using step doubling). Convergence is measured relative to the exact solution as the tolerance is decreased. A reset is performed after every 100 accepted steps for this convergence study. In (b), the step-sizes selected for $\texttt{rtol} = 10^{-3.5}$ and $\texttt{atol} = 10^{-6.5}$ are displayed as the solid curve and rejected steps as $\times$s; a reset is performed after every 100 steps. In (c), the reset is performed after every 400 steps. Observe that although the number of rejected steps increases, the overall Δt chosen remains qualitatively similar.

(if four processors are available) can be computed using

$$\frac{(1456 \times 5) + 99}{(1456 \times 2) + (14 \times 6) + 99} = 2.38.$$

The numerator consists of the total number of function evaluations arising from the number of steps taken and the computation of the error estimate using step doubling and the number of function evaluations arising from the rejected steps. The denominator consists of the number of concurrent function evaluations (including startup costs for the RIDC method). Note that three of the processors sit idle while that step doubling computation is being processed. The parallel speedup can be improved if more levels are chosen, or if the number of resets are reduced. If a reset is performed after every 400 steps (Figure 6(c)), the parallel speedup is

$$\frac{(1591 \times 5) + 88}{(1591 \times 2) + (4 \times 6) + 88} = 2.44.$$

4.2.2. *Embedded RK on the prediction level only.* In this numerical experiment, we repeat the orbit problem (ORBIT) using a fourth-order RIDC method constructed again using forward Euler integrators, but the step-size adaptivity on the prediction level uses a Heun–Euler embedded RK pair. This simple scheme combines Heun's method, which is second order, with the forward Euler method, which is first order. Figure 7(a) shows the convergence of this adaptive RIDC method as the tolerance is reduced. As the previous example, the RIDC integrator is reset after every 100 accepted steps for the convergence study. In Figure 7(b) and (c), we show the time steps chosen by the RIDC integrator with resets performed after 100 or 400 steps, respectively.

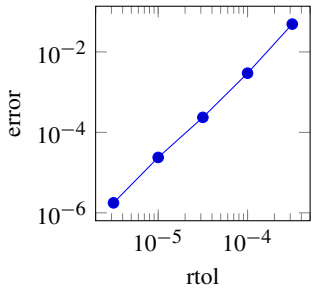
For the computation in Figure 7(b) where a reset is performed after every 100 steps, the parallel speedup (if four processors are available) is

$$\frac{(2441 \times 5) + 60}{(2441 \times 2) + (24 \times 6) + 60} = 2.41.$$

If a reset is performed after every 400 steps (Figure 7(c)), the parallel speedup is

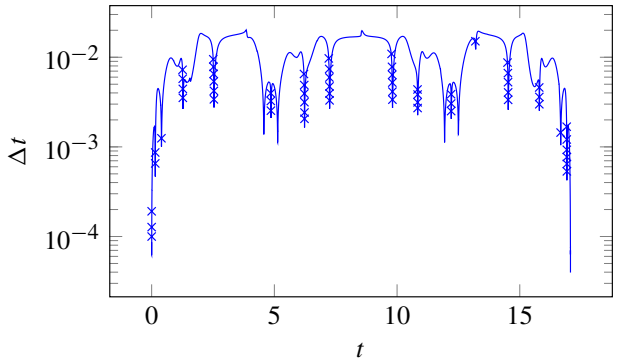
$$\frac{(2276 \times 5) + 80}{(2276 \times 2) + (5 \times 6) + 80} = 2.46.$$

Not surprisingly, the time steps chosen by the RIDC method are dependent on the specified tolerances and the error estimator (and consequently the integrators used to obtain a provisional solution to (1)) used for the control strategy. One can easily construct a RIDC integrator using higher-order embedded RK pairs to solve for a provisional solution to (1), and then use the forward Euler method to solve the error equation (3) on subsequent levels. For example, Figure 8 shows the step-sizes chosen when the Bogacki–Shampine method [2] (a 3(2) embedded RK pair) and the popular Runge–Kutta–Fehlberg 4(5) pair [13] are used to compute the provisional solution (and error estimate) for the RIDC integrator. The same

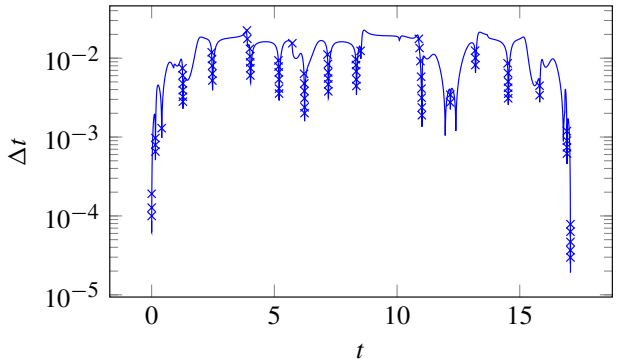


rtol	atol	error	naccept	nreject
$10^{-3.5}$	$10^{-6.5}$	$4.91 \cdot 10^{-2}$	2082	93
$10^{-4.0}$	$10^{-7.0}$	$2.96 \cdot 10^{-3}$	3754	71
$10^{-4.5}$	$10^{-7.5}$	$2.36 \cdot 10^{-4}$	6703	50
$10^{-5.0}$	$10^{-8.0}$	$2.28 \cdot 10^{-5}$	11945	20
$10^{-5.5}$	$10^{-8.5}$	$1.77 \cdot 10^{-6}$	21277	10

(a) Convergence study.



(b) Adaptive step-sizes selected (reset every 100 steps).



(c) Adaptive step-sizes selected (reset every 400 steps).

Figure 7. Orbit problem: (a) convergence of a fourth-order RIDC method constructed with forward Euler integrators and adaptive step-size control on the prediction level (using an embedded RK pair to estimate the error). Convergence is measured relative to the exact solution as the tolerance is decreased. A reset is performed after every 100 accepted steps for this convergence study. In (b), the step-sizes selected for $\texttt{rtol} = 10^{-3.5}$ and $\texttt{atol} = 10^{-6.5}$ are displayed as the solid curve and rejected steps as $\times$s; a reset is performed after every 100 steps. In (c), the reset is performed after every 400 steps.

tolerance of $\texttt{rtol} = 10^{-3.5}$ is used to generate both graphs. As the order and accuracy of the predictor increases, one can take larger time steps. For this example, using higher-order embedded RK pairs as step-size control mechanisms for RIDC methods result in less variations in time steps.

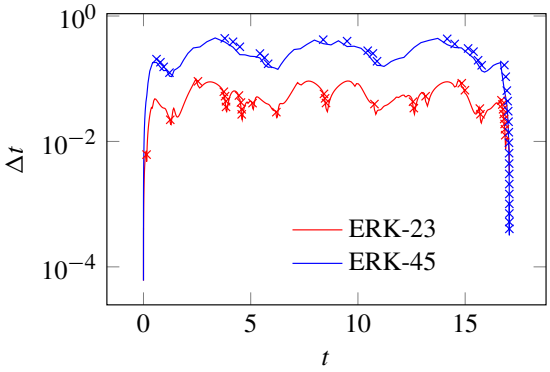
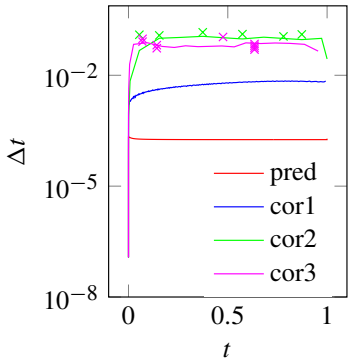


Figure 8. Step-sizes selected by RIDC methods constructed using a Bogacki–Shampine method, a 3(2) embedded pair (red) and the Runge–Kutta–Fehlberg 4(5) pair. Rejected steps are indicated with $\times$s.

4.2.3. *Step doubling on all levels.* As mentioned in Section 3.2, it might be advantageous to use adaptive step-size control when solving the error equations. This affords a myriad of parameters that can be used to tune the step-size control mechanism. In this set of numerical experiments, we explore how the choice of tolerances for the prediction/correction levels affect the step-size selection.

We first solve the Auzinger IVP using step doubling on all the levels, i.e., both predictor and corrector levels. In Figure 9, we show the computed step-sizes when we naively choose the same tolerances on each level. As expected, the predictor has to take many steps (to satisfy the stringent user-supplied tolerance), whereas life is easy for the correctors. The effective parallel speedup is

$$\frac{(5479+196+19+24)\times 2+15}{(5481\times 2)+15} = 1.04.$$



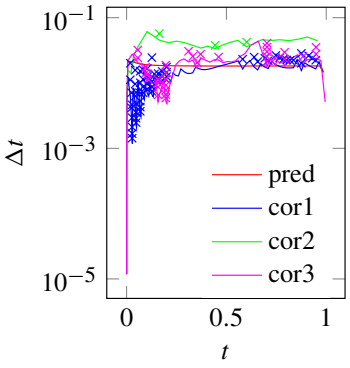
ℓ	rtol	atol	error	naccept	nreject
0	10^{-8}	10^{-10}	$2.028\cdot10^{-5}$	5479	0
1	10^{-8}	10^{-10}	$8.793\cdot10^{-7}$	196	0
2	10^{-8}	10^{-10}	$2.618\cdot10^{-8}$	19	6
3	10^{-8}	10^{-10}	$1.486\cdot10^{-6}$	24	9

Figure 9. Auzinger IVP: step-size control is implemented on all prediction and correction levels. The same tolerances are used for each level. As expected, the predictor has a hard time (forward Euler must satisfy a stringent tolerance); on the other hand, life is easy for the correctors. Rejected steps are indicated with $\times$s. For this set of tolerances, 5481 sets of concurrent function evaluations are needed.

In principle, the correctors are not even needed. Equally important to note is that the error *increases* after the last correction loop. This might seem surprising at first glance but ultimately may not unreasonable because the steps selected to solve the third correction are not based on the solution to the error equation but rather the original IVP.

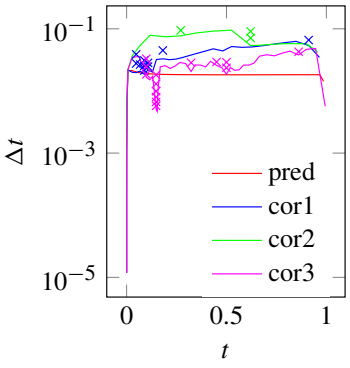
Instead of naively choosing the same tolerances on each level, we now change the tolerance at each level, as described in Figure 10. By making this simple change, the number of accepted steps on each level are now on the same order of magnitude. Not surprisingly, the predictor still selects good steps. Interestingly in Figure 10(a), the first correction is "noisy", especially initially. For this set of tolerances, the effective parallel speedup is

$$\frac{(58+7+30+61)\times 2+(52+7+24)}{(135\times 2)+(52+7+24)} = 1.52.$$



ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$2.026{\cdot}10^{-3}$	58	0
1	$1{\cdot}10^{-6}$	$1{\cdot}10^{-8}$	$6.945{\cdot}10^{-5}$	78	52
2	$1{\cdot}10^{-8}$	$1{\cdot}10^{-10}$	$1.265{\cdot}10^{-7}$	30	7
3	$1{\cdot}10^{-10}$	$1{\cdot}10^{-12}$	$9.579{\cdot}10^{-8}$	61	24

(a) Set 1 of tolerances.



ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$2.026{\cdot}10^{-3}$	58	0
1	$1{\cdot}10^{-5}$	$1{\cdot}10^{-7}$	$1.805{\cdot}10^{-4}$	29	12
2	$1{\cdot}10^{-7}$	$1{\cdot}10^{-9}$	$1.172{\cdot}10^{-6}$	20	6
3	$1{\cdot}10^{-9}$	$1{\cdot}10^{-11}$	$7.216{\cdot}10^{-7}$	39	11

(b) Set 2 of tolerances.

Figure 10. Auzinger IVP: different tolerances at each level. With the first set of tolerances, the step-size controller for the predictor is well behaved, as it is for the second and third correctors. The step-size controller for the first corrector however is noisy. 135 sets of concurrent function evaluations are needed to generate (b). With the second set of tolerances, the step-size controller for all correctors is reasonably well behaved. Here, 64 sets of concurrent function evaluations are needed.

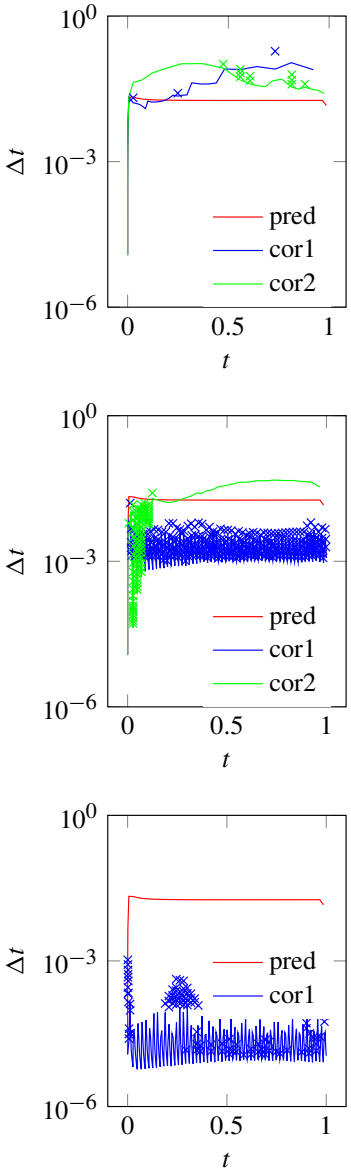
By picking a different set of tolerances, we can eliminate the noise, as shown in Figure 10(b). For this set of tolerances, the parallel speedup is

$$\frac{(58+24+20+39)\times 2+(12+6+11)}{(64\times 2)+(12+6+11)} = 1.98.$$

4.2.4. *Using solutions from the error equation.* As mentioned in Section 3.3, using the solution from the error equation (3) as the local error estimate for step-size control on a given level is potentially problematic because the step-size controller can only control the local error introduced on that level whereas the true local error generally contains contributions from all previous levels. For completeness, we present the results of this adaptive RIDC formulation applied to the Orbit problem (Figure 12) and the Auzinger problem (Figure 11). Step doubling is used for step-size adaptivity on the predictor level, solutions from the error equation are used to control step-sizes for the corrector levels. For the Auzinger problem, we observe in the top figure that if the tolerances are held fixed on each level, each correction level improves the solution. If the tolerance is reduced slightly on each level, the step-size controller gives a poor step-size selection (many rejected steps), even for this smoothly varying problem. For the Orbit IVP, Figure 12 shows that the corrector improves the solution if the tolerances are held fixed at all levels; however the corrector requires *many* steps. A second correction loop was not attempted. Reducing the tolerance for the first corrector resulted in inordinately many rejected steps.

5. Conclusions

In this paper, we formulated RIDC methods that incorporate local error estimation and adaptive step-size control. Several formulations were discussed in detail: (i) step doubling on the prediction level, (ii) embedded RK pairs on the prediction level, (iii) step doubling on the prediction and error levels, and (iv) step doubling for the prediction level but using the solution from the error equation for step-size control; other formulations are also alluded to. A convergence theorem from [17] can be extended to RIDC methods that use adaptive step-size control on the prediction level. Numerical experiments demonstrate that RIDC methods with nonuniform steps converge as designed and illustrate the type of behavior that might be observed when adaptive step-size control is used on the prediction and correction levels. Based on our numerical study, we conclude that adaptive step-size control on the prediction level is viable for RIDC methods. In a practical application where a user gives a specified tolerance, this prescribed tolerance must be transformed to a specific tolerance that is fed to the predictor.



ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$2.031{\cdot}10^{-3}$	59	0
1	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$7.249{\cdot}10^{-4}$	33	3
2	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$6.513{\cdot}10^{-6}$	26	10

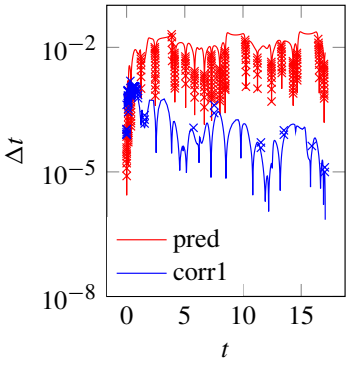
ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$2.031{\cdot}10^{-3}$	59	0
1	$1{\cdot}10^{-5}$	$1{\cdot}10^{-7}$	$1.063{\cdot}10^{-5}$	657	305
2	$1{\cdot}10^{-6}$	$1{\cdot}10^{-8}$	$9.446{\cdot}10^{-8}$	75	76

ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-6}$	$2.031{\cdot}10^{-3}$	59	0
1	$1{\cdot}10^{-7}$	$1{\cdot}10^{-9}$	$1.178{\cdot}10^{-7}$	60571	94

Figure 11. Auzinger problem: step doubling on prediction level, using successive levels for error estimation for step control on the error equation. Step-size controller for the corrector is noisy.

Acknowledgments

This publication was based on work supported in part by award no. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST), AFRL and AFOSR under contract and grants FA9550-12-1-0455, NSF grant number DMS-0934568, NSERC grant number RGPIN-228090-2013, and the Oxford Center for Collaborative and Applied Mathematics (OCCAM).



ℓ	rtol	atol	error	naccept	nreject
0	$1{\cdot}10^{-4}$	$1{\cdot}10^{-4}$	2.405	2261	230
1	$1{\cdot}10^{-4}$	$1{\cdot}10^{-4}$	$7.234{\cdot}10^{-1}$	475181	84

Figure 12. Orbit problem: step doubling on prediction level, using successive levels for error estimation for step control on the error equation.

References

[1] W. Auzinger, H. Hofstätter, W. Kreuzer, and E. Weinmüller, *Modified defect correction algorithms for ODEs, I: general theory*, Numer. Algorithms **36** (2004), no. 2, 135–155. MR 2005h:65096

[2] P. Bogacki and L. F. Shampine, *A* 3(2) *pair of Runge–Kutta formulas*, Appl. Math. Lett. **2** (1989), no. 4, 321–325. MR 1025845 Zbl 0705.65055

[3] B. Bradie, *A friendly introduction to numerical analysis: with C and MATLAB materials on website*, Pearson Education, Upper Saddle River, NJ, 2006.

[4] A. Christlieb, A. Melfi, and B. Ong, *Distributed parallel semi-implicit time integrators*, preprint, 2012. arXiv 1209.4297v1

[5] A. Christlieb, M. Morton, B. Ong, and J.-M. Qiu, *Semi-implicit integral deferred correction constructed with additive Runge–Kutta methods*, Commun. Math. Sci. **9** (2011), no. 3, 879–902. MR 2865808 Zbl 1271.65109

[6] A. Christlieb and B. Ong, *Implicit parallel time integrators*, J. Sci. Comput. **49** (2011), no. 2, 167–179. MR 2012k:65067 Zbl 1243.65076

[7] A. Christlieb, B. Ong, and J.-M. Qiu, *Comments on high-order integrators embedded within integral deferred correction methods*, Commun. Appl. Math. Comput. Sci. **4** (2009), 27–56. MR 2010e:65094 Zbl 1167.65389

[8] ———, *Integral deferred correction methods constructed with high order Runge–Kutta integrators*, Math. Comp. **79** (2010), no. 270, 761–783. MR 2011c:65122 Zbl 1209.65073

[9] A. J. Christlieb, R. D. Haynes, and B. W. Ong, *A parallel space-time algorithm*, SIAM J. Sci. Comput. **34** (2012), no. 5, C233–C248. MR 3023735 Zbl 1259.65143

[10] A. J. Christlieb, C. B. Macdonald, and B. W. Ong, *Parallel high-order integrators*, SIAM J. Sci. Comput. **32** (2010), no. 2, 818–835. MR 2011g:65105 Zbl 1211.65089

[11] J. R. Dormand and P. J. Prince, *A family of embedded Runge–Kutta formulae*, J. Comput. Appl. Math. **6** (1980), no. 1, 19–26. MR 81g:65098 Zbl 0448.65045

[12] A. Dutt, L. Greengard, and V. Rokhlin, *Spectral deferred correction methods for ordinary differential equations*, BIT **40** (2000), no. 2, 241–266. MR 2001e:65104 Zbl 0959.65084

[13] E. Fehlberg, *Low-order classical Runge–Kutta formulas with step size control and their application to some heat transfer problems*, technical report, R-315, NASA, 1969.

[14] S. Güttel and G. Klein, *Efficient high-order rational integration and deferred correction with equispaced data*, *Electron. Trans. Numer. Anal.* **41** (2014), 443–464.

[15] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving ordinary differential equations, I: Nonstiff problems*, 2nd ed., Springer Series in Computational Mathematics, no. 8, Springer, Berlin, 1993. MR 94c:65005

[16] E. Hairer and G. Wanner, *Solving ordinary differential equations, II: Stiff and differential-algebraic problems*, 2nd ed., Springer Series in Computational Mathematics, no. 14, Springer, Berlin, 1996. MR 97m:65007 Zbl 0859.65067

[17] Y. Xia, Y. Xu, and C.-W. Shu, *Efficient time discretization for local discontinuous Galerkin methods*, Discrete Contin. Dyn. Syst. Ser. B **8** (2007), no. 3, 677–693. MR 2008e:65307 Zbl 1141.65076

Received October 9, 2013. Revised December 10, 2014.

ANDREW J. CHRISTLIEB: *Department of Mathematics, Michigan State University, East Lansing, 48823, United States*

COLIN B. MACDONALD: macdonald@maths.ox.ac.uk
Mathematical Institute, Oxford University, Oxford, OX2 6GG, United Kingdom

BENJAMIN W. ONG: ongbw@mtu.edu
Department of Mathematics, Michigan Technological University, Houghton, MI 49931, United States

RAYMOND J. SPITERI: spiteri@cs.usask.ca
Department of Computer Science, University of Saskatchewan, Saskatoon S7N 5C9, Canada

Communications in Applied Mathematics and Computational Science

msp.org/camcos

See inside back cover or msp.org/camcos for submission instructions.

The subscription price for 2015 is US $85/year for the electronic version, and $120/year (+$15, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues from the last three years and changes of subscribers address should be sent to MSP.

Communications in Applied Mathematics and Computational Science (ISSN 2157-5452 electronic, 1559-3940 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

CAMCoS peer review and production are managed by EditFLOW® from MSP.

PUBLISHED BY

mathematical sciences publishers

nonprofit scientific publishing

http://msp.org/

Communications in Applied Mathematics and Computational Science

vol. 10 no. 1 2015

1559-3940(2015)10:1;1-O

Appl. Comput. Harmon. Anal. 37 (2014) 531–544

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

Phase retrieval for sparse signals

Yang Wang [a,1], Zhiqiang Xu [b,*,2]

[a] *Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA*
[b] *LSEC, Inst. Comp. Math., Academy of Mathematics and System Science, Chinese Academy of Sciences, Beijing 100091, China*

ARTICLE INFO

Article history:
Received 10 October 2013
Received in revised form 9 April 2014
Accepted 21 April 2014
Available online 28 April 2014
Communicated by Radu Balan

Keywords:
Signal recovery
Phase retrieval
Compressed sensing
Null space property

ABSTRACT

The aim of this paper is to build up the theoretical framework for the recovery of sparse signals from the magnitude of the measurements. We first investigate the minimal number of measurements for the success of the recovery of sparse signals from the magnitude of samples. We completely settle the minimality question for the real case and give a bound for the complex case. We then study the recovery performance of the ℓ_1 minimization for the sparse phase retrieval problem. In particular, we present the null space property which, to our knowledge, is the first sufficient and necessary condition for the success of ℓ_1 minimization for k-sparse phase retrieval.

1. Introduction

The theory of compressive sensing has generated enormous interest in recent years. The goal of compressive sensing is to recover a sparse signal from its linear measurements, where the number of measurements is much smaller than the dimension of the signal, see e.g. [4–6,12]. The aim of this paper is to study the problem of compressive sensing without the phase information. In this problem the goal is to recover a sparse signal from the magnitude of its linear samples.

Recovering a signal from the magnitude of its linear samples, commonly known as *phase retrieval* or *phaseless reconstruction*, has gained considerable attention in recent years [1,2,7,8]. It has important application in X-ray imaging, crystallography, electron microscopy, coherence theory and other applications. In many applications the signals to be reconstructed are sparse. Thus it is natural to extend compressive sensing to the phase retrieval problem.

* Corresponding author.
E-mail addresses: ywang@math.msu.edu (Y. Wang), xuzq@lsec.cc.ac.cn (Z. Xu).

[1] Yang Wang was supported in part by the National Science Foundation grant DMS-1043034 and AFOSR grant FA9550-12-1-0455.
[2] Zhiqiang Xu was supported by NSFC grants 11171336, 11331012, 11021101 and National Basic Research Program of China (973 Program 2010CB832702).

http://dx.doi.org/10.1016/j.acha.2014.04.001
1063-5203/

We first introduce the notation and briefly describe the mathematical background of the problem. Let $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$ be a set of vectors in $\mathbb{H}^d$ where $\mathbb{H}$ is either $\mathbb{R}$ or $\mathbb{C}$. Assume that $x \in \mathbb{H}^d$ such that $b_j = |\langle x, f_j\rangle|$. The phase retrieval problem asks whether we can reconstruct x from $\{b_j\}_{j=1}^m$. Obviously, if $y = cx$ where $|c| = 1$ then $|\langle y, f_j\rangle| = |\langle x, f_j\rangle|$. Thus the best phase retrieval can do is to reconstruct x up to a unimodular constant.

Consider the equivalence relation $\sim$ on $\mathbf{H} := \mathbb{H}^d$: $x \sim y$ if and only if there is a constant $c \in \mathbb{H}$ with $|c| = 1$ such that $x = cy$. Let $\widetilde{\mathbf{H}} := \mathbf{H}/\sim$. We shall use $\tilde{x}$ to denote the equivalent class containing x. For a given $\mathcal{F}$ in $\mathbf{H}$ define the map $\mathbf{M}_{\mathcal{F}} : \widetilde{\mathbf{H}} \longrightarrow \mathbb{R}_+^m$ by

$$\mathbf{M}_{\mathcal{F}}(\tilde{x}) = \left[\left|\langle \tilde{x}, f_1\rangle\right|^2, \dots, \left|\langle \tilde{x}, f_m\rangle\right|^2\right]^\top. \tag{1.1}$$

The phase retrieval problem asks whether an $\tilde{x} \in \widetilde{\mathbf{H}}$ is uniquely determined by $\mathbf{M}_{\mathcal{F}}(\tilde{x})$, i.e. $\tilde{x}$ is recoverable from $\mathbf{M}_{\mathcal{F}}(\tilde{x})$. We say that a set of vectors $\mathcal{F}$ has the *phase retrieval property*, or is *phase retrievable*, if $\mathbf{M}_{\mathcal{F}}$ is injective on $\widetilde{\mathbf{H}} = \mathbb{H}^d/\sim$.

It is known that in the real case $\mathbb{H} = \mathbb{R}$ the set $\mathcal{F}$ needs to have at least $m \geq 2d - 1$ vectors to have the phase retrieval property; furthermore a generic set of $m \geq 2d-1$ elements in $\mathbb{R}^d$ will have the phase retrieval property, (cf. Balan, Casazza and Edidin [1]). In the complex case $\mathbb{H} = \mathbb{C}$ the same question remains open, and is perhaps the most prominent open problem in phase retrieval. It is known that $m \geq 4d - 2$ generic vectors in $\mathbb{C}^d$ have the phase retrieval property [1]. The result is improved to $m \geq 4d - 4$ in [10]. The $m = 4d - 4$ vectors having the phase retrieval property are also constructed in [3]. The current conjecture is that phase retrieval property in $\mathbb{C}^d$ can only hold when $m \geq 4d - 4$.

The aforementioned results concern the general phase retrieval problem in $\mathbb{H}^d$. In many applications, however, the signal x is often sparse with $\|x\|_0 = k \ll d$.

We use the standard notation $\mathbb{H}_k^d$ to denote the subset of $\mathbb{H}^d$ whose elements x have $\|x\|_0 \leq k$. Let $\widetilde{\mathbf{H}}_k$ denote $\mathbb{H}_k^d/\sim$. A set $\mathcal{F}$ of vectors in $\mathbb{H}^d$ is said to have the *k-sparse phase retrieval property*, or is *k-sparse phase retrievable*, if any $\tilde{x} \in \widetilde{\mathbf{H}}_k$ is uniquely determined by $\mathbf{M}_{\mathcal{F}}(\tilde{x})$. In other words, the map $\mathbf{M}_{\mathcal{F}}$ is injective on $\widetilde{\mathbf{H}}_k$. One may naturally ask: *How many vectors does $\mathcal{F}$ need to have so that $\mathcal{F}$ is k-sparse phase retrievable?*

The best current results on the k-sparse phase retrieval property are proved by Li and Voroninski [16], which state that k-sparse phase retrieval property can be achieved by having $m \geq 4k$ and $m \geq 8k$ vectors for the real and complex case, respectively (see also [18]).

In Section 2, we prove sharper results for a set of vectors $\mathcal{F}$ to have the k-sparse phase retrieval property. In the real case $\mathbb{H} = \mathbb{R}$ we obtain a sharp result. We show that for any $k < d$ the set $\mathcal{F}$ must have at least $m \geq 2k$ elements to be k-sparse phase retrievable. Furthermore, any $m \geq 2k$ generic vectors will be k-sparse phase retrievable. In the complex case $\mathbb{H} = \mathbb{C}$ we proved that any $m \geq 4k - 2$ generic vectors have the k-sparse phase retrieval property. We conjecture that this bound is also sharp, namely for $k < d$ a set $\mathcal{F}$ in $\mathbb{C}^d$ needs at least $4k - 2$ vectors to have the k-sparse phase retrieval property.

A foundation of compressive sensing is built on the fact that the recovery of a sparse signal from a system of under-determined linear equations is equivalent to finding the extremal value of ℓ_1 minimization under certain conditions. The ℓ_1 minimization is extended to the phase retrieval in [17] and one also develops many algorithms to compute it (see [20,22]). However, there have been few theoretical results on the recovery performance of ℓ_1 minimization for sparse phase retrieval. In Section 3, we present the null space property, which, to our knowledge, is the first sufficient and necessary condition for the success of ℓ_1 minimization for k-sparse phase retrieval. If we take $k = d$, the null space property is reduced to a condition of the set of vectors $\mathcal{F}$ under which $\mathbf{M}_{\mathcal{F}}$ is injective on $\mathbb{C}^d/\sim$ and we present it in Section 4.

2. Minimal sample number for k-sparse phase retrieval

In this section we study the problem of minimal number of samples (measurements) required for k-sparse phase retrieval. We shall introduce more notations here. Often it is convenient to identify a set of vectors $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$ with the matrix $F = [f_1, f_2, \dots, f_m]$ whose columns are the vectors f_j. When $\mathcal{F}$ is a frame this is known as the *frame matrix* of $\mathcal{F}$. We shall use the term frame matrix for F regardless whether $\mathcal{F}$ is a frame or not. Also for integers $n \le m$ we use the notation $[n:m]$ to denote the set $\{n, n+1, \dots, m\}$. For $x \in \mathbb{H}^d$, we set $|x| := [|x_1|, \dots, |x_d|]$. Similar to before, we let

$$\mathbb{R}^d_k := \big\{x \in \mathbb{R}^d : \|x\|_0 \le k\big\}.$$

Our first theorem completely settles the minimality question for k-sparse phase retrieval in the real case $\mathbb{H} = \mathbb{R}$.

Theorem 2.1. *Let $\mathcal{F} = \{f_1, \dots, f_m\}$ be a set of vectors in $\mathbb{R}^d$. Assume that $\mathcal{F}$ is k-sparse phase retrievable on $\mathbb{R}^d$. Then $m \ge \min\{2k, 2d-1\}$. Furthermore, a set $\mathcal{F}$ of $m \ge \min\{2k, 2d-1\}$ generically chosen vectors in $\mathbb{R}^d$ is k-sparse phase retrievable.*

Proof. Note that the full sparsity case $k = d$ is already known: $m \ge 2d-1$ vectors are needed for phase retrieval and a generic set of $\mathcal{F}$ with $m \ge 2d-1$ vectors will have the phase retrieval property. So we will focus only on $k < d$.

We first prove that $m \ge 2k$. Assume $\mathcal{F}$ has $m < 2k$ elements. We prove $\mathcal{F}$ does not have the k-sparse phase retrieval property by constructing $x, y \in \mathbb{R}^d_k$ with $|\langle x, f_j\rangle| = |\langle y, f_j\rangle|$ but $x \ne \pm y$.

We divide $\mathcal{F}$ into two groups: $\mathcal{F}_1 = \{f_j : j \in [1:k]\}$ and $\mathcal{F}_2 = \{f_j : j \in [k+1:m]\}$. Let the corresponding frame matrices be F_1 and F_2, respectively. Consider the subspace

$$W = \big\{[x_1, x_2, \dots, x_{k+1}, 0, \dots, 0]^\top \in \mathbb{R}^d : x_1, \dots, x_{k+1} \in \mathbb{R}\big\}.$$

For the first group $\mathcal{F}_1$, there exists a $u \in W \setminus \{0\}$ such that $F_1^\top u = 0$, i.e. $\langle f_j, u\rangle = 0$ for all $1 \le j \le k$. This is because $\dim(W) = k+1$ and there are only k equations. Note also that there are at most $k-1$ vectors in the second group $\mathcal{F}_2$ since $m - k < 2k - k = k$. Thus the solution space

$$\big\{v \in W : F_2^\top v = 0\big\}$$

has dimension at least 2. Hence, there exist linearly independent $\alpha, \beta \in W$ so that for all $t, s \in \mathbb{R}$

$$v = t\alpha + s\beta$$

satisfies

$$F_2^\top v = 0, \quad \text{i.e.} \quad \langle f_j, v\rangle = 0 \quad \text{for } j \in [k+1:m].$$

Write $u = [u_1, u_2, \dots, u_d]^\top$ (where $u_j = 0$ for $j > k+1$). Since α and β are linearly independent, we may without loss of generality assume $[\alpha_1, \alpha_2]^\top$ and $[\beta_1, \beta_2]^\top$ are linearly independent, where $\alpha = [\alpha_1, \dots, \alpha_d]^\top$ and $\beta = [\beta_1, \dots, \beta_d]^\top$. We first consider the case where either $u_1 \ne 0$ or $u_2 \ne 0$. Then there exist $s_0, t_0 \in \mathbb{R}$ with $(s_0, t_0) \ne (0,0)$ so that

$$u_1 = t_0\alpha_1 + s_0\beta_1,$$
$$-u_2 = t_0\alpha_2 + s_0\beta_2.$$

Now set $\bar{v} = t_0\alpha + s_0\beta$ and

$$x := u + \bar{v}, \qquad y := u - \bar{v}.$$

Clearly $x, y \in \mathbb{R}_k^d$ since $\mathrm{supp}(x) \subseteq \{1, 3, \ldots, k+1\}$ and $\mathrm{supp}(y) \subseteq \{2, 3, \ldots, k+1\}$. Moreover

$$\langle f_j, x\rangle = \langle f_j, u\rangle + \langle f_j, \bar{v}\rangle = \begin{cases} \langle f_j, \bar{v}\rangle & j \le k \\ \langle f_j, u\rangle & j > k, \end{cases}$$

and similarly

$$\langle f_j, y\rangle = \langle f_j, u\rangle - \langle f_j, \bar{v}\rangle = \begin{cases} \langle f_j, \bar{v}\rangle & j \le k \\ -\langle f_j, u\rangle & j > k. \end{cases}$$

It follows that $|\langle f_j, x\rangle| = |\langle f_j, y\rangle|$ for all j but $x \neq \pm y$. We next consider the case where $u_1 = u_2 = 0$. Then there exist $s_0, t_0 \in \mathbb{R}$ with $(s_0, t_0) \neq (0, 0)$ so that

$$0 = t_0\alpha_1 + s_0\beta_1,$$
$$1 = t_0\alpha_2 + s_0\beta_2.$$

Similar to before, we set $\bar{v} = t_0\alpha + s_0\beta$ and

$$x := u + \bar{v}, \qquad y := u - \bar{v}.$$

Then $x, y \in \mathbb{R}_k^d$ and $|\langle f_j, x\rangle| = |\langle f_j, y\rangle|$ for all j but $x \neq \pm y$. Thus $\mathcal{F}$ does not have the k-sparse phase retrieval property in $\mathbb{R}_k^d$.

We next prove that a set $\mathcal{F}$ of $m \ge 2k$ generic vectors will have the k-sparse phase retrieval property. Let us first fix $I, J \subset [1:N]$ with $\#I = \#J = k$. The goal is to prove that if $x, y \in \mathbb{R}_k^N$ with $\mathrm{supp}(x) \subset I$ and $\mathrm{supp}(y) \subset J$ satisfying

$$\big|\langle f_j, x\rangle\big|^2 = \big|\langle f_j, y\rangle\big|^2, \quad j = 1, \ldots, m, \tag{2.1}$$

then $x = \pm y$. Eq. (2.1) implies that for all j we have

$$\langle f_j, x - y\rangle \cdot \langle f_j, x + y\rangle = 0. \tag{2.2}$$

Thus either $\langle f_j, x - y\rangle = 0$ or $\langle f_j, x + y\rangle = 0$. Without loss of generality, we assume that

$$\begin{aligned} &\langle f_j, x - y\rangle = 0, \quad j \in [1:n] \\ &\langle f_j, x + y\rangle = 0, \quad j \in [n+1:m]. \end{aligned} \tag{2.3}$$

Set

$$L := I \cap J \quad \text{and} \quad \ell := \#L.$$

For convenience we write

$$\begin{aligned} x &= u_x + v_x, \quad \mathrm{supp}(u_x) \subset L, \quad \mathrm{supp}(v_x) \subset I \setminus L, \\ y &= u_y + v_y, \quad \mathrm{supp}(u_y) \subset L, \quad \mathrm{supp}(v_y) \subset J \setminus L. \end{aligned}$$

We abuse the notation a little by viewing $v_x \in \mathbb{R}^{k-\ell}$ since it is supported on $I \setminus L$ with $\#(I \setminus L) = k - \ell$. Similarly we view $v_y \in \mathbb{R}^{k-\ell}$ and $u_x, u_y \in \mathbb{R}^{\ell}$. Set

$$w_- := u_x - u_y, \qquad w_+ := u_x + u_y, \quad \text{and} \quad z := \begin{bmatrix} v_x \\ v_y \\ w_- \\ w_+ \end{bmatrix} \in \mathbb{R}^{2k}.$$

Using the notions above, we have

$$\begin{aligned} \langle f_j, x - y \rangle &= \langle f_j, v_x \rangle - \langle f_j, v_y \rangle + \langle f_j, w_- \rangle, \\ \langle f_j, x + y \rangle &= \langle f_j, v_x \rangle + \langle f_j, v_y \rangle + \langle f_j, w_+ \rangle. \end{aligned} \tag{2.4}$$

Set $A := F^\top$ where F is the frame matrix of $\mathcal{F}$. Combining (2.3) and (2.4) now yields

$$\begin{bmatrix} A_{[1:n],I\setminus L} & -A_{[1:n],J\setminus L} & A_{[1:n],L} & 0 \\ A_{[n+1:m],I\setminus L} & A_{[n+1:m],J\setminus L} & 0 & A_{[n+1:m],L} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ w_- \\ w_+ \end{bmatrix} = 0, \tag{2.5}$$

where for any index sets J_1, J_2 we use the notation A_{J_1,J_2} to denote the sub-matrix of A with the rows indexed in J_1 and columns indexed in J_2. To show $x = \pm y$ we only need to show that the linear equations (2.5) force $v_x = 0, v_y = 0$ and either $w_- = 0$ or $w_+ = 0$.

We first consider the case $n \geq 2k - \ell$. In this case, we consider only the first set of Eqs. (2.5)

$$\begin{bmatrix} A_{[1:n],I\setminus L} & -A_{[1:n],J\setminus L} & A_{[1:n],L} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ w_- \end{bmatrix} = 0. \tag{2.6}$$

Note that the matrix

$$\begin{bmatrix} A_{[1:n],I\setminus L} & -A_{[1:n],J\setminus L} & A_{[1:n],L} \end{bmatrix}$$

has dimensions $n \times (2k - \ell)$. The elements are generically chosen. Thus it has full rank $2k - \ell$. It follows that (2.6) has only trivial solution $v_x = 0$, $v_y = 0$ and $w_- = 0$. Hence $x = y$.

We next consider the case with $m - n \geq 2k - \ell$. Here we consider the second set of Eq. (2.5):

$$\begin{bmatrix} A_{[n+1:m],I\setminus L} & A_{[n+1:m],J\setminus L} & A_{[n+1:m],L} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ w_+ \end{bmatrix} = 0. \tag{2.7}$$

The same argument used for the case $n \geq 2k - \ell$ now applies to yield $v_x = 0, v_y = 0$ and $w_+ = 0$. Hence in this case $x = -y$.

We finally consider the case where $n < 2k - \ell$ and $m - n < 2k - \ell$. In this case we must have

$$2k - \ell > m - n \geq 2k - n,$$

and hence $n > \ell$. Similarly, we have $\ell < 2k - n \leq m - n$. We argue that the rank of the matrix in (2.5) is $2k$ when $F^\top$ is generic. Let B denote the matrix in (2.5). If $\operatorname{rank}(B) < 2k$ then all $2k \times 2k$ sub-matrices of

B have determinant 0. Note that each determinant is either identically 0 or a nontrivial polynomial of the entries of F. Hence if there exists a single example of a matrix B with $\operatorname{rank}(B) = 2k$ then $\operatorname{rank}(B) = 2k$ for a generic choice of F. We shall construct an example of such an F with $\operatorname{rank}(B) = 2k$. Set

$$A_{[1:n],L} = \begin{bmatrix} I_\ell \\ 0 \end{bmatrix}, \qquad A_{[n+1:m],L} = \begin{bmatrix} I_\ell \\ 0 \end{bmatrix},$$

$$[A_{[1:n],I\setminus L}, -A_{[1:n],J\setminus L}] = \begin{bmatrix} 0 \\ H_1 \end{bmatrix}, \qquad [A_{[n+1:m],I\setminus L}, -A_{[n+1:m],J\setminus L}] = \begin{bmatrix} 0 \\ H_2 \end{bmatrix},$$

where I_ℓ denotes the $\ell \times \ell$ identity matrix. With this choice, for almost all $H_1 \in \mathbb{R}^{(n-\ell)\times(2k-2\ell)}, H_2 \in \mathbb{R}^{(m-n-\ell)\times(2k-2\ell)}$ we have $\operatorname{rank}(B) = 2k$. The solution to (2.5) is thus trivial, namely $v_x = 0$, $v_y = 0$, $w_- = 0$ and $w_+ = 0$. Thus $x = y = 0$. The theorem is now proved. □

We next consider the complex case. Similar to the real case we set

$$\mathbb{C}^d_k := \{x \in \mathbb{C}^d : \|x\|_0 \le k\}.$$

Then we have

Theorem 2.2. *A set $\mathcal{F}$ of $m \ge 4k-2$ generically chosen vectors in $\mathbb{C}^d$ is k-sparse phase retrievable.*

Proof. We shall identify $\mathcal{F}$ with F where $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$ is the corresponding frame matrix, $F = [f_{ij}]$. Following the technique in [1] we shall view F as an element in $\mathbb{R}^{2md}$. The goal here is to show that the set of matrices F that are not k-sparse phase retrievable has local real dimension strictly smaller than $2md$ provided $m \ge 4k-2$.

For any subset of indices $I, J \subset [1:d]$ with $\#I = \#J = k$ let $G_{I,J}$ denote the set of matrices in $\mathbb{C}^{d\times m}$ with the following property: There exist $x, y \in \mathbb{C}^d$ where $\operatorname{supp}(x) \subset I$, $\operatorname{supp}(y) \subset J$ and $x \ne cy$ with $|c| = 1$ such that $\mathbf{M}_\mathcal{F}(x) = \mathbf{M}_\mathcal{F}(y)$, i.e. $|\langle f_j, x\rangle| = |\langle f_j, y\rangle|$ for all j. Now if $\mathbf{M}_\mathcal{F}(x) = \mathbf{M}_\mathcal{F}(y)$, then for any $a, \omega \in \mathbb{C}$ with $|\omega| = 1$ we also have $\mathbf{M}_\mathcal{F}(ax) = \mathbf{M}_\mathcal{F}(a\omega y)$. Thus for any $F \in G_{I,J}$ we may find $x, y \in \mathbb{C}^d$ with $\mathbf{M}_\mathcal{F}(x) = \mathbf{M}_\mathcal{F}(y)$ such that

- $\operatorname{supp}(x) \subset I$, $\operatorname{supp}(y) \subset J$.
- The first nonzero entry of x is 1.
- The first nonzero entry of y is real and positive.

Let X denote the subset of $\mathbb{C}^d$ consisting of elements $x \in \mathbb{C}^d$ whose first nonzero entry is 1. Let Y denote the subset of $\mathbb{C}^d$ consisting of elements $y \in \mathbb{C}^d$ whose first nonzero entry, if it exists, is real and positive. Note that in essence X can be viewed as the projective space $\mathbb{P}^{d-1} \setminus \{0\}$ and Y can be viewed as the set $\mathbb{C}^d/\!\sim$. Let $\mathbb{C}^d_I$ denote the set of vectors $x \in \mathbb{C}^d$ such that $\operatorname{supp}(x) \subseteq I$. Now consider the set of 3-tuples

$$\mathcal{A}_{I,J} := \{(F, x, y)\}$$

with the following properties:

- $x \in X \cap \mathbb{C}^d_I$ and $y \in Y \cap \mathbb{C}^d_J$.
- $x \ne \omega y$ for any $\omega \in \mathbb{C}$ with $|\omega| = 1$.
- $\mathbf{M}_\mathcal{F}(x) = \mathbf{M}_\mathcal{F}(y)$.

Now the projection of $\mathcal{A}_{I,J}$ to the first component gives the full set $G_{I,J}$. Each $(F,x,y) \in \mathcal{A}_{I,J}$ gives rise to the constraints $|\langle f_j, x\rangle| = |\langle f_j, y\rangle|$ for $j \in [1:m]$, which lead to the set of quadratic equations in $\mathrm{Re}(f_{ij})$, $\mathrm{Im}(f_{ij})$ (by viewing x, y as fixed)

$$\left|\sum_{k=1}^{N} f_{kj}x_k\right|^2 = \left|\sum_{k=1}^{N} f_{kj}y_k\right|^2, \quad j = 1,\dots,m. \tag{2.8}$$

Note that all equations are independent and each is non-trivial because $x \neq y$ in $\mathbb{C}^d/\!\sim$. Thus for any fixed x, y the set of such $A = [f_{ij}]$ satisfying (2.8) is a real algebraic variety of (real) codimension m. Hence, $\mathcal{A}_{I,J}$ has local dimension everywhere at most

$$2md - m + \dim_{\mathbb{R}}\big(X \cap \mathbb{C}_I^d\big) + \dim_{\mathbb{R}}\big(Y \cap \mathbb{C}_J^d\big) = 2md - m + 2k - 2 + 2k - 1 = 2md - (m - 4k + 3).$$

It follows from $m \geq 4k-2$ that $\mathcal{A}_{I,J}$ has local (real) dimension at most $2md-1$. Now $G_{I,J}$ is the projection of $\mathcal{A}_{I,J}$ onto the first component. Thus, $G_{I,J}$ has dimension at most $2md-1$. In other words, a generic $F \in \mathbb{C}^{d\times m}$ is not in $G_{I,J}$.

Finally, the set of $F \in \mathbb{C}^{d\times m}$ not having the k-sparse phase retrieval property for $\mathbb{C}_k^d$ is the union of all $G_{I,J}$ with $\#I = \#J = k$. It is a finite union. The theorem is now proved. □

Remark. Although the above theorem shows that in the complex case any $m \geq 4k-2$ generically chosen vectors are k-sparse phase retrievable, it is unknown whether $4k-2$ is in fact the minimal number required. It will be interesting to use the technology developed in [10] to improve the result.

3. Null space property for sparse phase retrieval

In this section, we investigate the performance of ℓ_1 minimization for sparse phase retrieval with extending the null space property in compressed sensing to the phase retrieval setting. We first introduce the null space property in compressed sensing, and then extend it to the phase retrieval setting on $\mathbb{R}_k^d$ and $\mathbb{C}_k^d$, respectively.

3.1. Null space property

A key concept in compressive sensing is the so-called *null space property* of a matrix. For a given frame $\mathcal{F} = \{f_1,\dots,f_m\} \subset \mathbb{H}^d$, we use F to denote the frame matrix. Let $\mathcal{N}(F)$ denote the kernel of $F^\top$, i.e.,

$$\mathcal{N}(F) = \big\{\eta \in \mathbb{H}^d : \langle f_j, \eta\rangle = 0, j = 1,\dots,m\big\}.$$

To state conveniently, when $F = \emptyset$, we set $\mathcal{N}(F) := \mathbb{H}^d$.

Definition 3.1. The matrix F satisfies the *null space property of order k* if for any nonzero $\eta = [\eta_1,\dots,\eta_d]^\top \in \mathcal{N}(F)$ and any $T \subset [1:d]$ with $\#T \leq k$ it holds that

$$\|\eta_T\|_1 < \|\eta_{T^c}\|_1,$$

where T^c is the complementary index set of T and η_T is the restriction of η to T.

A fundamental result in compressed sensing is that a signal $x \in \mathbb{H}_k^d$ can be recovered via the ℓ_1 minimization if and only if the sensing matrix A has the null space property of order k. We state it as follows (see [9,13–15,19]):

Theorem 3.1. *Let $\mathcal{F}$ be a set of vectors in $\mathbb{H}^d$ and F be the associated frame matrix. Then F satisfies the null space property of order k if and only if it has*

$$\operatorname*{argmin}_{x\in\mathbb{H}^d}\{\|x\|_1 : F^\top x = F^\top x_0\} = x_0$$

for every $x_0 \in \mathbb{H}^d_k$.

3.2. The null space property for the real sparse phase retrieval

Our goal here is to extend Theorem 3.1 to the phase retrieval for the real signal. For a given frame $\mathcal{F} = \{f_1, \dots, f_m\}$ and a subset S of $[1:m]$ we shall use $\mathcal{F}_S$ to denote the set $\mathcal{F}_S := \{f_j : j \in S\}$. Similarly for the frame matrix we shall use F_S to denote the corresponding frame matrix of $\mathcal{F}_S$, i.e. the matrix whose columns are the vectors of $\mathcal{F}_S$. We first consider the real case.

Theorem 3.2. *Let $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$ be a set of vectors in $\mathbb{R}^d$ and F be the associated frame matrix. The following properties are equivalent:*

(A) *For any $x_0 \in \mathbb{R}^d_k$ we have*

$$\operatorname*{argmin}_{x\in\mathbb{R}^d}\{\|x\|_1 : |F^\top x| = |F^\top x_0|\} = \{\pm x_0\}, \tag{3.1}$$

where $|F^\top x| = [|\langle f_1, x\rangle|, \dots, |\langle f_m, x\rangle|]^\top$.

(B) *For every $S \subseteq [1:m]$, it holds*

$$\|u+v\|_1 < \|u-v\|_1$$

for all nonzero $u \in \mathcal{N}(F_S)$ and $v \in \mathcal{N}(F_{S^c})$ satisfying $\|u+v\|_0 \le k$.

Proof. First we show (B) $\Rightarrow$ (A). Let $b = [b_1, \dots, b_m]^\top := |F^\top x_0|$ where $x_0 \in \mathbb{R}^d_k$. For a fixed $\epsilon \in \{1,-1\}^m$ set $b_\epsilon := [\epsilon_1 b_1, \dots, \epsilon_m b_m]^\top$. We now consider the following minimization problem:

$$\min \|x\|_1 \quad \text{s.t.} \quad F^\top x = b_\epsilon. \tag{3.2}$$

The solution to (3.2) is denoted as x_ϵ. We claim that for any $\epsilon \in \{1,-1\}^m$ we must have

$$\|x_\epsilon\|_1 \ge \|x_0\|_1$$

if x_ϵ exists (it may not exist), and the equality holds if and only if $x_\epsilon = \pm x_0$.

To prove the claim let $\epsilon^* \in \{1,-1\}^m$ such that $b_{\epsilon_*} = F^\top x_0$. Note that property (B) implies the classical null space property of order k. To see this, for any nonzero $\eta \in \mathcal{N}(F)$ and $T \subseteq [1:d]$ with $\#T \le k$, set $u := \eta$ and $v := \eta_T - \eta_{T^c}$. Let $S = [1:m]$. Then $u \in \mathcal{N}(F_S)$ and $v \in \mathcal{N}(F_{S^c})$. The hypothesis of (B) now implies

$$2\|\eta_T\|_1 = \|u+v\|_1 < \|u-v\|_1 = 2\|\eta_{T^c}\|_1.$$

Consequently we must have $x_{\epsilon^*} = x_0$ by Theorem 3.1. Now for any $\epsilon \in \{-1,1\}^m \ne \pm\epsilon^*$, if x_ϵ doesn't exist then we have nothing to prove. Assume it does exist. Set $S_* := \{j : \epsilon_j = \epsilon^*_j\}$. Then

$$\langle f_j, x_\epsilon\rangle = \begin{cases} \langle f_j, x_0\rangle & j \in S_*, \\ -\langle f_j, x_0\rangle & j \in S_*^c. \end{cases}$$

Set $u := x_0 - x_\epsilon$ and $v := x_0 + x_\epsilon$. Clearly $u \in \mathcal{N}(F_{S_*})$ and $v \in \mathcal{N}(F_{S_*^c})$. Furthermore $u + v = 2x_0 \in \mathbb{R}_k^d$. By the hypothesis of (B) we must have

$$2\|x_0\|_1 = \|u + v\|_1 < \|u - v\|_1 = 2\|x_\epsilon\|_1.$$

This proves (A).

Next we prove (A) $\Rightarrow$ (B). Assume (B) is false, namely, there exist nonzero $u \in \mathcal{N}(F_S)$ and $v \in \mathcal{N}(F_{S^c})$ such that $\|u + v\|_1 \geq \|u - v\|_1$ and $u + v \in \mathbb{R}_k^d$. Now set

$$x_0 := u + v \in \mathbb{R}_k^d.$$

Clearly,

$$\big|\langle f_j, x_0\rangle\big| = \big|\langle f_j, u + v\rangle\big| = \big|\langle f_j, u - v\rangle\big|, \quad j = 1, \ldots, m$$

since either $\langle f_j, u\rangle = 0$ or $\langle f_j, v\rangle = 0$. In other words, $|F^\top x_0| = |F^\top(u - v)|$. Note that $u - v \neq -x_0$, for otherwise we would have $u = 0$, a contradiction. It follows from the hypothesis of (A) that we must have

$$\|x_0\|_1 = \|u + v\|_1 < \|u - v\|_1.$$

This is a contradiction. □

Remark. Theorem 3.2 extends results for the null space property of order k in compressive sensing to phase retrieval. It will be very interesting for constructing matrix $A \in \mathbb{R}^{m\times d}$ with $m \asymp k \log d$ satisfying (B) in Theorem 3.2.

3.3. The null space property for the complex sparse phase retrieval

We now consider the complex case $\mathbb{H} = \mathbb{C}$. Throughout this subsection, we say that $\mathcal{S} = \{S_1, \ldots, S_p\}$, $p \geq 2$, is a partition of $[1 : m]$ if

$$S_j \subset [1 : m], \quad \bigcup_{j=1}^{p} S_j = [1 : m] \quad \text{and} \quad S_j \cap S_\ell = \emptyset \quad \text{for all } j \neq \ell.$$

To state conveniently, we set $\mathbb{S} := \{c \in \mathbb{C} : |c| = 1\}$ and

$$\mathbb{S}^m := \big\{(c_1, \ldots, c_m) \in \mathbb{C}^m : |c_j| = 1, j \in [1 : m]\big\}.$$

Then we have:

Theorem 3.3. *Let $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ be a set of vectors in $\mathbb{C}^d$ and F be the associated frame matrix. The following properties are equivalent.*

(A) *For any $x_0 \in \mathbb{C}_k^d$ we have*

$$\operatorname*{argmin}_{\tilde{x}\in\mathbb{C}^d/\sim} \big\{\|x\|_1 : \big|F^\top x\big| = \big|F^\top x_0\big|\big\} = \tilde{x}_0, \tag{3.3}$$

where $|F^\top x| = [|\langle f_1, x\rangle|, \dots, |\langle f_m, x\rangle|]^\top$ *and* $\tilde{x}_0$ *denotes the equivalent class* $\{cx_0 : c \in \mathbb{S}\}$ *in* $\mathbb{C}^d/\sim$ *containing* x_0.

(B) *Suppose that* $S_1, \dots, S_p$ *is any partition of* $[1:m]$ *and that* $c_1, \dots, c_p \in \mathbb{S}$ *are any* p *pairwise distinct complex numbers. If* $\eta_j \in \mathcal{N}(F_{S_j}) \setminus \{0\}$, $j \in [1:m]$, *satisfy*

$$\frac{\eta_1 - \eta_\ell}{c_1 - c_\ell} = \frac{\eta_1 - \eta_j}{c_1 - c_j} \in \mathbb{C}_k^d \setminus \{0\} \quad \text{for all } \ell, j \in [2:p], \tag{3.4}$$

then

$$\|\eta_j - \eta_\ell\|_1 < \|c_\ell \eta_j - c_j \eta_\ell\|_1,$$

for all $j, \ell \in [1:p]$ *with* $j \neq \ell$.

Proof. We first show (B) $\Rightarrow$ (A). Let $b = [b_1, \dots, b_m]^\top := |F^\top x_0|$ where $x_0 \in \mathbb{C}_k^d$. For a fixed $\epsilon \in \mathbb{S}^m$ set $b_\epsilon := [\epsilon_1 b_1, \dots, \epsilon_m b_m]^\top$. We now consider the following minimization problem:

$$\min \|x\|_1 \quad \text{s.t.} \quad F^\top x = b_\epsilon. \tag{3.5}$$

The solution to (3.5) is denoted as x_ϵ. We claim that for any $\epsilon \in \mathbb{S}^m$ we must have

$$\|x_\epsilon\|_1 \geq \|x_0\|_1$$

if x_ϵ exists (it may not exist), and the equality holds if and only if $\tilde{x}_\epsilon = \tilde{x}_0$.

To prove the claim let $\epsilon^* \in \mathbb{S}^m$ such that $b_{\epsilon^*} = F^\top x_0$. Note that property (B) implies the classical null space property of order k. To see this, take $S_1 = [1:m]$, $S_2 = \emptyset$, $c_1 = 1$ and $c_2 = -1$. Then (3.4) is reduced to require that $\eta_1 - \eta_2 \in \mathbb{C}_k^d$, i.e., $\eta_1 - \eta_2$ is k-sparse. Given $T \subset [1:d]$ with $\#T \leq k$ and $\eta_1 \in \mathcal{N}(F) = \mathcal{N}(F_{S_1})$, set

$$\eta_2 := (\eta_1)_{T^c} - (\eta_1)_T \in \mathcal{N}(F_{S_2}) = \mathbb{C}^d.$$

Then $\eta_1 - \eta_2 \in \mathbb{C}_k^d$. The (B) implies that

$$2\big\|(\eta_1)_T\big\|_1 = \|\eta_1 - \eta_2\|_1 < \|\eta_1 + \eta_2\| = 2\big\|(\eta_1)_{T^c}\big\|_1,$$

which implies the classical null space property.

Consequently we must have $\tilde{x}_{\epsilon^*} = \tilde{x}_0$ by Theorem 3.1. Now we consider an arbitrary $\epsilon \in \mathbb{S}^m$. If $\tilde{\epsilon} = \tilde{\epsilon^*}$, then $\tilde{x}_\epsilon = \tilde{x}_0$. So, we only consider the case where $\tilde{\epsilon} \neq \tilde{\epsilon^*}$. If x_ϵ does not exist then we have nothing to prove. Assume it does exist. Set $c'_j := \epsilon_j/\epsilon_j^*$ and $\eta'_j := c'_j x_{\epsilon^*} - x_\epsilon$ for $1 \leq j \leq m$. We can use c'_j to define an equivalence relation on $[1:m]$, namely $j \sim \ell$ if $c'_j = c'_\ell$. This equivalence relation leads to a partition $\mathcal{S} = \{S_1, \dots, S_p\}$ of $[1:m]$. Now we set $c_j := c'_\ell$ where $\ell \in S_j$. Clearly all c_j, $1 \leq j \leq p$, are distinct and unimodular.

Now set $\eta_j := c_j x_{\epsilon^*} - x_\epsilon$. Then we have

$$\eta_j \in \mathcal{N}(F_{S_j}) \setminus \{0\}, \quad \text{for all } j \in [1:p]$$

and

$$\frac{\eta_1 - \eta_j}{c_1 - c_j} = \frac{\eta_1 - \eta_\ell}{c_1 - c_\ell} \in \mathbb{C}_k^d \quad \text{for all } j, \ell \in [2:p].$$

By the hypothesis of (B) we must have

$$|c_j - c_\ell| \cdot \|x_0\|_1 = \|\eta_j - \eta_\ell\|_1 < \|c_\ell \eta_j - c_j \eta_\ell\|_1 = |c_j - c_\ell| \cdot \|x_\epsilon\|_1,$$

which implies that

$$\|x_0\|_1 < \|x_\epsilon\|_1.$$

This proves (A).

We next prove (A) $\Rightarrow$ (B). Assume (B) is false, namely, there exist nonzero $\eta_j \in \mathcal{N}(F_{S_j})$, $j \in [1:p]$ satisfying (3.4) but

$$\|\eta_{j_0} - \eta_{\ell_0}\|_1 \geq \|c_{\ell_0}\eta_{j_0} - c_{j_0}\eta_{\ell_0}\|_1$$

for some distinct $j_0, \ell_0 \in [1:p]$. Note that (3.4) implies that

$$\frac{\eta_j - \eta_\ell}{c_j - c_\ell} = \frac{\eta_m - \eta_n}{c_m - c_n} \in \mathbb{C}_k^d \setminus \{0\}, \tag{3.6}$$

for all $j, \ell, m, n \in [1:p]$ with $j \neq \ell$ and $m \neq n$. Without loss of generality, we assume that $j_0 = 1$, $\ell_0 = 2$, i.e.,

$$\|\eta_1 - \eta_2\|_1 \geq \|c_2\eta_1 - c_1\eta_2\|_1. \tag{3.7}$$

Set

$$x_0 := \eta_1 - \eta_2,$$

and (3.6) implies that $x_0 \in \mathbb{C}_k^d \setminus \{0\}$. We claim that

$$\big|\langle f_j, x_0\rangle\big| = \big|\langle f_j, \eta_1 - \eta_2\rangle\big| = \big|\langle f_j, c_2\eta_1 - c_1\eta_2\rangle\big|, \quad \text{for all } j \in [1:p]. \tag{3.8}$$

Note that x_0 is k-sparse. Combining (3.8), (3.7) and (3.3) now yields

$$cx_0 = c\eta_1 - c\eta_2 = c_2\eta_1 - c_1\eta_2$$

for some $c \in \mathbb{S}$. Consequently we obtain

$$(c - c_2)\eta_1 = (c - c_1)\eta_2,$$

which implies that

$$\eta_2 = \frac{c - c_2}{c - c_1}\eta_1. \tag{3.9}$$

Here, note that $c \notin \{c_1, c_2\}$, for otherwise we will have either $\eta_1 = 0$ or $\eta_2 = 0$. Combining (3.4) and (3.9) leads to

- η_1 is k-sparse;
- for all $j \in [2:p]$, η_j and η_1 are linear dependent and hence $\eta_1 \in \mathcal{N}(F_{S_j})$.

And hence we have $F^\top \eta_1 = 0$. By the hypothesis of (A) and $\eta_1 \in \mathbb{C}_k^d$ we have $\eta_1 = 0$. A contradiction.

We remain to prove (3.8). First, when $j \in S_1 \cup S_2$, (3.8) holds, since either $\langle f_j, \eta_1\rangle = 0$ or $\langle f_j, \eta_2\rangle = 0$. We consider the case where $j \in S_3$. Set $y_0 := \frac{\eta_1 - \eta_2}{c_1 - c_2}$. Then (3.6) implies that

$$\frac{\eta_1 - \eta_3}{c_1 - c_3} = \frac{\eta_2 - \eta_3}{c_2 - c_3} = y_0$$

and hence

$$\eta_1 = (c_1 - c_3)y_0 + \eta_3,$$
$$\eta_2 = (c_2 - c_3)y_0 + \eta_3.$$

Note that $\langle f_j, \eta_3\rangle = 0$ with $j \in S_3$. Then

$$\begin{aligned}\big|\langle f_j, c_2\eta_1 - c_1\eta_2\rangle\big| &= \big|\big\langle f_j, c_2(c_1 - c_3)y_0 - c_1(c_2 - c_3)y_0\big\rangle\big| \\ &= \big|\big\langle f_j, c_3(c_1 - c_2)y_0\big\rangle\big| = \big|\langle f_j, \eta_1 - \eta_2\rangle\big| = \big|\langle f_j, x_0\rangle\big|.\end{aligned}$$

Using a similar argument, we easily prove the claim for $j \in S_4, \ldots, S_p$. □

Remark. When $p = 2$, Eq. (3.4) is reduced to $\eta_1 - \eta_2 \in \mathbb{C}_k^d \setminus \{0\}$. And hence, if take $p = 2$, the (B) in Theorem 3.3 implies that

$$\|\eta_1 - \eta_2\|_1 < \|c_2\eta_1 - c_1\eta_2\|_1$$

for all nonzero $\eta_1 \in \mathcal{N}(F_{S_1})$ and $\eta_2 \in \mathcal{N}(F_{S_2})$ satisfying $\eta_1 - \eta_2 \in \mathbb{C}_k^d \setminus \{0\}$ and all $c_1, c_2 \in \mathbb{S}$ with $c_1 \neq c_2$.

Remark. In [21], Tillmann and Pfetsch investigated the computational complexity of the classical null space property, and d'Aspremont and Ghaou also designed algorithms to test it [11]. It will be very interesting to extend the result to the null space property introduced in Theorem 3.3 in the future research.

4. Null space property for general phase retrieval

Theorem 3.2 and Theorem 3.3 present the null space property for the phase retrievable on $\mathbb{R}_k^d$ and $\mathbb{C}_k^d$, respectively. In phase retrieval, one is also interested in the condition under which F is phase retrievable on $\mathbb{R}^d$ or $\mathbb{C}^d$. For the real case, such a condition is presented in [1]:

Theorem 4.1. *(See [1].) Let $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ be a set of vectors in $\mathbb{R}^d$ and F be the associated frame matrix. The following properties are equivalent:*

(A) *F is phase retrievable on $\mathbb{R}^d$;*
(B) *For every subset $S \subset \{1, \ldots, m\}$, either $\{f_j\}_{j\in S}$ spans $\mathbb{R}^d$ or $\{f_j\}_{j\in S^c}$ spans $\mathbb{R}^d$.*

We next consider the complex case. Motivated by Theorem 3.3, we can present the null space property under which F is phase retrievable on $\mathbb{C}^d$. It can be considered as an extension of Theorem 4.1:

Theorem 4.2. *Let $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ be a set of vectors in $\mathbb{C}^d$ and F be the associated frame matrix. The following properties are equivalent:*

(A) *F is phase retrievable on $\mathbb{C}^d$;*

(B) *Suppose that* $S_1, \dots, S_p$ *is any partition of* $[1:m]$ *and that* $c_1, \dots, c_p \in \mathbb{S}$ *are any* p *pairwise distinct complex numbers. There exists no* $\eta_j \in \mathcal{N}(F_{S_j}) \setminus \{0\}$, $j = 1, \dots, p$, *such that*

$$\frac{\eta_1 - \eta_\ell}{c_1 - c_\ell} = \frac{\eta_1 - \eta_j}{c_1 - c_j} \neq 0 \quad \text{for all } \ell, j \in [2:p]. \tag{4.1}$$

Proof. We first prove (A) ⇒ (B). Assume (B) is false, namely, there exist nonzero $\eta_j \in \mathcal{N}(F_{S_j})$, $j \in [1:p]$, satisfying (4.1). Set

$$x_0 := \eta_1 - \eta_2.$$

Using a similar method as the proof of (3.8), we obtain that

$$\big|\langle f_j, x_0\rangle\big| = \big|\langle f_j, \eta_1 - \eta_2\rangle\big| = \big|\langle f_j, c_2\eta_1 - c_1\eta_2\rangle\big|, \quad \text{for all } j \in [1:p].$$

Then, according to (A) and the definition of phase retrievable, we have

$$cx_0 = c\eta_1 - c\eta_2 = c_2\eta_1 - c_1\eta_2$$

for some unimodular constant $c \in \mathbb{S} \setminus \{c_1, c_2\}$, which implies that

$$\eta_2 = \frac{c - c_2}{c - c_1}\eta_1. \tag{4.2}$$

Combining (4.1) and (4.2), we obtain that, for all $j \in [2:p]$, η_j and η_1 are linear dependent and hence $\eta_1 \in \mathcal{N}(F_{S_j})$. So, $F^\top \eta_1 = 0$. Then (A) implies that $\eta_1 = 0$, a contradiction.

We next show (B) ⇒ (A). Set $b = [b_1, \dots, b_m]^\top := |F^\top x_0|$ where $x_0 \in \mathbb{C}^d \setminus \{0\}$. For a fixed $\epsilon \in \mathbb{S}^m$ set $b_\epsilon := [\epsilon_1 b_1, \dots, \epsilon_m b_m]^\top$. We now consider the solution to

$$F^\top x = b_\epsilon. \tag{4.3}$$

The solution to (4.3) is denoted as x_ϵ. We claim that if x_ϵ exists then $\tilde{x}_\epsilon = \tilde{x}_0$, which implies (A). Recall that $\tilde{x}_0$ denotes the equivalent class $\{cx_0 : c \in \mathbb{S}\}$ in $\mathbb{C}^d/\!\sim$ containing x_0. To prove the claim let $\epsilon^* \in \mathbb{S}^m$ such that $b_{\epsilon^*} = F^\top x_0$. Then (B) implies that the rank of F is d. Consequently we must have $x_{\epsilon^*} = x_0$. Now we consider an arbitrary $\epsilon \in \mathbb{S}^m$. If $\tilde{\epsilon} = \tilde{\epsilon^*}$, then $\tilde{x}_\epsilon = \tilde{x}_0$. To this end, we only need prove that x_ϵ does not exist if $\tilde{\epsilon} \neq \tilde{\epsilon^*}$. Assume x_ϵ does exist. Set $c'_j := \epsilon_j/\epsilon^*_j$ and $\eta'_j := c'_j x_{\epsilon^*} - x_\epsilon$ for $1 \le j \le m$. We can use c'_j to define an equivalence relation on $[1:m]$, namely $j \sim \ell$ if $c'_j = c'_\ell$. This equivalence relation leads to a partition $\mathcal{S} = \{S_1, \dots, S_p\}$ of $[1:m]$. Now we set $c_j := c'_\ell$ where $\ell \in S_j$. Clearly all c_j, $1 \le j \le p$, are distinct and unimodular. Now set $\eta_j := c_j x_{\epsilon^*} - x_\epsilon$. By definition for all $1 \le j \le p$ we have

$$\eta_j \in \mathcal{N}(F_{S_j}) \setminus \{0\}$$

and

$$\frac{\eta_1 - \eta_j}{c_1 - c_j} = \frac{\eta_1 - \eta_\ell}{c_1 - c_\ell} \neq 0 \quad \text{for all } j, \ell \in [2:p],$$

which contradicts with (B). And hence x_ϵ does not exist if $\tilde{\epsilon} \neq \tilde{\epsilon}^*$. This proves (A). □

Remark. When $p = 2$, Eq. (4.1) is reduced to $\eta_1 - \eta_2 \neq 0$ which in turn implies that either $\mathcal{N}(F_{S_1}) = \{0\}$ or $\mathcal{N}(F_{S_2}) = \{0\}$.

References

[1] R. Balan, P. Casazza, D. Edidin, On signal reconstruction without phase, Appl. Comput. Harmon. Anal. 20 (2006) 345–356.
[2] R. Balan, B. Bodmann, P. Casazza, D. Edidin, Painless reconstruction from magnitudes of frame coefficients, J. Fourier Anal. Appl. 15 (4) (2009) 488–501.
[3] B.G. Bodmann, N. Hammen, Stable phase retrieval with low-redundancy frame, arXiv:1302.5487.
[4] E.J. Candès, T. Tao, Decoding by linear programming, IEEE Trans. Inform. Theory 51 (2005) 4203–4215.
[5] E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inform. Theory 52 (2) (2006) 489–509.
[6] E.J. Candès, J. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements, Comm. Pure Appl. Math. 59 (8) (2006) 1207–1223.
[7] E. Candès, T. Strohmer, V. Voroninski, PhaseLift: exact and stable signal recovery from magnitude measurements via convex programming, Comm. Pure Appl. Math. 66 (8) (2013) 1241–1274.
[8] E. Candès, Y. Eldar, T. Strohmer, V. Voroninski, Phase retrieval via matrix completion problem, SIAM J. Imaging Sci. 6 (1) (2013) 199–225.
[9] A. Cohen, W. Dahmen, R. DeVore, Compressed sensing and best k-term approximation, J. Amer. Math. Soc. 22 (2009) 211–231.
[10] A. Conca, D. Edidin, M. Hering, C. Vinzant, Algebraic characterization of injectivity in phase retrieval, arXiv:1312.0158, 2013.
[11] Alexandre d'Aspremont, Laurent El Ghaoui, Testing the nullspace property using semidefinite programming, Math. Program., Ser. B 127 (2011) 123–144.
[12] D.L. Donoho, Compressed sensing, IEEE Trans. Inform. Theory 52 (4) (2006) 1289–1306.
[13] D.L. Donoho, X. Huo, Uncertainty principles and ideal atomic decompositions, IEEE Trans. Inform. Theory 47 (2001) 2845–2862.
[14] M. Elad, A.M. Bruckstein, A generalized uncertainty principle and sparse representation in pairs of bases, IEEE Trans. Inform. Theory 48 (2002) 2558–2567.
[15] R. Gribonval, M. Nielsen, Sparse representations in unions of bases, IEEE Trans. Inform. Theory 49 (2003) 3320–3325.
[16] Xiaodong Li, V. Voroninski, Sparse signal recovery from quadratic measurements via convex programming, SIAM J. Math. Anal. 45 (5) (2013) 3019–3033.
[17] M. Moravec, J. Romberg, R. Baraniuk, Compressive phase retrieval, in: Proceedings of SPIE, International Society for Optics and Photonics, 2007.
[18] H. Ohlsson, Y.C. Eldar, On conditions for uniqueness in sparse phase retrieval, arXiv:1308.5447.
[19] A. Pinkus, On L^1-Approximation, Cambridge Tracts in Math., vol. 93, Cambridge University Press, Cambridge, 1989.
[20] P. Schniter, S. Rangan, Compressive phase retrieval via generalized approximate message passing, in: Proceedings of Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, Oct. 2012, 2012.
[21] Andreas M. Tillmann, Marc E. Pfetsch, The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing, IEEE Trans. Inform. Theory 60 (2014) 1248–1259.
[22] Zai Yang, Cishen Zhang, Lihua Xie, Robust compressive phase retrieval via L1 minimization with application to image reconstruction, arXiv:1302.0081.

Linear Algebra and its Applications 449 (2014) 475–499

Contents lists available at ScienceDirect

Linear Algebra and its Applications

www.elsevier.com/locate/laa

Phase retrieval from very few measurements

Matthew Fickus [a], Dustin G. Mixon [a,*], Aaron A. Nelson [a], Yang Wang [b]

[a] *Department of Mathematics and Statistics, Air Force Institute of Technology, Wright–Patterson AFB, OH 45433, USA*
[b] *Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA*

ARTICLE INFO

Article history:
Received 14 September 2013
Accepted 4 February 2014
Available online 17 March 2014
Submitted by V. Mehrmann

MSC:
42C15
68Q17

Keywords:
Phase retrieval
Informationally complete
Unit norm tight frames
Computational complexity

ABSTRACT

In many applications, signals are measured according to a linear process, but the phases of these measurements are often unreliable or not available. To reconstruct the signal, one must perform a process known as phase retrieval. This paper focuses on completely determining signals with as few intensity measurements as possible, and on efficient phase retrieval algorithms from such measurements. For the case of complex M-dimensional signals, we construct a measurement ensemble of size $4M - 4$ which yields injective intensity measurements; this is conjectured to be the smallest such ensemble. For the case of real signals, we devise a theory of "almost" injective intensity measurements, and we characterize such ensembles. Later, we show that phase retrieval from $M + 1$ almost injective intensity measurements is NP-hard, indicating that computationally efficient phase retrieval must come at the price of measurement redundancy.

Published by Elsevier Inc.

* Corresponding author.
E-mail address: dustin.mixon@afit.edu (D.G. Mixon).

http://dx.doi.org/10.1016/j.laa.2014.02.011
0024-3795/Published by Elsevier Inc.

1. Introduction

Given an ensemble $\Phi = \{\varphi_n\}_{n=1}^N$ of M-dimensional vectors (real or complex), the *phase retrieval problem* is to recover a signal x from the *intensity measurements* $\mathcal{A}(x) := \{|\langle x, \varphi_n\rangle|^2\}_{n=1}^N$. Note that for any scalar ω of unit modulus, $\mathcal{A}(\omega x) = \mathcal{A}(x)$, and so the best one can hope to do is recover the set of signals $\{\omega x\colon\ |\omega| = 1\}$. Intensity measurements arise in a number of applications in which phase is either unreliable or not available, such as diffractive imaging [10,30,34,35] and optics [21,35,41]. For example, in high-power coherent diffractive imaging, only the intensities of diffracted X-rays can be recorded, and so to reconstruct material density profiles one must obtain the lost phase information after the fact [10]. Intensity measurements also appear in quantum state tomography when measuring a rank-1 quantum state using a positive operator-valued measure (POVM) consisting of rank-1 elements [27,28,31]. In most of these applications it is desirable to perform phase retrieval from as few measurements as possible, since increasing N invariably makes the measurement process more expensive or time consuming.

Recently, there has been a lot of work on algorithmic phase retrieval. For example, by viewing intensity measurements as Hilbert–Schmidt inner products between rank-1 operators [3,14], phase retrieval can be formulated as a low-rank matrix recovery problem [12, 18,23,39], and with this formulation phase retrieval is possible from $N = O(M)$ intensity measurements [13]. Another approach is to exploit the polarization identity along with expander graphs to design a measurement ensemble and apply spectral methods to perform phase retrieval [1,6]. One can also formulate phase retrieval in terms of MaxCut, and solvers for this formulation are equivalent to a popular solver (PhaseLift) for the matrix recovery formulation [38,40]. While this recent work has focused on stable and efficient phase retrieval from asymptotically few measurements (namely, $N = O(M)$), the present paper focuses on injectivity and algorithmic efficiency with the absolute minimum number of measurements.

In the next section, we construct an ensemble of $N = 4M - 4$ measurement vectors in $\mathbb{C}^M$ which yield injective intensity measurements. This is the second known injective ensemble of this size (the first is due to Bodmann and Hammen [9]), and it is conjectured to be the smallest-possible injective ensemble [5]. The same conjecture suggests that $4M-4$ generic measurement vectors yield injectivity (that is, there exists a measure-zero set of ensembles of $4M - 4$ vectors such that every ensemble of $4M - 4$ vectors outside of this set yields injectivity). The following summarizes what is currently known about the so-called "$4M - 4$ conjecture":

- The conjecture holds for $M = 2$ and $M = 2^m + 1$, $m = 1, 2, 3, \ldots$ [19] (cf. [5]).
- If $N < 4M - 2\alpha(M - 1) - 3$, then $\mathcal{A}$ is not injective [31]; here, $\alpha(M - 1) \leqslant \log_2 M$ denotes the number of 1's in the binary expansion of $M - 1$.
- For each $M \geqslant 2$, there exists an ensemble Φ of $N = 4M - 4$ measurement vectors such that $\mathcal{A}$ is injective [9] (see also Section 2 of this paper).
- If $N \geqslant 4M - 4$, then $\mathcal{A}$ is injective for generic Φ [19] (cf. [4]).

Bodmann and Hammen [9] leverage the Dirichlet kernel and the Cayley map to prove injectivity of their ensemble, but it is unclear whether phase retrieval is algorithmically feasible from their ensemble. By contrast, for the ensemble in this paper, we use basic ideas from harmonic analysis over cyclic groups to devise a corresponding phase retrieval algorithm, and we demonstrate injectivity in Theorem 6 by proving that the algorithm recovers any noiseless signal up to global phase.

In Section 3, we devise a theory of ensembles for which the corresponding intensity measurements are "almost" injective, that is, $\mathcal{A}^{-1}(\mathcal{A}(x)) = \{\omega x\colon\ |\omega| = 1\}$ for almost every x. In this section, we focus on the real case, meaning phase retrieval is up to a global sign factor $\omega = \pm 1$, and our approach is inspired by the characterization of injectivity in the real case by Balan, Casazza and Edidin [4]. After characterizing almost injectivity in the real case, we find a particularly satisfying sufficient condition for almost injectivity: that Φ forms a unit norm tight frame with M and N relatively prime. Characterizing almost injectivity in the complex case remains an open problem.

We conclude with Section 4, in which we consider algorithmic phase retrieval in the real case from $N = M+1$ almost injective intensity measurements. Specifically, we show that phase retrieval in this case is NP-hard by reduction from the subset sum problem. The hardness of phase retrieval in this minimal case suggests a new problem for phase retrieval: What is the smallest C for which there exists a family of ensembles of size $N = CM + o(M)$ such that phase retrieval can be performed in polynomial time?

2. $4M - 4$ injective intensity measurements

In this section, we provide an ensemble of $4M - 4$ measurement vectors which yield injective intensity measurements for $\mathbb{C}^M$. The vectors in our ensemble are modulated discrete cosine functions, and they are explicitly constructed at the end of this section. We start here by motivating our construction, specifically by identifying the significance of *circular autocorrelation*, which we define in (1) below.

Consider the P-dimensional complex vector space $\ell(\mathbb{Z}_P) := \{u : \mathbb{Z} \to \mathbb{C}\colon\ u[p+P] = u[p],\ \forall p \in \mathbb{Z}\}$. The discrete Fourier basis in $\ell(\mathbb{Z}_P)$ is the sequence of P vectors $\{f_q\}_{q\in\mathbb{Z}_P}$ defined by $f_q[p] := e^{2\pi ipq/P}$ (the notation "$q \in \mathbb{Z}_P$" is taken to mean a set of coset representatives of $\mathbb{Z}$ with respect to the subgroup $P\mathbb{Z}$). The discrete Fourier transform (DFT) on $\mathbb{Z}_P$ is $F^* : \ell(\mathbb{Z}_P) \to \ell(\mathbb{Z}_P)$, with corresponding inverse DFT $(F^*)^{-1} = \frac{1}{P}F$, defined by

$$\big(F^*u\big)[q] = \langle u, f_q\rangle = \sum_{p\in\mathbb{Z}_P} u[p]e^{-2\pi ipq/P}$$

$$(Fv)[p] = \sum_{q\in\mathbb{Z}_P} v[q]f_q[p] = \sum_{q\in\mathbb{Z}_P} v[q]e^{2\pi ipq/P}.$$

Now let $T^p : \ell(\mathbb{Z}_P) \to \ell(\mathbb{Z}_P)$ be the translation operator $(T^pu)[p'] := u[p'-p]$. The circular autocorrelation of u is then $\mathrm{CirAut}(u) \in \ell(\mathbb{Z}_P)$, defined entrywise by

$$\mathrm{CirAut}(u)[p] := \big\langle u, T^pu\big\rangle = \sum_{p'\in\mathbb{Z}_P} u\big[p'\big]\overline{u\big[p'-p\big]}. \tag{1}$$

Consider the DFT of a circular autocorrelation:

$$\begin{aligned}\big(F^*\operatorname{CirAut}(u)\big)[q] &= \sum_{p\in\mathbb{Z}_P}\sum_{p'\in\mathbb{Z}_P} u\big[p'\big]\overline{u\big[p'-p\big]}\,e^{-2\pi ipq/P}\\ &= \sum_{p'\in\mathbb{Z}_P} u\big[p'\big]e^{-2\pi ip'q/P}\overline{\bigg(\sum_{p\in\mathbb{Z}_P} u\big[p'-p\big]e^{-2\pi i(p'-p)q/P}\bigg)}\\ &= \sum_{p'\in\mathbb{Z}_P} u\big[p'\big]e^{-2\pi ip'q/P}\overline{\bigg(\sum_{p''\in\mathbb{Z}_P} u\big[p''\big]e^{-2\pi ip''q/P}\bigg)} = \big|\langle u,f_q\rangle\big|^2. \qquad (2)\end{aligned}$$

As such, if one has the intensity measurements $\{|\langle u,f_q\rangle|^2\}_{q\in\mathbb{Z}_P}$, then one may compute the circular autocorrelation $\operatorname{CirAut}(u)$ by applying the inverse DFT. In order to perform phase retrieval from $\{|\langle u,f_q\rangle|^2\}_{q\in\mathbb{Z}_P}$, it therefore suffices to determine u from $\operatorname{CirAut}(u)$. This is the motivation for our approach in this section.

To see how to "invert" CirAut, let's consider an example. Take $x=(a,b,c)\in\mathbb{C}^3$ and consider the circular autocorrelation of x as a signal in $\ell(\mathbb{Z}_3)$:

$$\operatorname{CirAut}(x) = \big(|a|^2+|b|^2+|c|^2, a\bar{c}+b\bar{a}+c\bar{b}, a\bar{b}+b\bar{c}+c\bar{a}\big).$$

Notice that every entry of $\operatorname{CirAut}(x)$ is a nonlinear combination of the entries of x, from which it is unclear how to compute the entries of x. To simplify the structure, we pad x with zeros and enforce even symmetry; then the circular autocorrelation of $u:=(2a,b,c,0,0,0,0,c,b)\in\ell(\mathbb{Z}_9)$ is

$$\begin{aligned}\operatorname{CirAut}(u) = \big(&4|a|^2+|b|^2+|c|^2, 2\operatorname{Re}(2a\bar{b}+b\bar{c}), |b|^2+4\operatorname{Re}(a\bar{c}), 2\operatorname{Re}(b\bar{c}), |c|^2,\\ &|c|^2, 2\operatorname{Re}(b\bar{c}), |b|^2+4\operatorname{Re}(a\bar{c}), 2\operatorname{Re}(2a\bar{b}+b\bar{c})\big). \qquad (3)\end{aligned}$$

Although it still appears rather complicated, this circular autocorrelation actually lends itself well to recovering the entries of x.

Before explaining this further, first note that $9=4(3)-3$, and we can generalize our mapping $x\mapsto u$ by sending vectors in $\mathbb{C}^M$ to members of $\ell(\mathbb{Z}_{4M-3})$. To make this clear, consider the reversal operator $R:\ell(\mathbb{Z}_P)\to\ell(\mathbb{Z}_P)$ defined by $(Ru)[p]=u[-p]$. Then given a vector $x\in\mathbb{C}^M$, padding with zeros and enforcing even symmetry is equivalent to embedding x in $\ell(\mathbb{Z}_{4M-3})$ by appending $3M-3$ zeros to x and then taking $u=x+Rx\in\ell(\mathbb{Z}_{4M-3})$. (From this point forward we use x to represent both the original signal in $\mathbb{C}^M$ and the version of x embedded in $\ell(\mathbb{Z}_{4M-3})$ via zero-padding; the distinction will be clear from context.) Computing $x\in\mathbb{C}^M$ then reduces to determining the first M entries of $x\in\ell(\mathbb{Z}_{4M-3})$ from $\operatorname{CirAut}(x+Rx)$. If x is completely real-valued, then this is indeed possible. For instance, consider the circular autocorrelation (3). If the entries of x are all real, then this becomes

$$\begin{aligned}\operatorname{CirAut}(x+Rx) = \big(&4a^2+b^2+c^2, 4ab+2bc, b^2+4ac, 2bc, c^2, c^2, 2bc,\\ &b^2+4ac, 4ab+2bc\big).\end{aligned}$$

Since $\mathrm{CirAut}(x + Rx)[4] = c^2$, we simply take a square root to obtain c up to a sign. Assuming c is nonzero, we then divide $\mathrm{CirAut}(x + Rx)[3]$ by $2c$ to determine b up to the same sign. Then subtracting b^2 from $\mathrm{CirAut}(x + Rx)[2]$ and dividing by $4c$ gives a up to the same sign.

From this example, we see that the process of recovering the entries of x from $\mathrm{CirAut}(x + Rx)$ is iterative, working backward through its first $2M - 2$ entries. But what happens if c is zero? Fortunately, our process doesn't break: In this case, we have

$$\mathrm{CirAut}(x + Rx) = \big(4a^2 + b^2, 4ab, b^2, 0, 0, 0, 0, b^2, 4ab\big).$$

Thus, we need only start with $\mathrm{CirAut}(x + Rx)[2]$ to determine the remaining entries of x up to a sign. This observation brings to light the important role of the last nonzero entry of x in our iteration. The relationship between this coordinate and the entries of $\mathrm{CirAut}(x + Rx)$ will become more rigorous later.

The above example illustrated how a real signal x is determined by $\mathrm{CirAut}(x + Rx)$. A complex-valued signal, on the other hand, is not completely determined from $\mathrm{CirAut}(x + Rx)$. Luckily, this can be fixed by introducing a second vector in $\ell(\mathbb{Z}_{4M-3})$ obtained from x, and we will demonstrate this later, but for now we focus on $x + Rx$. To this end, let's first take a closer look at the entries of $\mathrm{CirAut}(x + Rx)$. Since this circular autocorrelation has even symmetry by construction, we need only consider all entries of $\mathrm{CirAut}(x + Rx)$ up to index $2M - 2$. This leads to the following lemma:

Lemma 1. *Let x denote an M-dimensional complex signal embedded in $\ell(\mathbb{Z}_{4M-3})$ such that $x[p] = 0$ for all $p = M, \ldots, 4M - 4$. Then $\mathrm{CirAut}(x + Rx)[p] = 2\,\mathrm{Re}\langle x, T^p x\rangle + \langle x, RT^{-p}x\rangle$ for all $p = 1, \ldots, 2M - 2$.*

Proof. First note that by the definition of the circular autocorrelation in (1) we have

$$\mathrm{CirAut}(x + Rx)[p] = \big\langle x + Rx, T^p(x + Rx)\big\rangle = 2\,\mathrm{Re}\big\langle x, T^p x\big\rangle + \big\langle x, RT^{-p}x\big\rangle + \big\langle x, RT^p x\big\rangle.$$

Thus, to complete the proof it suffices to show that $\langle x, RT^p x\rangle = 0$ for all $p = 1, \ldots, 2M - 2$. Since x is only nonzero in its first M entries, we have

$$\big\langle x, RT^p x\big\rangle = \sum_{p'=0}^{M-1} x\big[p'\big]\overline{\big(RT^p x\big)\big[p'\big]} = \sum_{p'=0}^{M-1} x\big[p'\big]\overline{\big(T^p x\big)\big[-p'\big]} = \sum_{p'=0}^{M-1} x\big[p'\big]\overline{x\big[-p' - p\big]},$$

where the summand is zero whenever $-p' - p \notin [0, M - 1]$ modulo $4M - 3$. This is equivalent to having $-p$ not lie in the Minkowski sum $p' + [0, M - 1]$, and since $p' \in [0, M - 1]$ we see that $\langle x, RT^p x\rangle = 0$ for all $p = 1, \ldots, 2M - 2$. □

As a consequence of Lemma 1, the following theorem expresses the entries of $\mathrm{CirAut}(x + Rx)$ in terms of the entries of x:

Theorem 2. *Let x denote an M-dimensional complex signal embedded in $\ell(\mathbb{Z}_{4M-3})$ such that $x[p] = 0$ for all $p = M, \ldots, 4M-4$. Then we have*

$$\mathrm{CirAut}(x + Rx)[p] = \begin{cases} 2\,\mathrm{Re}\left(\displaystyle\sum_{p'=\frac{p+1}{2}}^{M-1} x[p'] \left(\overline{x[p'-p]} + \overline{x[p-p']} \right) \right) \\ \quad \text{if } p \text{ is odd} \\ 2\,\mathrm{Re}\left(\displaystyle\sum_{p'=\frac{p}{2}+1}^{M-1} x[p'] \left(\overline{x[p'-p]} + \overline{x[p-p']} \right) \right) + \left| x\left[\frac{p}{2}\right] \right|^2 \\ \quad \text{if } p \text{ is even} \end{cases} \tag{4}$$

for all $p = 1, \ldots, 2M-2$.

Proof. We first use Lemma 1 to get

$$\begin{aligned} \mathrm{CirAut}(x + Rx)[p] &= 2\,\mathrm{Re}\langle x, T^p x \rangle + \langle x, RT^{-p} x \rangle \\ &= 2\,\mathrm{Re}\left(\sum_{p'=0}^{M-1} x[p'] \overline{x[p'-p]} \right) + \sum_{p'=0}^{M-1} x[p'] \overline{x[p-p']} \\ &= 2\,\mathrm{Re}\left(\sum_{p'=p}^{M-1} x[p'] \overline{x[p'-p]} \right) + \sum_{p'=\max\{p-(M-1),0\}}^{\min\{p,M-1\}} x[p'] \overline{x[p-p']}, \end{aligned} \tag{5}$$

where the last equality takes into account that the first summand is nonzero only when $p' - p \in [0, M-1]$ and the second summand is nonzero only when $p - p' \in [0, M-1]$, i.e., when $p' \in [p, p + (M-1)]$ and $p' \in [p - (M-1), p]$, respectively. To continue, we divide our analysis into cases.

For $p = 1, \ldots, M-1$, (5) gives

$$\mathrm{CirAut}(x + Rx)[p] = 2\,\mathrm{Re}\left(\sum_{p'=p}^{M-1} x[p'] \overline{x[p'-p]} \right) + \sum_{p'=0}^{p} x[p'] \overline{x[p-p']}. \tag{6}$$

If p is odd we can then write

$$\begin{aligned} \sum_{p'=0}^{p} x[p'] \overline{x[p-p']} &= \sum_{p'=0}^{\frac{p-1}{2}} x[p'] \overline{x[p-p']} + \sum_{p'=\frac{p+1}{2}}^{p} x[p'] \overline{x[p-p']} \\ &= \sum_{p''=\frac{p+1}{2}}^{p} x[p-p''] \overline{x[p'']} + \sum_{p'=\frac{p+1}{2}}^{p} x[p'] \overline{x[p-p']} \\ &= 2\,\mathrm{Re}\left(\sum_{p'=\frac{p+1}{2}}^{p} x[p'] \overline{x[p-p']} \right), \end{aligned} \tag{7}$$

while if p is even we similarly write

$$\sum_{p'=0}^{p} x[p']\overline{x[p-p']} = 2\,\mathrm{Re}\left(\sum_{p'=\frac{p}{2}+1}^{p} x[p']\overline{x[p-p']}\right) + \left|x\left[\frac{p}{2}\right]\right|^2. \tag{8}$$

Substituting (7) and (8) into (6) then gives (4).

For the remaining case, $p = M, \ldots, 2M-2$ and (5) gives

$$\mathrm{CirAut}(x+Rx)[p] = \sum_{p'=p-(M-1)}^{M-1} x[p']\overline{x[p-p']}. \tag{9}$$

Similar to the previous case, taking p to be odd yields

$$\sum_{p'=p-(M-1)}^{M-1} x[p']\overline{x[p-p']} = 2\,\mathrm{Re}\left(\sum_{p'=\frac{p+1}{2}}^{M-1} x[p']\overline{x[p-p']}\right), \tag{10}$$

while taking p to be even yields

$$\sum_{p'=p-(M-1)}^{M-1} x[p']\overline{x[p-p']} = 2\,\mathrm{Re}\left(\sum_{p'=\frac{p}{2}+1}^{M-1} x[p']\overline{x[p-p']}\right) + \left|x\left[\frac{p}{2}\right]\right|^2, \tag{11}$$

and substituting (10) and (11) into (9) also gives (4). □

Notice (4) shows that each member of $\{\mathrm{CirAut}(x+Rx)[p]\}_{p=1}^{2M-2}$ can be written as a combination of the first M entries of x, but only those at or beyond the $\lceil\frac{p}{2}\rceil$th index. As such, the index of the last nonzero entry of x is closely related to that of the last nonzero entry of $\{\mathrm{CirAut}(x+Rx)[p]\}_{p=1}^{2M-2}$. This corresponds to our observation earlier in the case of $x \in \mathbb{R}^3$ where the third coordinate was assumed to be zero. We identify the relationship between the locations of these nonzero entries in the following lemma:

Lemma 3. *Let x denote an M-dimensional complex signal embedded in $\ell(\mathbb{Z}_{4M-3})$ such that $x[p] = 0$ for all $p = M, \ldots, 4M-4$. Then the last nonzero entry of $\{\mathrm{CirAut}(x+Rx)[p]\}_{p=0}^{2M-2}$ has index $p = 2q$, where q is the index of the last nonzero entry of x.*

Proof. If $q \geqslant 1$, then (4) gives that $\mathrm{CirAut}(x+Rx)[2q] = |x[q]|^2 \neq 0$. Note that since $x[p'] = 0$ for every $p' > q$, (4) also gives that $\mathrm{CirAut}(x+Rx)[p] = 0$ for every $p > 2q$. For the remaining case where $q = 0$, (4) immediately gives that $\mathrm{CirAut}(x+Rx)[p] = 0$ for every $p \geqslant 1$. To show that $\mathrm{CirAut}(x+Rx)[0] \neq 0$ in this case, we apply the definition of circular autocorrelation (1):

$$\mathrm{CirAut}(x+Rx)[0] = \langle x+Rx, x+Rx\rangle = \|x+Rx\|^2 = \left|2x[0]\right|^2 \neq 0,$$

where the last equality uses the fact that x is only supported at 0 since $q = 0$. □

As previously mentioned, we are unable to recover the entries of a complex signal x solely from $\mathrm{CirAut}(x + Rx)$. One way to address this is to rotate the entries of x in the complex plane and also take the circular autocorrelation of this modified signal. If we rotate by an angle which is not an integer multiple of π, this will produce new entries which are linearly independent from the corresponding entries of x when viewed as vectors in the complex plane. As we will see, the problem of recovering the entries of x then reduces to solving a linear system.

Take any $(4M-3)\times(4M-3)$ diagonal modulation operator E whose diagonal entries $\{\omega_k\}_{k=0}^{4M-4}$ are of unit modulus satisfying $\omega_j\overline{\omega_k} \notin \mathbb{R}$ for all $j \neq k$ and consider the new vector $Ex \in \ell(\mathbb{Z}_{4M-3})$. Then Theorem 2 gives

$$\mathrm{CirAut}(Ex + REx)[p] = \begin{cases} 2\,\mathrm{Re}\left(\displaystyle\sum_{p'=\frac{p+1}{2}}^{M-1} \omega_{p'}x\big[p'\big]\big(\overline{\omega_{p'-p}}\,\overline{x\big[p'-p\big]} + \overline{\omega_{p-p'}}\,\overline{x\big[p-p'\big]}\,\big)\right) \\ \quad \text{if } p \text{ is odd} \\ 2\,\mathrm{Re}\left(\displaystyle\sum_{p'=\frac{p}{2}+1}^{M-1} \omega_{p'}x\big[p'\big]\big(\overline{\omega_{p'-p}}\,\overline{x\big[p'-p\big]} + \overline{\omega_{p-p'}}\,\overline{x\big[p-p'\big]}\,\big)\right) + \left|x\left[\frac{p}{2}\right]\right|^2 \\ \quad \text{if } p \text{ is even} \end{cases} \tag{12}$$

for all $p = 1, \ldots, 2M-2$. We will see that (4) and (12) together allow us to solve for the entries of x (up to a global phase factor) by working iteratively backward through the entries of $\mathrm{CirAut}(x + Rx)$ and $\mathrm{CirAut}(Ex + REx)$. As alluded to earlier, each entry index forms a linear system which can be solved using the following lemma:

Lemma 4. *Let $a, b \in \mathbb{C} \setminus \{0\}$ and $\omega \in \mathbb{C} \setminus \mathbb{R}$ with $|\omega| = 1$. Then*

$$b = \frac{i}{\bar{a}\,\mathrm{Im}(\omega)}\big(\mathrm{Re}(\omega a\bar{b}) - \omega\,\mathrm{Re}(a\bar{b})\big). \tag{13}$$

Proof. After some manipulation, we have

$$\begin{aligned} \mathrm{Re}(\omega a\bar{b}) - \omega\,\mathrm{Re}(a\bar{b}) &= \mathrm{Re}(\omega)\,\mathrm{Re}(a\bar{b}) - \mathrm{Im}(\omega)\,\mathrm{Im}(a\bar{b}) - \omega\,\mathrm{Re}(a\bar{b}) \\ &= -i\,\mathrm{Im}(\omega)\big(\mathrm{Re}(a\bar{b}) - i\,\mathrm{Im}(a\bar{b})\big) = -i\bar{a}b\,\mathrm{Im}(\omega). \end{aligned}$$

Rearranging then yields the desired result. □

We now use this lemma to describe how to recover x up to global phase. By Lemma 3, the last nonzero entry of $\{\mathrm{CirAut}(x + Rx)[p]\}_{p=0}^{2M-2}$ has index $p = 2q$, where q indexes the last nonzero entry of x. As such, we know that $x[k] = 0$ for every $k > q$, and $x[q]$ can be estimated up to a phase factor ($\hat{x}[q] = e^{i\psi}x[q]$) by taking the square root of $\mathrm{CirAut}(x + Rx)[2q] = |x[q]|^2$ (we will verify this soon, but this corresponds to the examples we have seen so far). Next, if we know $\mathrm{Re}(x[q]\overline{x[k]})$ and $\mathrm{Re}(\omega_q\overline{\omega_k}x[q]\overline{x[k]})$ for some $k < q$, then we can use these to estimate $x[k]$:

$$\hat{x}[k] := \frac{i}{\overline{\hat{x}[q]}\operatorname{Im}(\omega_q\overline{\omega_k})}\big(\operatorname{Re}\big(\omega_q\overline{\omega_k}x[q]\overline{x[k]}\big) - \omega_q\overline{\omega_k}\operatorname{Re}\big(x[q]\overline{x[k]}\big)\big) = e^{i\psi}x[k], \tag{14}$$

where the last equality follows from substituting $a = x[q]$, $b = x[k]$ and $\omega = \omega_q\overline{\omega_k}$ into (13). Overall, once we know $x[q]$ up to phase, then we can find $x[k]$ relative to this same phase for each $k = 0, \ldots, q-1$, provided we know $\operatorname{Re}(x[q]\overline{x[k]})$ and $\operatorname{Re}(\omega_q\overline{\omega_k}x[q]\overline{x[k]})$ for these k's. Thankfully, these values can be determined from the entries of $\operatorname{CirAut}(x + Rx)$ and $\operatorname{CirAut}(Ex + REx)$:

Theorem 5. *Let x denote an M-dimensional complex signal embedded in $\ell(\mathbb{Z}_{4M-3})$ such that $x[p] = 0$ for all $p = M, \ldots, 4M-4$ and E be a $(4M-3)\times(4M-3)$ diagonal modulation operator with diagonal entries $\{\omega_k\}_{k=0}^{4M-4}$ satisfying $|\omega_k| = 1$ for all $k = 0, \ldots, 4M-4$ and $\omega_j\overline{\omega_k} \notin \mathbb{R}$ for all $j \neq k$. Then x can be recovered up to a global phase factor from $\operatorname{CirAut}(x + Rx)$ and $\operatorname{CirAut}(Ex + REx)$.*

Proof. Letting q denote the index of the last nonzero entry of x, it suffices to estimate $\{x[k]\}_{k=0}^{q}$ up to a global phase factor. To this end, recall from Lemma 3 that the last nonzero entry of $\{\operatorname{CirAut}(x + Rx)[p]\}_{p=0}^{2M-2}$ has index $p = 2q$. If $q = 0$, then we have already seen that $\operatorname{CirAut}(x + Rx)[0] = 4|x[0]|^2$. Since there exists $\psi \in [0, 2\pi)$ such that $x[0] = e^{-i\psi}|x[0]|$, we may take $\hat{x}[0] := \frac{1}{2}\sqrt{\operatorname{CirAut}(x + Rx)[0]} = |x[0]| = e^{i\psi}x[0]$. Otherwise $q \in [1, M-1]$, and (4) gives

$$\operatorname{CirAut}(x + Rx)[2q] = \big|x[q]\big|^2 + 2\operatorname{Re}\left(\sum_{p'=q+1}^{M-1} x\big[p'\big]\big(\overline{x\big[p'-2q\big]} + \overline{x\big[2q-p'\big]}\big)\right) = \big|x[q]\big|^2.$$

Thus, taking $\hat{x}[q] := \sqrt{\operatorname{CirAut}(x + Rx)[2q]} = |x[q]|$ gives us $\hat{x}[q] = e^{i\psi}x[q]$ for some $\psi \in [0, 2\pi)$.

In the case where $q = 1$, all that remains to determine is $\hat{x}[0]$, a calculation which we save for the end of the proof. For now, suppose $q \geqslant 2$. Since we already know $\hat{x}[q] = e^{i\psi}x[q]$, we would like to determine $\hat{x}[k]$ for $k = 1, \ldots, q-1$. To this end, take $r \in [0, q-2]$ and suppose we have $\hat{x}[k] = e^{i\psi}x[k]$ for all $k = q-r, \ldots, q$. If we can obtain $\hat{x}[q-(r+1)]$ up to the same phase from this information, then working iteratively from $r = 0$ to $r = q-2$ will give us $\hat{x}[k]$ up to global phase for all but the zeroth entry (which we address later). Note when r is even, (4) gives

$$\begin{aligned}
&\operatorname{CirAut}(x + Rx)\big[2q - (r+1)\big] \\
&\quad = 2\operatorname{Re}\left(\sum_{p'=q-\frac{r}{2}}^{q} x\big[p'\big]\big(\overline{x\big[p' - \big(2q-(r+1)\big)\big]} + \overline{x\big[\big(2q-(r+1)\big) - p'\big]}\big)\right) \\
&\quad = 2\operatorname{Re}\big(x[q]\overline{x\big[q-(r+1)\big]}\big) + 2\sum_{p'=q-\frac{r}{2}}^{q-1} \operatorname{Re}\big(x\big[p'\big]\overline{x\big[\big(2q-(r+1)\big) - p'\big]}\big),
\end{aligned}$$

where the last equality follows from the observation that $p'-(2q-(r+1)) \leqslant -q+(r+1) \leqslant -1$ over the range of the sum, meaning $x[p'-(2q-(r+1))]=0$ throughout the sum. Similarly when r is odd, (4) gives

$$\begin{aligned}&\operatorname{CirAut}(x+Rx)\big[2q-(r+1)\big]\\&\quad=2\operatorname{Re}\big(x[q]\overline{x\big[q-(r+1)\big]}\,\big)\\&\qquad+2\sum_{p'=q-\frac{r-1}{2}}^{q-1}\operatorname{Re}\big(x\big[p'\big]\overline{x\big[\big(2q-(r+1)\big)-p'\big]}\,\big)+\left|x\left[q-\frac{r+1}{2}\right]\right|^2.\end{aligned}$$

In either case, we can isolate $\operatorname{Re}(x[q]\overline{x[q-(r+1)]})$ to get an expression in terms of $\operatorname{CirAut}(x+Rx)[2q-(r+1)]$ and other terms of the form $\operatorname{Re}(x[k]\overline{x[k']})$ or $|x[k]|^2$ for $k,k'\in[q-r,q-1]$. By the induction hypothesis, we have $\hat{x}[k]=e^{i\psi}x[k]$ for $k=q-r,\ldots,q-1$, and so we can use these estimates to determine these other terms:

$$\operatorname{Re}\big(\hat{x}[k]\overline{\hat{x}\big[k'\big]}\big)=\operatorname{Re}\big(e^{i\psi}x[k]\overline{e^{i\psi}x\big[k'\big]}\big)=\operatorname{Re}\big(x[k]\overline{x\big[k'\big]}\big),\qquad\big|\hat{x}[k]\big|^2=\big|e^{i\psi}x[k]\big|^2=\big|x[k]\big|^2.$$

As such, we can use $\operatorname{CirAut}(x+Rx)[2q-(r+1)]$ along with the higher-indexed estimates $\hat{x}[k]$ to determine $\operatorname{Re}(x[q]\overline{x[q-(r+1)]})$. Similarly, we can use $\operatorname{CirAut}(Ex+REx)[2q-(r+1)]$ along with the higher-indexed estimates $\hat{x}[k]$ to determine $\operatorname{Re}(\omega_q\overline{\omega_{(q-(r+1))}}x[q]\overline{x[q-(r+1)]})$. We then plug these into (14), along with the estimate $\hat{x}[q]=e^{i\psi}x[q]$ (which is also available by the induction hypothesis), to get $\hat{x}[2q-(r+1)]=e^{i\psi}x[2q-(r+1)]$.

At this point, we have determined $\{x[k]\}_{k=1}^{q}$ up to a global phase factor whenever $q\geqslant 1$, and so it remains to find $\hat{x}[0]$. For this, note that when q is odd, (4) gives

$$\operatorname{CirAut}(x+Rx)[q]=4\operatorname{Re}\big(x[q]\overline{x[0]}\,\big)+2\sum_{p'=\frac{q+1}{2}}^{q-1}\operatorname{Re}\big(x\big[p'\big]\overline{x\big[q-p'\big]}\,\big),$$

while for even q, we have

$$\operatorname{CirAut}(x+Rx)[q]=4\operatorname{Re}\big(x[q]\overline{x[0]}\,\big)+2\sum_{p'=\frac{q}{2}+1}^{q-1}\operatorname{Re}\big(x\big[p'\big]\overline{x\big[q-p'\big]}\,\big)+\left|x\left[\frac{q}{2}\right]\right|^2.$$

As before, isolating $\operatorname{Re}(x[q]\overline{x[0]})$ in either case produces an expression in terms of $\operatorname{CirAut}(x+Rx)[q]$ and other terms of the form $\operatorname{Re}(x[k]\overline{x[k']})$ or $|x[k]|^2$ for $k,k'\in[1,q-1]$. These other terms can be calculated using the estimates $\{\hat{x}[k]\}_{k=1}^{q-1}$, and so we can also calculate $\operatorname{Re}(x[q]\overline{x[0]})$ from $\operatorname{CirAut}(x+Rx)[q]$. Similarly, we can calculate $\operatorname{Re}(\omega_q\overline{\omega_0}x[q]\overline{x[0]})$ from $\{\hat{x}[k]\}_{k=1}^{q-1}$ and $\operatorname{CirAut}(Ex+REx)[q]$, and plugging these into (14) along with $\hat{x}[q]$ produces the estimate $\hat{x}[0]=e^{i\psi}x[0]$. □

Theorem 5 establishes that it is possible to recover a signal $x \in \mathbb{C}^M$ up to a global phase factor from $\{\mathrm{CirAut}(x+Rx)\}_{q=0}^{2M-2}$ and $\{\mathrm{CirAut}(Ex+REx)\}_{q=0}^{2M-2}$. We now return to how these circular autocorrelations relate to intensity measurements. Recall from (2) that the DFT of the circular autocorrelation is the modulus squared of the DFT of the original signal: $(F^*\,\mathrm{CirAut}(u))[q] = |(F^*u)[q]|^2$. Also note that the DFT commutes with the reversal operator:

$$\big(F^*Ru\big)[q] = \sum_{p\in\mathbb{Z}_P} u[-p]e^{-2\pi ipq/P} = \sum_{p'\in\mathbb{Z}_P} u\big[p'\big]e^{-2\pi ip'(-q)/P} = \big(F^*u\big)[-q] = \big(RF^*u\big)[q].$$

With this, we can express $\mathrm{CirAut}(x + Rx)$ in terms of intensity measurements with a particular ensemble:

$$\begin{aligned}\big(F^*\,\mathrm{CirAut}(x+Rx)\big)[q] &= \big|\big(F^*(x+Rx)\big)[q]\big|^2\\ &= \big|\big(F^*x\big)[q] + \big(F^*Rx\big)[q]\big|^2 = \big|\big(F^*x\big)[q] + \big(F^*x\big)[-q]\big|^2\\ &= \big|\langle x, f_q + f_{-q}\rangle\big|^2.\end{aligned}$$

Defining the qth *discrete cosine function* $c_q \in \ell(\mathbb{Z}_{4M-3})$ by

$$c_q[p] := 2\cos\left(\frac{2\pi pq}{4M-3}\right) = e^{2\pi ipq/(4M-3)} + e^{-2\pi ipq/(4M-3)} = (f_q + f_{-q})[p],$$

this means that $(F^*\,\mathrm{CirAut}(x + Rx))[q] = |\langle x, c_q\rangle|^2$ for all $q \in \mathbb{Z}_{4M-3}$. Similarly, if we take the modulation matrix E to have diagonal entries $\omega_k = e^{2\pi ik/(2M-1)}$ for all $k = 0, \ldots, 4M-4$, we find

$$\big(F^*\,\mathrm{CirAut}(Ex+REx)\big)[q] = \big|\langle Ex, c_q\rangle\big|^2 = \big|\big\langle x, E^*c_q\big\rangle\big|^2.$$

Thus, coupling the DFT with Theorem 5 allows us to recover the signal x from $4M-2$ intensity measurements, namely with the ensemble $\{c_q\}_{q=0}^{2M-2} \cup \{E^*c_q\}_{q=0}^{2M-2}$. Note that since $x \in \ell(\mathbb{Z}_{4M-3})$ is actually a zero-padded version of $x \in \mathbb{C}^M$, we may view c_q and E^*c_q as members of $\mathbb{C}^M$ by discarding the entries indexed by $p = M, \ldots, 4M-4$.

Considering this section promised phase retrieval from only $4M-4$ intensity measurements, we must somehow find a way to discard two of these $4M-2$ measurement vectors. To do this, first note that

$$\begin{aligned}\mathrm{CirAut}(Ex+REx)[0] &= \|Ex+REx\|^2\\ &= \sum_{k\in\mathbb{Z}_{4M-3}} \big|e^{2\pi ik/(2M-1)}x[k] + e^{2\pi i(-k)/(2M-1)}x[-k]\big|^2\\ &= \sum_{k=-(2M-2)}^{-1} \big|e^{2\pi i(-k)/(2M-1)}x[-k]\big|^2 + \big|2x[0]\big|^2\end{aligned}$$

$$\begin{aligned}
&+ \sum_{k=1}^{2M-2} \left| e^{2\pi ik/(2M-1)} x[k] \right|^2 \\
&= \|x + Rx\|^2 \\
&= \mathrm{CirAut}(x + Rx)[0].
\end{aligned}$$

Moreover, we have

$$\begin{aligned}
\mathrm{CirAut}(Ex + REx)[2M-2] &= \sum_{k \in \mathbb{Z}_{4M-3}} (Ex + REx)[k] \overline{(Ex + REx)\big[k - (2M-2)\big]} \\
&= (Ex + REx)[M-1] \overline{(Ex + REx)\big[-(M-1)\big]} \\
&= (Ex + REx)[M-1] \overline{(Ex + REx)[M-1]},
\end{aligned}$$

where the last equality is by even symmetry. Since x is only supported on $k = 0, \ldots, M-1$, we then have

$$\begin{aligned}
\mathrm{CirAut}(Ex + REx)[2M-2] &= \left| (Ex + REx)[M-1] \right|^2 \\
&= \left| e^{2\pi i(M-1)/(2M-1)} x[M-1] \right. \\
&\quad \left. + e^{-2\pi i(M-1)/(2M-1)} x\big[-(M-1)\big] \right|^2 \\
&= \left| e^{2\pi i(M-1)/(2M-1)} x[M-1] \right|^2 = \left| x[M-1] \right|^2 \\
&= \mathrm{CirAut}(x + Rx)[2M-2].
\end{aligned}$$

Furthermore, the even symmetry of the circular autocorrelation also gives

$$\begin{aligned}
\mathrm{CirAut}(Ex + REx)\big[-(2M-2)\big] &= \mathrm{CirAut}(Ex + REx)[2M-2] \\
&= \mathrm{CirAut}(x + Rx)[2M-2] \\
&= \mathrm{CirAut}(x + Rx)\big[-(2M-2)\big].
\end{aligned}$$

These redundancies between $\mathrm{CirAut}(x + Rx)$ and $\mathrm{CirAut}(Ex + REx)$ indicate that we might be able to remove measurement vectors from our ensemble while maintaining our ability to perform phase retrieval. The following theorem confirms this suspicion:

Theorem 6. *Let $c_q \in \mathbb{C}^M$ be the truncated discrete cosine function defined by $c_q[p] := 2\cos(\frac{2\pi pq}{4M-3})$ for all $p = 0, \ldots, M-1$, and let E be the $M \times M$ diagonal modulation operator with diagonal entries $\omega_k = e^{2\pi ik/(2M-1)}$ for all $k = 0, \ldots, M-1$. Then the intensity measurement mapping $\mathcal{A} : \mathbb{C}^M/\mathbb{T} \to \mathbb{R}^{4M-4}$ defined by $\mathcal{A}(x) := \{|\langle x, c_q\rangle|^2\}_{q=0}^{2M-2} \cup \{|\langle x, E^* c_q\rangle|^2\}_{q=1}^{2M-3}$ is injective.*

Proof. Since Theorem 5 allows us to reconstruct any $x \in \mathbb{C}^M$ up to a global phase factor from the entries of $\mathrm{CirAut}(x + Rx)$ and $\mathrm{CirAut}(Ex + REx)$, it suffices to show that the intensity measurements $\{|\langle x, c_q\rangle|^2\}_{q=0}^{2M-2} \cup \{|\langle x, E^* c_q\rangle|^2\}_{q=1}^{2M-3}$ allow us to recover the

entries of these circular autocorrelations. To this end, recall from (2) that these quantities are related through the inverse DFT:

$$\mathrm{CirAut}(x+Rx) = \big(F^*\big)^{-1}\big\{\big|\langle x, c_q\rangle\big|^2\big\}_{q\in\mathbb{Z}_{4M-3}},$$
$$\mathrm{CirAut}(Ex+REx) = \big(F^*\big)^{-1}\big\{\big|\langle x, E^*c_q\rangle\big|^2\big\}_{q\in\mathbb{Z}_{4M-3}}.$$

Since we have $\{|\langle x, c_q\rangle|^2\}_{q=0}^{2M-2}$, we can exploit even symmetry to determine the rest of $\{|\langle x, c_q\rangle|^2\}_{q\in\mathbb{Z}_{4M-3}}$, and then apply the inverse DFT to get $\mathrm{CirAut}(x+Rx)$. Moreover, by the previous discussion, we also obtain the 0, $2M-2$, and $-(2M-2)$ entries of $\mathrm{CirAut}(Ex+REx)$ from the corresponding entries of $\mathrm{CirAut}(x+Rx)$. Organize this information about $\mathrm{CirAut}(Ex+REx)$ into a vector $w\in\ell(\mathbb{Z}_{4M-3})$ whose 0, $2M-2$, and $-(2M-2)$ entries come from $\mathrm{CirAut}(Ex+REx)$ and whose remaining entries are populated by even symmetry from $\{|\langle x, E^*c_q\rangle|^2\}_{q=1}^{2M-3}$. We can express w as a matrix-vector product $w = A\{|\langle x, E^*c_q\rangle|^2\}_{q\in\mathbb{Z}_{4M-3}}$, where A is the identity matrix with the 0, $2M-2$, and $-(2M-2)$ rows replaced by the corresponding rows of the inverse DFT matrix. To complete the proof, it suffices to show that the matrix A is invertible, since this would imply $\mathrm{CirAut}(Ex+REx) = (F^*)^{-1}A^{-1}w$.

Using the cofactor expansion, note that $\det(A)$ reduces to a determinant of a 3×3 submatrix of $(F^*)^{-1}$. Specifically, letting $\theta := 2\pi(2M-2)^2/(4M-3)$ we have

$$\begin{aligned}\det(A) = \det\left(\begin{bmatrix}1 & 1 & 1\\ 1 & e^{i\theta} & e^{-i\theta}\\ 1 & e^{-i\theta} & e^{i\theta}\end{bmatrix}\right) &= \big(e^{2i\theta}-e^{-2i\theta}\big)-\big(e^{i\theta}-e^{-i\theta}\big)+\big(e^{-i\theta}-e^{i\theta}\big)\\ &= \big(e^{i\theta}+e^{-i\theta}-2\big)\big(e^{i\theta}-e^{-i\theta}\big)\\ &= 4i\big(\cos(\theta)-1\big)\sin(\theta),\end{aligned}$$

and so A is invertible if and only if $\cos(\theta)-1\neq 0$ and $\sin(\theta)\neq 0$. This equivalent to having π not divide θ, and indeed, the ratio

$$\frac{\theta}{\pi} = \frac{2(2M-2)^2}{4M-3} = 2M-\frac{5}{2}+\frac{1}{2(4M-3)}$$

is not an integer because $M\geqslant 2$. As such, A is invertible. □

We conclude this section by summarizing our measurement design and phase retrieval procedure:

Measurement design

- Define the qth truncated discrete cosine function $c_q := \{2\cos(\frac{2\pi pq}{4M-3})\}_{p=0}^{M-1}$
- Define the $M\times M$ diagonal matrix E with entries $\omega_k := e^{2\pi ik/(2M-1)}$ for all $k = 0,\ldots,M-1$
- Take $\Phi := \{c_q\}_{q=0}^{2M-2}\cup\{E^*c_q\}_{q=1}^{2M-3}$

Phase retrieval procedure

- Calculate $\{|\langle x, c_q\rangle|^2\}_{q\in\mathbb{Z}_{4M-3}}$ from $\{|\langle x, c_q\rangle|^2\}_{q=0}^{2M-2}$ by even extension
- Calculate $\mathrm{CirAut}(x + Rx) = (F^*)^{-1}\{|\langle x, c_q\rangle|^2\}_{q\in\mathbb{Z}_{4M-3}}$
- Define $w \in \ell(\mathbb{Z}_{4M-3})$ so that its 0, $2M-2$, and $-(2M-2)$ entries are the corresponding entries in $\mathrm{CirAut}(x + Rx)$ and its remaining entries are populated by even symmetry from $\{|\langle x, E^*c_q\rangle|^2\}_{q=1}^{2M-3}$
- Define A to be the identity matrix with the 0, $2M-2$, and $-(2M-2)$ rows replaced by the corresponding rows of the inverse DFT matrix $(F^*)^{-1}$
- Calculate $\mathrm{CirAut}(Ex + REx) = (F^*)^{-1}A^{-1}w$
- Recover x up to global phase from $\mathrm{CirAut}(x + Rx)$ and $\mathrm{CirAut}(Ex + REx)$ using the process described in the proof of Theorem 5

3. Almost injectivity

While $4M+o(M)$ measurements are necessary and generically sufficient for injectivity in the complex case, one can save a factor of 2 in the number of measurements by slightly weakening the desired notion of injectivity [4,28]. To be explicit, we start with the following definition:

Definition 7. Consider $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$. The intensity measurement mapping $\mathcal{A}:\mathbb{R}^M/\{\pm 1\} \to \mathbb{R}^N$ defined by $(\mathcal{A}(x))(n) := |\langle x, \varphi_n\rangle|^2$ is said to be *almost injective* if $\mathcal{A}^{-1}(\mathcal{A}(x)) = \{\pm x\}$ for almost every $x \in \mathbb{R}^M$.

The above definition specifically treats the real case, but it can be similarly defined for the complex case in the obvious way. For the complex case, it is known that $2M$ measurements are necessary for almost injectivity [28], and that $2M$ generic measurements suffice [4] (cf. [27]); this is the factor-of-2 savings mentioned above. For the real case, it is also known how many measurements are necessary and generically sufficient for almost injectivity: $M + 1$ [4]. Like the complex case, this is also a factor-of-2 savings from the injectivity requirement: $2M - 1$. This requirement for injectivity in the real case follows from the following result from [4], which we prove here because the proof is short and inspires the remainder of this section:

Theorem 8. *Consider $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ and the intensity measurement mapping $\mathcal{A}:\mathbb{R}^M/\{\pm 1\} \to \mathbb{R}^N$ defined by $(\mathcal{A}(x))(n) := |\langle x, \varphi_n\rangle|^2$. Then $\mathcal{A}$ is injective if and only if for every $S \subseteq \{1, \ldots, N\}$, either $\{\varphi_n\}_{n\in S}$ or $\{\varphi_n\}_{n\in S^c}$ spans $\mathbb{R}^M$.*

Proof. We will prove both directions by obtaining the contrapositives.

($\Rightarrow$) Assume there exists $S \subseteq \{1, \ldots, N\}$ such that neither $\{\varphi_n\}_{n\in S}$ nor $\{\varphi_n\}_{n\in S^c}$ spans $\mathbb{R}^M$. This implies that there are nonzero vectors $u, v \in \mathbb{R}^M$ such that $\langle u, \varphi_n\rangle = 0$ for all $n \in S$ and $\langle v, \varphi_n\rangle = 0$ for all $n \in S^c$. For each n, we then have

$$\big|\langle u \pm v, \varphi_n\rangle\big|^2 = \big|\langle u, \varphi_n\rangle\big|^2 \pm 2\langle u, \varphi_n\rangle\langle v, \varphi_n\rangle + \big|\langle v, \varphi_n\rangle\big|^2 = \big|\langle u, \varphi_n\rangle\big|^2 + \big|\langle v, \varphi_n\rangle\big|^2.$$

Since $|\langle u+v, \varphi_n\rangle|^2 = |\langle u-v, \varphi_n\rangle|^2$ for every n, we have $\mathcal{A}(u+v) = \mathcal{A}(u-v)$. Moreover, u and v are nonzero by assumption, and so $u+v \neq \pm(u-v)$.

($\Leftarrow$) Assume that $\mathcal{A}$ is not injective. Then there exist vectors $x, y \in \mathbb{R}^M$ such that $x \neq \pm y$ and $\mathcal{A}(x) = \mathcal{A}(y)$. Taking $S := \{n\colon \langle x, \varphi_n\rangle = -\langle y, \varphi_n\rangle\}$, we have $\langle x+y, \varphi_n\rangle = 0$ for every $n \in S$. Otherwise when $n \in S^{\mathrm{c}}$, we have $\langle x, \varphi_n\rangle = \langle y, \varphi_n\rangle$ and so $\langle x-y, \varphi_n\rangle = 0$. Furthermore, both $x+y$ and $x-y$ are nontrivial since $x \neq \pm y$, and so neither $\{\varphi_n\}_{n\in S}$ nor $\{\varphi_n\}_{n\in S^{\mathrm{c}}}$ spans $\mathbb{R}^M$. $\square$

Similar to the above result, in this section, we characterize ensembles of measurement vectors which yield almost injective intensity measurements, and similar to the above proof, the basic idea behind our analysis is to consider sums and differences of signals with identical intensity measurements. Our characterization starts with the following lemma:

Lemma 9. *Consider $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ and the intensity measurement mapping $\mathcal{A}\colon \mathbb{R}^M/\{\pm 1\} \to \mathbb{R}^N$ defined by $(\mathcal{A}(x))(n) := |\langle x, \varphi_n\rangle|^2$. Then $\mathcal{A}$ is almost injective if and only if almost every $x \in \mathbb{R}^M$ is not in the Minkowski sum $\operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$ for all $S \subseteq \{1, \ldots, N\}$. More precisely, $\mathcal{A}^{-1}(\mathcal{A}(x)) = \{\pm x\}$ if and only if $x \notin \operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$ for any $S \subseteq \{1, \ldots, N\}$.*

Proof. By the definition of the mapping $\mathcal{A}$, for $x, y \in \mathbb{R}^M$ we have $\mathcal{A}(x) = \mathcal{A}(y)$ if and only if $|\langle x, \varphi_n\rangle| = |\langle y, \varphi_n\rangle|$ for all $n \in \{1, \ldots, N\}$. This occurs precisely when there is a subset $S \subseteq \{1, \ldots, N\}$ such that $\langle x, \varphi_n\rangle = -\langle y, \varphi_n\rangle$ for every $n \in S$ and $\langle x, \varphi_n\rangle = \langle y, \varphi_n\rangle$ for every $n \in S^{\mathrm{c}}$. Thus, $\mathcal{A}^{-1}(\mathcal{A}(x)) = \{\pm x\}$ if and only if for every $y \neq \pm x$ and for every $S \subseteq \{1, \ldots, N\}$, either there exists an $n \in S$ such that $\langle x+y, \varphi_n\rangle \neq 0$ or an $n \in S^{\mathrm{c}}$ such that $\langle x-y, \varphi_n\rangle \neq 0$. We claim that this occurs if and only if x is not in the Minkowski sum $\operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$ for all $S \subseteq \{1, \ldots, N\}$, which would complete the proof. We verify the claim by seeking the contrapositive in each direction.

($\Rightarrow$) Suppose $x \in \operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$. Then there exist $u \in \operatorname{span}(\Phi_S)^\perp \setminus \{0\}$ and $v \in \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$ such that $x = u + v$. Taking $y := u - v$, we see that $x + y = 2u \in \operatorname{span}(\Phi_S)^\perp \setminus \{0\}$ and $x - y = 2v \in \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$, which means that there is no $n \in S$ such that $\langle x+y, \varphi_n\rangle \neq 0$ nor $n \in S^{\mathrm{c}}$ such that $\langle x-y, \varphi_n\rangle \neq 0$. Furthermore, u and v are nonzero, and so $y \neq \pm x$.

($\Leftarrow$) Suppose $y \neq \pm x$ and for every $S \subseteq \{1, \ldots, N\}$ there is no $n \in S$ such that $\langle x+y, \varphi_n\rangle \neq 0$ nor $n \in S^{\mathrm{c}}$ such that $\langle x-y, \varphi_n\rangle \neq 0$. Then $x + y \in \operatorname{span}(\Phi_S)^\perp \setminus \{0\}$ and $x - y \in \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$. Since $x = \frac{1}{2}(x+y) + \frac{1}{2}(x-y)$, we have that $x \in \operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^{\mathrm{c}}})^\perp \setminus \{0\}$. $\square$

Theorem 10. *Consider $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ and the intensity measurement mapping $\mathcal{A}\colon \mathbb{R}^M/\{\pm 1\} \to \mathbb{R}^N$ defined by $(\mathcal{A}(x))(n) := |\langle x, \varphi_n\rangle|^2$. Suppose Φ spans $\mathbb{R}^M$ and each*

φ_n is nonzero. Then $\mathcal{A}$ is almost injective if and only if the Minkowski sum $\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp$ *is a proper subspace of* $\mathbb{R}^M$ *for each nonempty proper subset* $S \subseteq \{1,\dots,N\}$.

Note that the above result is not terribly surprising considering Lemma 9, as the new condition involves a simpler Minkowski sum in exchange for additional (reasonable and testable) assumptions on Φ. The proof of this theorem amounts to measuring the difference between the two Minkowski sums:

Proof of Theorem 10. First note that the spanning assumption on Φ implies

$$\operatorname{span}(\Phi_S)^\perp \cap \operatorname{span}(\Phi_{S^c})^\perp = \big(\operatorname{span}(\Phi_S) + \operatorname{span}(\Phi_{S^c})\big)^\perp = \operatorname{span}(\Phi)^\perp = \{0\},$$

and so one can prove the following identity:

$$\begin{aligned}&\operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^c})^\perp \setminus \{0\}\\ &\quad = \big(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big) \setminus \big(\operatorname{span}(\Phi_S)^\perp \cup \operatorname{span}(\Phi_{S^c})^\perp\big).\end{aligned} \tag{15}$$

From Lemma 9 we know that $\mathcal{A}$ is almost injective if and only if almost every $x \in \mathbb{R}^M$ is not in the Minkowski sum $\operatorname{span}(\Phi_S)^\perp \setminus \{0\} + \operatorname{span}(\Phi_{S^c})^\perp \setminus \{0\}$ for any $S \subseteq \{1,\dots,N\}$. In other words, the Lebesgue measure (which we denote by $\operatorname{Leb}[\cdot]$) of this Minkowski sum is zero for each $S \subseteq \{1,\dots,N\}$. By (15), this equivalently means that the Lebesgue measure of $(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp) \setminus (\operatorname{span}(\Phi_S)^\perp \cup \operatorname{span}(\Phi_{S^c})^\perp)$ is zero for each $S \subseteq \{1,\dots,N\}$. Since Φ spans $\mathbb{R}^M$, this set is empty (and therefore has Lebesgue measure zero) when $S = \emptyset$ or $S = \{1,\dots,N\}$. Also, since each φ_n is nonzero, we know that $\operatorname{span}(\Phi_S)^\perp$ and $\operatorname{span}(\Phi_{S^c})^\perp$ are proper subspaces of $\mathbb{R}^M$ whenever S is a nonempty proper subset of $\{1,\dots,N\}$, and so in these cases both subspaces must have Lebesgue measure zero. As such, we have that for every nonempty proper subset $S \subseteq \{1,\dots,N\}$,

$$\begin{aligned}&\operatorname{Leb}\big[\big(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big) \setminus \big(\operatorname{span}(\Phi_S)^\perp \cup \operatorname{span}(\Phi_{S^c})^\perp\big)\big]\\ &\quad \geqslant \operatorname{Leb}\big[\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big] - \operatorname{Leb}\big[\operatorname{span}(\Phi_S)^\perp\big] - \operatorname{Leb}\big[\operatorname{span}(\Phi_{S^c})^\perp\big]\\ &\quad = \operatorname{Leb}\big[\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big]\\ &\quad \geqslant \operatorname{Leb}\big[\big(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big) \setminus \big(\operatorname{span}(\Phi_S)^\perp \cup \operatorname{span}(\Phi_{S^c})^\perp\big)\big].\end{aligned}$$

In summary, $(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp) \setminus (\operatorname{span}(\Phi_S)^\perp \cup \operatorname{span}(\Phi_{S^c})^\perp)$ having Lebesgue measure zero for each $S \subseteq \{1,\dots,N\}$ is equivalent to $\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp$ having Lebesgue measure zero for each nonempty proper subset $S \subseteq \{1,\dots,N\}$, which in turn is equivalent to the Minkowski sum $\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp$ being a proper subspace of $\mathbb{R}^M$ for each nonempty proper subset $S \subseteq \{1,\dots,N\}$, as desired. □

At this point, consider the following stronger restatement of Theorem 10: "Suppose each φ_n is nonzero. Then $\mathcal{A}$ is almost injective if and only if Φ spans $\mathbb{R}^M$ and the

Minkowski sum $\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp$ is a proper subspace of $\mathbb{R}^M$ for each nonempty proper subset $S \subseteq \{1,\dots,N\}$." Note that we can move the spanning assumption into the condition because if Φ does not span, then we can decompose almost every $x \in \mathbb{R}^M$ as $x = u + v$ such that $u \in \operatorname{span}(\Phi)$ and $v \in \operatorname{span}(\Phi)^\perp$ with $v \neq 0$, and defining $y := u - v$ then gives $\mathcal{A}(y) = \mathcal{A}(x)$ despite the fact that $y \neq \pm x$. As for the assumption that the φ_n's are nonzero, we note that having $\varphi_n = 0$ amounts to having the nth entry of $\mathcal{A}(x)$ be zero for all x. As such, Φ yields almost injectivity precisely when the nonzero members of Φ together yield almost injectivity. With this identification, the stronger restatement of Theorem 10 above can be viewed as a complete characterization of almost injectivity. Next, we will replace the Minkowski sum condition with a rather elegant condition involving the ranks of Φ_S and Φ_{S^c}:

Theorem 11. *Consider $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ and the intensity measurement mapping $\mathcal{A}\colon \mathbb{R}^M/\{\pm 1\} \to \mathbb{R}^N$ defined by $(\mathcal{A}(x))(n) := |\langle x, \varphi_n\rangle|^2$. Suppose each φ_n is nonzero. Then $\mathcal{A}$ is almost injective if and only if Φ spans $\mathbb{R}^M$ and $\operatorname{rank}\Phi_S + \operatorname{rank}\Phi_{S^c} > M$ for each nonempty proper subset $S \subseteq \{1,\dots,N\}$.*

Proof. Considering the discussion after the proof of Theorem 10, it suffices to assume that Φ spans $\mathbb{R}^M$. Furthermore, considering Theorem 10, it suffices to characterize when $\dim(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp) < M$. By the inclusion–exclusion principle for subspaces, we have

$$\begin{aligned}&\dim\big(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big)\\ &\quad= \dim\big(\operatorname{span}(\Phi_S)^\perp\big) + \dim\big(\operatorname{span}(\Phi_{S^c})^\perp\big) - \dim\big(\operatorname{span}(\Phi_S)^\perp \cap \operatorname{span}(\Phi_{S^c})^\perp\big).\end{aligned}$$

Since Φ is assumed to span $\mathbb{R}^M$, we also have that $\operatorname{span}(\Phi_S)^\perp \cap \operatorname{span}(\Phi_{S^c})^\perp = \{0\}$, and so

$$\begin{aligned}\dim\big(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp\big) &= \big(M - \dim\big(\operatorname{span}(\Phi_S)\big)\big) + \big(M - \dim\big(\operatorname{span}(\Phi_{S^c})\big)\big) - 0\\ &= 2M - \operatorname{rank}\Phi_S - \operatorname{rank}\Phi_{S^c}.\end{aligned}$$

As such, $\dim(\operatorname{span}(\Phi_S)^\perp + \operatorname{span}(\Phi_{S^c})^\perp) < M$ precisely when $\operatorname{rank}\Phi_S + \operatorname{rank}\Phi_{S^c} > M$. □

At this point, we point out some interesting consequences of Theorem 11. First of all, Φ cannot be almost injective if $N < M+1$ since $\operatorname{rank}\Phi_S + \operatorname{rank}\Phi_{S^c} \leqslant |S| + |S^c| = N$. Also, in the case where $N = M+1$, we note that Φ is almost injective precisely when Φ is *full spark*, that is, every size-M subcollection is a spanning set (note this implies that all of the φ_n's are nonzero). In fact, every full spark Φ with $N \geqslant M+1$ yields almost injective intensity measurements, which in turn implies that a generic Φ yields almost injectivity when $N \geqslant M+1$ [4]. This is in direct analogy with injectivity in the real case; here, injectivity requires $N \geqslant 2M-1$, injectivity with $N = 2M-1$ is equivalent to being full spark, and being full spark suffices for injectivity whenever $N \geqslant 2M-1$ [4].

Another thing to check is that the condition for injectivity implies the condition for almost injectivity (it does).

Having established that full spark ensembles of size $N \geqslant M+1$ yield almost injective intensity measurements, we note that checking whether a matrix is full spark is NP-hard in general [33]. Granted, there are a few explicit constructions of full spark ensembles which can be used [2,36], but it would be nice to have a condition which is not computationally difficult to test in general. We provide one such condition in the following theorem, but first, we briefly review the requisite frame theory.

A *frame* is an ensemble $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ together with *frame bounds* $0 < A \leqslant B < \infty$ with the property that for every $x \in \mathbb{R}^M$,

$$A\|x\|^2 \leqslant \sum_{n=1}^N \big|\langle x, \varphi_n\rangle\big|^2 \leqslant B\|x\|^2.$$

When $A = B$, the frame is said to be *tight*, and such frames come with a painless reconstruction formula:

$$x = \frac{1}{A}\sum_{n=1}^N \langle x, \varphi_n\rangle \varphi_n.$$

To be clear, the theory of frames originated in the context of infinite-dimensional Hilbert spaces [22,24], and frames have since been studied in finite-dimensional settings, primarily because this is the setting in which they are applied computationally. Of particular interest are so-called *unit norm tight frames (UNTFs)*, which are tight frames whose frame elements have unit norm: $\|\varphi_n\| = 1$ for every $n = 1, \ldots, N$. Such frames are useful in applications; for example, if one encodes a signal x using frame coefficients $\langle x, \varphi_n\rangle$ and transmits these coefficients across a channel, then UNTFs are optimally robust to noise [29] and one erasure [17]. Intuitively, this optimality comes from the fact that frame elements of a UNTF are particularly well-distributed in the unit sphere [7]. Another pleasant feature of UNTFs is that it is straightforward to test whether a given frame is a UNTF: Letting $\Phi = [\varphi_1 \cdots \varphi_N]$ denote an $M \times N$ matrix whose columns are the frame elements, then Φ is a UNTF precisely when each of the following occurs simultaneously:

(i) the rows have equal norm
(ii) the rows are orthogonal
(iii) the columns have unit norm

(This is a direct consequence of the tight frame's reconstruction formula and the fact that a UNTF has unit-norm frame elements; furthermore, since the columns have unit norm, it is not difficult to see that the rows will necessarily have norm $\sqrt{N/M}$.) In addition to being able to test that an ensemble is a UNTF, various UNTFs can be constructed using

spectral tetris [16] (though such frames necessarily have $N \geqslant 2M$), and *every* UNTF can be constructed using the recent theory of *eigensteps* [11,26]. Now that UNTFs have been properly introduced, we relate them to almost injectivity for phase retrieval:

Theorem 12. *If M and N are relatively prime, then every unit norm tight frame $\Phi = \{\varphi_n\}_{n=1}^N \subseteq \mathbb{R}^M$ yields almost injective intensity measurements.*

Proof. Pick a nonempty proper subset $S \subseteq \{1,\dots,N\}$. By Theorem 11, it suffices to show that $\operatorname{rank}\Phi_S + \operatorname{rank}\Phi_{S^c} > M$, or equivalently, $\operatorname{rank}\Phi_S\Phi_S^* + \operatorname{rank}\Phi_{S^c}\Phi_{S^c}^* > M$. Note that since Φ is a unit norm tight frame, we also have

$$\Phi_S\Phi_S^* + \Phi_{S^c}\Phi_{S^c}^* = \Phi\Phi^* = \frac{N}{M}I,$$

and so $\Phi_S\Phi_S^*$ and $\Phi_{S^c}\Phi_{S^c}^*$ are simultaneously diagonalizable, i.e., there exists a unitary matrix U and diagonal matrices D_1 and D_2 such that

$$UD_1U^* + UD_2U^* = \Phi_S\Phi_S^* + \Phi_{S^c}\Phi_{S^c}^* = \frac{N}{M}I.$$

Conjugating by U^*, this then implies that $D_1 + D_2 = \frac{N}{M}I$. Let $L_1 \subseteq \{1,\dots,M\}$ denote the diagonal locations of the nonzero entries in D_1, and $L_2 \subseteq \{1,\dots,M\}$ similarly for D_2. To complete the proof, we need to show that $|L_1|+|L_2| > M$ (since $|L_1|+|L_2| = \operatorname{rank}\Phi_S\Phi_S^* + \operatorname{rank}\Phi_{S^c}\Phi_{S^c}^*$). Note that $L_1 \cup L_2 \neq \{1,\dots,M\}$ would imply that $D_1 + D_2$ has at least one zero in its diagonal, contradicting the fact that $D_1 + D_2$ is a nonzero multiple of the identity; as such, $L_1 \cup L_2 = \{1,\dots,M\}$ and $|L_1| + |L_2| \geqslant M$. We claim that this inequality is strict due to the assumption that M and N are relatively prime. To see this, it suffices to show that $L_1 \cap L_2$ is nonempty. Suppose to the contrary that L_1 and L_2 are disjoint. Then since $D_1 + D_2 = \frac{N}{M}I$, every nonzero entry in D_1 must be N/M. Since S is a nonempty proper subset of $\{1,\dots,N\}$, this means that there exists $K \in (0, M)$ such that D_1 has K entries which are N/M and $M - K$ which are 0. Thus,

$$|S| = \operatorname{Tr}\big[\Phi_S^*\Phi_S\big] = \operatorname{Tr}\big[\Phi_S\Phi_S^*\big] = \operatorname{Tr}\big[UD_1U^*\big] = \operatorname{Tr}[D_1] = K(N/M),$$

implying that $N/M = |S|/K$ with $K \neq M$ and $|S| \neq N$. Since this contradicts the assumption that N/M is in lowest form, we have the desired result. □

In general, whether a UNTF Φ yields almost injective intensity measurements is determined by whether it is *orthogonally partitionable*: Φ is orthogonally partitionable if there exists a partition $S \sqcup S^c = \{1,\dots,N\}$ such that $\operatorname{span}(\Phi_S)$ is orthogonal to $\operatorname{span}(\Phi_{S^c})$. Specifically, a UNTF yields almost injective intensity measurements precisely when it is not orthogonally partitionable. Historically, this property of UNTFs has been pivotal to the understanding of singularities in the algebraic variety of UNTFs [25], and it has also played a key role in solutions to the Paulsen problem [8,15]. However, it is not

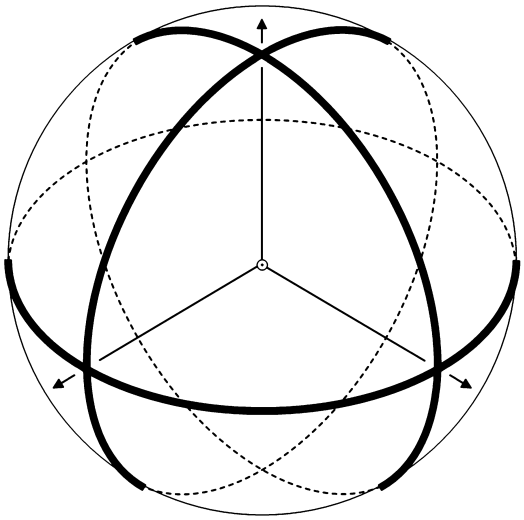


Fig. 1. The simplex in $\mathbb{R}^3$. Pointing out of the page is the vector $\frac{1}{\sqrt{3}}(1,1,1)$, while the other vectors are the three permutations of $\frac{1}{\sqrt{3}}(1,-1,-1)$. Together, these four vectors form a unit norm tight frame, and since $M=3$ and $N=4$ are relatively prime, these yield almost injective intensity measurements in accordance with Theorem 12. For this ensemble, the points x such that $\mathcal{A}^{-1}(\mathcal{A}(x)) \neq \{\pm x\}$ are contained in the three coordinate planes. Above, we depict the intersection between these planes and the unit sphere. According to Theorem 14, performing phase retrieval with simplices such as this is NP-hard.

clear in general how to efficiently test for this property; this is why Theorem 12 is so powerful.

4. The computational complexity of phase retrieval

The previous section characterized the real ensembles which yield almost injective intensity measurements. The benefit of seeking almost injectivity instead of injectivity is that we can get away with much smaller ensembles. For example, a full spark ensemble in $\mathbb{R}^M$ of size $M+1$ suffices for almost injectivity (see Fig. 1), while $2M-1$ measurements are required for injectivity. In this section, we demonstrate that this savings in the number of measurements can come at a substantial price in computational requirements for phase retrieval. In particular, we consider the following problem:

Problem 13. Let $\mathcal{F} = \{\Phi_M\}_{M=2}^{\infty}$ be a family of ensembles $\Phi_M = \{\varphi_{M;n}\}_{n=1}^{N(M)} \subseteq \mathbb{R}^M$, where $N(M) = \mathsf{poly}(M)$. Then ConsistentIntensities$[\mathcal{F}]$ is the following problem: Given $M \geqslant 2$ and a rational sequence $\{b_n\}_{n=1}^{N(M)}$, does there exist $x \in \mathbb{R}^M$ such that $|\langle x, \varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, N(M)$?

In this section, we will evaluate the computational complexity of ConsistentIntensities$[\mathcal{F}]$ for a large class of families of small ensembles $\mathcal{F}$, but first, we briefly review the main concepts involved. Complexity theory is chiefly concerned with *complexity classes*, which are sets of problems that share certain computational requirements, such as time or space. For example, the complexity class P is the set of problems which can be solved in an amount of time that is bounded by some polynomial of the bit-length

of the input. As another example, NP contains all problems for which an affirmative answer comes with a certificate that can be verified in polynomial time; note that $\mathsf{P} \subseteq \mathsf{NP}$ since for every problem $A \in \mathsf{P}$, one may ignore the certificate and find the affirmative answer in polynomial time. One key tool that is used to evaluate the complexity of a problem is called *polynomial-time reduction*. This is a polynomial-time algorithm that solves a problem A by exploiting an oracle which solves another problem B, indicating that solving A is no harder than solving B (up to polynomial factors in time); if such a reduction exists, we write $A \leqslant B$. For example, any efficient phase retrieval procedure for $\mathcal{F}$ can be used as a subroutine to solve ConsistentIntensities$[\mathcal{F}]$, indicating that phase retrieval for $\mathcal{F}$ is at least as hard as ConsistentIntensities$[\mathcal{F}]$. A problem B is called NP-*hard* if $B \geqslant A$ for every problem $A \in \mathsf{NP}$. Note that since $\leqslant$ is transitive, it suffices to show that $B \geqslant C$ for some NP-hard problem C. Finally, a problem B is called NP-*complete* if $B \in \mathsf{NP}$ is NP-hard; intuitively, NP-complete problems are the hardest of problems in NP. It is an open problem whether $\mathsf{P} = \mathsf{NP}$, but inequality is widely believed [20]; note that under this assumption, NP-hard problems have no computationally efficient solution. This provides a proper context for the main result of this section:

Theorem 14. *Let $\mathcal{F} = \{\Phi_M\}_{M=2}^{\infty}$ be a family of full spark ensembles $\Phi_M = \{\varphi_{M;n}\}_{n=1}^{M+1} \subseteq \mathbb{R}^M$ with rational entries that can be computed in polynomial time. Then* ConsistentIntensities$[\mathcal{F}]$ *is* NP-*complete.*

Note that since the ensembles Φ_M are full spark, the existence of a solution to the phase retrieval problem $|\langle x, \varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$ implies uniqueness by Theorem 11. Before proving this theorem, we first relate it to a previous hardness result from [37]. Specifically, this result can be restated using the terminology in this paper as follows: There exists a family $\mathcal{F} = \{\Phi_M\}_{M=2}^{\infty}$ of ensembles $\Phi_M = \{\varphi_{M;n}\}_{n=1}^{2M} \subseteq \mathbb{C}^M$, each of which yielding almost injective intensity measurements, such that ConsistentIntensities$[\mathcal{F}]$ is NP-complete. Interestingly, these are the smallest possible almost injective ensembles in the complex case, and we suspect that the result can be strengthened to the obvious analogy of Theorem 14:

Conjecture 15. *Let $\mathcal{F} = \{\Phi_M\}_{M=2}^{\infty}$ be a family of ensembles $\Phi_M = \{\varphi_{M;n}\}_{n=1}^{2M} \subseteq \mathbb{C}^M$ which yield almost injective intensity measurements and have complex rational entries that can be computed in polynomial time. Then* ConsistentIntensities$[\mathcal{F}]$ *is* NP-*complete.*

To prove Theorem 14, we devise a polynomial-time reduction from the following problem which is well-known to be NP-complete [32]:

Problem 16 *(*SubsetSum*).* Given a finite collection of integers A and an integer z, does there exist a subset $S \subseteq A$ such that $\sum_{a \in S} a = z$?

Proof of Theorem 14. We first show that ConsistentIntensities$[\mathcal{F}]$ is in NP. Note that if there exists an $x \in \mathbb{R}^M$ such that $|\langle x, \varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$, then x will have all rational entries. Indeed, $v := \Phi_M^* x$ has all rational entries, being a signed version of $\{b_n\}_{n=1}^{M+1}$, and so $x = (\Phi_M \Phi_M^*)^{-1} \Phi_M v$ is also rational. Thus, we can view x as a certificate of finite bit-length, and for each $n = 1, \ldots, M+1$, we know that $|\langle x, \varphi_{M;n}\rangle| = b_n$ can be verified in time which is polynomial in this bit-length, as desired.

Now we show that ConsistentIntensities$[\mathcal{F}]$ is NP-hard by reduction from SubsetSum. To this end, take a finite collection of integers A and an integer z. Set $M := |A|$ and label the members of A as $\{a_m\}_{m=1}^M$. Let Ψ denote the $M \times M$ matrix whose columns are the first M members of Φ_M. Since Φ_M is full spark, Ψ is invertible and $\Psi^{-1}\Phi_M$ has the form $[I\ w]$, where w has all nonzero entries; indeed, if the mth entry of w were zero, then $\Phi_M \setminus \{\varphi_{M;m}\}$ would not span, violating full spark. Now define

$$b_n := \begin{cases} \left|\dfrac{a_n}{w_n}\right| & \text{if } n = 1, \ldots, M \\ \left|2z - \displaystyle\sum_{m=1}^M a_m\right| & \text{if } n = M+1. \end{cases} \tag{16}$$

We claim that an oracle for ConsistentIntensities$[\mathcal{F}]$ would return "yes" from the inputs M and $\{b_n\}_{n=1}^{M+1}$ defined above if and only if there exists a subset $S \subseteq A$ such that $\sum_{a \in S} a = z$, which would complete the reduction.

To prove our claim, we start with ($\Rightarrow$): Suppose there exists $x \in \mathbb{R}^M$ such that $|\langle x, \varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$. Then $y := \Psi^* x$ satisfies $|\langle y, \Psi^{-1}\varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$. Since $\Psi^{-1}\Phi_M = [I\ w]$, then by (16), the entries of y satisfy

$$|y_m| = \left|\frac{a_m}{w_m}\right| \quad \forall m = 1, \ldots, M, \qquad \left|\sum_{m=1}^M y_m w_m\right| = \left|2z - \sum_{m=1}^M a_m\right|.$$

By the first equation above, there exists a sequence $\{\varepsilon_m\}_{m=1}^M$ of ± 1's such that $y_m = \varepsilon_m a_m / w_m$ for every $m = 1, \ldots, M$, and so the second equation above gives

$$\begin{aligned} \left|2z - \sum_{m=1}^M a_m\right| &= \left|\sum_{m=1}^M y_m w_m\right| = \left|\sum_{m=1}^M \varepsilon_m a_m\right| \\ &= \left|\sum_{\substack{m=1 \\ \varepsilon_m = 1}}^M a_m - \sum_{\substack{m=1 \\ \varepsilon_m = -1}}^M a_m\right| = \left|2\sum_{\substack{m=1 \\ \varepsilon_m = 1}}^M a_m - \sum_{m=1}^M a_m\right|. \end{aligned}$$

Removing the absolute values, this means the left-hand side above is equal to the right-hand side, up to a sign factor. At this point, isolating z reveals that $z = \sum_{m \in S} a_m$, where S is either $\{m\colon\ \varepsilon_m = 1\}$ or $\{m\colon\ \varepsilon_m = -1\}$, depending on the sign factor.

For ($\Leftarrow$), suppose there is a subset $S \subseteq \{1, \ldots, M\}$ such that $z = \sum_{m \in S} a_m$. Define $\varepsilon_m := 1$ when $m \in S$ and $\varepsilon_m := -1$ when $m \notin S$. Then

$$\left|\sum_{m=1}^{M}\varepsilon_m a_m\right| = \left|\sum_{\substack{m=1\\ \varepsilon_m=1}}^{M} a_m - \sum_{\substack{m=1\\ \varepsilon_m=-1}}^{M} a_m\right| = \left|2\sum_{\substack{m=1\\ \varepsilon_m=1}}^{M} a_m - \sum_{m=1}^{M} a_m\right| = \left|2z - \sum_{m=1}^{M} a_m\right|.$$

By the analysis from the ($\Rightarrow$) direction, taking $y_m := \varepsilon_m a_m / w_m$ for each $m = 1, \ldots, M$ then ensures that $|\langle y, \Psi^{-1}\varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$, which in turn ensures that $x := (\Psi^*)^{-1}y$ satisfies $|\langle x, \varphi_{M;n}\rangle| = b_n$ for every $n = 1, \ldots, M+1$. $\square$

Based on Theorem 14, there is no polynomial-time algorithm to perform phase retrieval for minimal almost injective ensembles, assuming $\mathsf{P} \neq \mathsf{NP}$. On the other hand, there exist ensembles of size $2M-1$ for which phase retrieval is particularly efficient. For example, letting $\delta_{M;m} \in \mathbb{R}^M$ denote the mth identity basis element, consider the ensemble $\Phi_M := \{\delta_{M;m}\}_{m=1}^{M} \cup \{\delta_{M;1} + \delta_{M;m}\}_{m=2}^{M}$; then one can reconstruct (up to global phase) any x whose first entry is nonzero by first taking $\hat{x}[1] := |\langle x, \delta_{M;1}\rangle|$, and then taking

$$\hat{x}[m] := \frac{1}{2\hat{x}[1]}\Big(\big|\langle x, \delta_{M;1} + \delta_{M;m}\rangle\big|^2 - \big|\langle x, \delta_{M;1}\rangle\big|^2 - \big|\langle x, \delta_{M;m}\rangle\big|^2\Big) \quad \forall m = 2, \ldots, M.$$

Intuitively, we expect a redundancy threshold that determines whether phase retrieval can be efficient, and this suggests the following open problem: What is the smallest C for which there exists a family of ensembles of size $N = CM + o(M)$ such that phase retrieval can be performed in polynomial time?

Acknowledgements

The authors thank the Norbert Wiener Center for Harmonic Analysis and Applications at the University of Maryland, College Park for hosting a workshop on phase retrieval that helped solidify the main ideas in the almost injectivity portion of this paper. The authors also thank the anonymous referees for pointing out arguments which significantly shortened our proofs of Lemma 4 and Theorem 10. This work was supported by NSF DMS 1042701 and 1321779. The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

References

[1] B. Alexeev, A.S. Bandeira, M. Fickus, D.G. Mixon, Phase retrieval with polarization, available online: arXiv:1210.7752.
[2] B. Alexeev, J. Cahill, D.G. Mixon, Full spark frames, J. Fourier Anal. Appl. 18 (2012) 1167–1194.
[3] R. Balan, B.G. Bodmann, P.G. Casazza, D. Edidin, Painless reconstruction from magnitudes of frame coefficients, J. Fourier Anal. Appl. 15 (2009) 488–501.
[4] R. Balan, P. Casazza, D. Edidin, On signal reconstruction without phase, Appl. Comput. Harmon. Anal. 20 (2006) 345–356.
[5] A.S. Bandeira, J. Cahill, D.G. Mixon, A.A. Nelson, Saving phase: Injectivity and stability for phase retrieval, available online: arXiv:1302.4618.

[6] A.S. Bandeira, Y. Chen, D.G. Mixon, Phase retrieval from power spectra of masked signals, available online: arXiv:1303.4458.
[7] J.J. Benedetto, M. Fickus, Finite normalized tight frames, Adv. Comput. Math. 18 (2003) 357–385.
[8] B.G. Bodmann, P.G. Casazza, The road to equal-norm Parseval frames, J. Funct. Anal. 258 (2010) 397–420.
[9] B.G. Bodmann, N. Hammen, Stable phase retrieval with low-redundancy frames, available online: arXiv:1302.5487.
[10] O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D.K. Satapathy, J.F. van der Veen, Diffractive imaging for periodic samples: retrieving one-dimensional concentration profiles across microfluidic channels, Acta Crystallogr. Sect. A 63 (2007) 306–314.
[11] J. Cahill, M. Fickus, D.G. Mixon, M.J. Poteet, N. Strawn, Constructing finite frames of a given spectrum and set of lengths, Appl. Comput. Harmon. Anal. 35 (2013) 52–73.
[12] E.J. Candès, Y.C. Eldar, T. Strohmer, V. Voroninski, Phase retrieval via matrix completion, SIAM J. Imaging Sci. 6 (2013) 199–225.
[13] E.J. Candès, X. Li, Solving quadratic equations via PhaseLift when there are about as many equations as unknowns, available online: arXiv:1208.6247.
[14] E.J. Candès, T. Strohmer, V. Voroninski, PhaseLift: Exact and stable signal recovery from magnitude measurements via convex programming, Comm. Pure Appl. Math. 66 (2013) 1241–1274.
[15] P.G. Casazza, M. Fickus, D.G. Mixon, Auto-tuning unit norm frames, Appl. Comput. Harmon. Anal. 32 (2012) 1–15.
[16] P.G. Casazza, M. Fickus, D.G. Mixon, Y. Wang, Z. Zhou, Constructing tight fusion frames, Appl. Comput. Harmon. Anal. 30 (2011) 175–187.
[17] P.G. Casazza, J. Kovačević, Equal-norm tight frames with erasures, Adv. Comput. Math. 18 (2003) 387–430.
[18] A. Chai, M. Moscoso, G. Papanicolaou, Array imaging using intensity-only measurements, Inverse Problems 27 (2011) 015005.
[19] A. Conca, D. Edidin, M. Hering, C. Vinzant, An algebraic characterization of injectivity in phase retrieval, available online: arXiv:1312.0158.
[20] S. Cook, The P versus NP problem, available online: http://www.claymath.org/millennium/PvsNP/pvsnp.pdf.
[21] J.C. Dainty, J.R. Fienup, Phase retrieval and image reconstruction for astronomy, in: H. Stark (Ed.), Image Recovery: Theory and Application, Academic Press, New York, 1987.
[22] I. Daubechies, A. Grossmann, Y. Meyer, Painless nonorthogonal expansions, J. Math. Phys. 27 (1986) 1271–1283.
[23] L. Demanet, P. Hand, Stable optimizationless recovery from phaseless linear measurements, available online: arXiv:1208.1803.
[24] R.J. Duffin, A.C. Schaeffer, A class of nonharmonic Fourier series, Trans. Amer. Math. Soc. 72 (1952) 341–366.
[25] K. Dykema, N. Strawn, Manifold structure of spaces of spherical tight frames, Int. J. Pure Appl. Math. 28 (2006) 217–256.
[26] M. Fickus, D.G. Mixon, M.J. Poteet, N. Strawn, Constructing all self-adjoint matrices with prescribed spectrum and diagonal, available online: arXiv:1107.2173.
[27] J. Finkelstein, Pure-state informationally complete and "really" complete measurements, Phys. Rev. A 70 (2004) 052107.
[28] S.T. Flammia, A. Silberfarb, C.M. Caves, Minimal informationally complete measurements for pure states, Found. Phys. 35 (2005) 1985–2006.
[29] V.K. Goyal, M. Vetterli, N.T. Thao, Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis, and algorithms, IEEE Trans. Inform. Theory 44 (1998) 1–31.
[30] R.W. Harrison, Phase problem in crystallography, J. Opt. Soc. Amer. A 10 (1993) 1046–1055.
[31] T. Heinosaari, L. Mazzarella, M.M. Wolf, Quantum tomography under prior information, Comm. Math. Phys. 318 (2013) 355–374.
[32] R.M. Karp, Reducibility among combinatorial problems, in: R.E. Miller, J.W. Thatcher (Eds.), Complexity of Computer Computations, 1972, pp. 85–103.
[33] L. Khachiyan, On the complexity of approximating extremal determinants in matrices, J. Complexity 11 (1995) 138–153.
[34] J. Miao, T. Ishikawa, Q. Shen, T. Earnest, Extending X-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes, Annu. Rev. Phys. Chem. 59 (2008) 387–410.
[35] R.P. Millane, Phase retrieval in crystallography and optics, J. Opt. Soc. Amer. A 7 (1990) 394–411.

[36] M. Püschel, J. Kovačević, Real, tight frames with maximal robustness to erasures, in: Proc. Data Compr. Conf., 2005, pp. 63–72.
[37] H. Sahinoglou, S.D. Cabrera, On phase retrieval of finite-length sequences using the initial time sample, IEEE Trans. Circuits Syst. 38 (1991) 954–958.
[38] V. Voroninski, A comparison between the PhaseLift and PhaseCut algorithms, available online: http://math.berkeley.edu/~vladv/PhaseCutProofs.pdf.
[39] V. Voroninski, Phase retrieval from quadratic unitary measurements and implications for Wright's conjecture, available online: http://math.berkeley.edu/~vladv/UnitaryCase.pdf.
[40] I. Waldspurger, A. d'Aspremont, S. Mallat, Phase recovery, MaxCut and complex semidefinite programming, available online: arXiv:1206.0102.
[41] A. Walther, The question of phase retrieval in optics, Opt. Acta 10 (1963) 41–49.

Advances in Adaptive Data Analysis
Vol. 5, No. 1 (2013) 1350003 (25 pages)

DOI: 10.1142/S1793536913500039

ADAPTIVE SUB-LINEAR TIME FOURIER ALGORITHMS

DAVID LAWLOR*,†,‡,††, YANG WANG*,§,‖ and ANDREW CHRISTLIEB*,¶,**

*Department of Mathematics, Michigan State University
619 Red Cedar Road, East Lansing, MI 48824, USA

†Mathematics Department, Duke University
Box 90320, Durham, NC 27708-0320, USA
‡djl@msu.edu
‡djl@math.duke.edu
§ywang@math.msu.edu
¶christlieb@math.msu.edu

Received 17 September 2012
Revised 10 January 2013
Accepted 10 January 2013
Published 9 April 2013

We present a new deterministic algorithm for the sparse Fourier transform problem, in which we seek to identify $k \ll N$ significant Fourier coefficients from a signal of bandwidth N. Previous deterministic algorithms exhibit quadratic runtime scaling, while our algorithm scales linearly with k in the average case. Underlying our algorithm are a few simple observations relating the Fourier coefficients of time-shifted samples to unshifted samples of the input function. This allows us to detect when aliasing between two or more frequencies has occurred, as well as to determine the value of unaliased frequencies. We show that empirically our algorithm is orders of magnitude faster than competing algorithms.

Keywords: Fourier algorithm.

1. Introduction

The Fast Fourier Transform (FFT) is arguably the most ubiquitous numerical algorithm in scientific computing. In addition to being named one of the "Top Ten Algorithms" of the past century [Dongarra and Sullivan (2000)], the FFT is a critical tool in myriad applications, ranging from signal processing to computational PDE and machine learning. At the time of its introduction, it represented a major leap forward in the size of problems that could be solved on available hardware, as it reduces the runtime complexity of computing the Discrete Fourier Transform (DFT) of a length-N array from $\mathrm{O}(N^2)$ to $\mathrm{O}(N \log N)$.

‖Y. Wang was supported in part by NSF-DMS awards 0813750 and 1043034.
**A. Christlieb was supported in part by NSF-DMS-FRG award 0652833.
††Corresponding author.

Any algorithm which computes all N Fourier coefficients has a runtime complexity of $\Omega(N)$, since it takes that much time merely to report the output. However, in many applications it is known that the DFT of the signal of interest is highly sparse — that is, only a small number of coefficients are non-zero. In this case, it is possible to break the $\Omega(N)$ barrier by asking only for the largest k terms in the signal's DFT. When $k \ll N$ existing algorithms can significantly outperform even highly optimized FFT implementations [Iwen *et al.* (2007); Iwen (2010); Hassanieh *et al.* (2012b)].

1.1. *Related work*

The first works to implicitly address the sparse approximate DFT problem appeared in the theoretical computer science literature in the early 1990s. In Linial *et al.* [1993], a variant of the Fourier transform for Boolean functions was shown to have applications for learnability. A polynomial-time algorithm to find large coefficients in this basis was given in Kushilevitz and Mansour [1993], while the interpolation of sparse polynomials over finite fields was considered in Mansour [1995]. It was later realized [Gilbert *et al.* (2005)] that this last algorithm could be considered as an approximate DFT for the special case when N is a power of two.

In the past 10 or so years, a number of algorithms have appeared which directly address the problem of computing sparse approximate Fourier transforms. When comparing the results in the literature, care must be taken to identify the class of signals over which a specific algorithm is to perform, as well as to identify the error bounds of a given method. Different algorithms have been devised in different research communities, and so have varying assumptions on the underlying signals as well as different levels of acceptable error.

The first result with sub-linear runtime and sampling requirements appeared in Gilbert *et al.* [2002]. They give a $\text{poly}(k, \log N, \log(1/\delta), 1/\varepsilon)$ time algorithm for finding, with probability $1-\delta$, an approximation $\hat{y}$ of the DFT of the input $\hat{x}$ that is nearly optimal, in the sense that $\|\hat{x}-\hat{y}\|_2^2 \leq (1+\varepsilon)\|\hat{x}-\hat{x}_{\text{opt}}\|_2^2$, where $\hat{x}_{\text{opt}}$ is the best k-term approximation to $\hat{x}$. Here, the exponent of k in the runtime is two, so the algorithm is *quadratic* in the sparsity. Moreover, the algorithm is non-adaptive in the sense that the samples used are independent of the input x. This algorithm was modified in Gilbert *et al.* [2005] to bring the dependence on k down to linear.[a] This was accomplished mainly by replacing uniform random variables (used to sample the input) by random arithmetic progressions, which allowed the use of non-equispaced FFTs to sample from intermediate representations and to estimate the coefficients in near-linear time. The increased overhead of this procedure, however, limited the range of k for which the algorithm outperformed a standard FFT implementation [Iwen *et al.* (2007)].

[a]See Gilbert *et al.* [2008] for a "user-friendly" description of the improved algorithm.

Around the same time, a similar algorithm was developed in the context of list decoding for proving hard-core predicates for one-way functions [Akavia *et al.* (2003)]. This can be considered an extension of Kushilevitz and Mansour [1993], and like Gilbert *et al.* [2002, 2005] is a randomized algorithm. Since the goal in this work was to give a polynomial-time algorithm for list decoding, no effort was made to optimize the dependence on k; it stands at $k^{11/2}$, considerably higher than Gilbert *et al.* [2002, 2005]. The randomness in this algorithm is used only to construct a sample set on which norms are estimated, and in Akavia [2010] this set is replaced with a deterministic construction. This construction is based on the notion of ε-approximating the uniform distribution over arithmetic progressions, and relies on existing constructions of ε-biased sets of small size [Katz (1989); Ajtai *et al.* (1990)]. Depending on the size of the ε-biased sets used, the sampling and runtime complexities are $\mathrm{O}(k^4 \log^c N)$ and $\mathrm{O}(k^6 \log^c N)$, respectively, for some $c > 4$.[b]

In the series of works [Iwen (2008, 2010, 2012)], a different deterministic algorithm for sparse Fourier approximation was given that relies on the combinatorial properties of *aliasing*, or collisions among frequencies in sub-sampled DFTs. By taking enough short DFTs of co-prime lengths, and employing the Chinese Remainder Theorem (CRT) to reconstruct energetic frequencies from their residues modulo these sample lengths, the author is able to prove sampling and runtime bounds of $\mathrm{O}(k^2 \log^4 N)$. The error bound is of the form $\|\hat{x} - \hat{y}\|_2 \leq \|\hat{x} - \hat{x}_{\mathrm{opt}}\|_2 + k^{-1/2}\|\hat{x} - \hat{x}_{\mathrm{opt}}\|_1$; it has been shown that the stronger "ℓ_2-ℓ_2" guarantee of Gilbert *et al.* [2005] cannot hold for a sub-linear, deterministic algorithm [Cohen *et al.* (2009)]. Moreover, the range of k for which this algorithm is faster than the FFT is smaller in practice than that of Gilbert *et al.* [2005].

Most recently, the authors of Hassanieh *et al.* [2012b] presented a randomized algorithm that extends by an order of magnitude the range of sparsity for which it is faster than the FFT. This is accomplished by removing the iterative aspect from Gilbert *et al.* [2005] by using more efficient filters, which are nearly flat within the passband and which decay exponentially outside. In contrast, the box-car filters used in Gilbert *et al.* [2005] have a frequency response which oscillates and decays like $|\omega|^{-1}$. In addition, the identification of significant frequencies is done by direct estimation after hashing into a large number of bins rather than the binary search technique of Gilbert *et al.* [2005]. These changes give a runtime bound of $\mathrm{O}(\log N\sqrt{Nk \log N})$ and a somewhat stronger error bound $\|\hat{x} - \hat{y}\|_\infty^2 \leq \varepsilon k^{-1}\|\hat{x} - \hat{x}_{\mathrm{opt}}\|_2^2 + \delta\|\hat{x}\|_1^2$ with probability $1 - 1/N$, where $\varepsilon > 0$ and $\delta = N^{-\mathrm{O}(1)}$ is a precision parameter.

[b]Specifically, the runtime is $\mathrm{O}(k^2 \cdot \log N \cdot |S|)$, where S is the set of samples read by the algorithm. This set takes the form $S = \bigcup_{\ell=1}^{\lfloor \log N \rfloor} A - B_\ell$, where A has ε-discrepancy on rank 2 Bohr sets, B_ℓ ε-approximates the uniform distribution on $[0, 2^\ell - 1] \cap \mathbb{Z}$, and $A - B_\ell$ is the difference set. Using constructions from Katz [1989] one has $|A| = \mathrm{O}(\varepsilon^{-1} \log^4 N), |B_\ell| = \mathrm{O}(\varepsilon^{-3} \log^4 N)$; setting $\varepsilon = \Theta(k^{-1})$ and noting that $|\bigcup A - B_\ell| = \mathrm{O}(\sum |A - B_\ell|)$ and $|A - B_\ell| = \mathrm{O}(|A||B_\ell|)$ [see, e.g. Tao and Vu (2006)] one obtains the stated sampling and runtime complexities.

These existing algorithms generally take one of two approaches to the sparse Fourier transform problem. In Gilbert *et al.* [2002], Akavia *et al.* [2003], Gilbert *et al.* [2005] and Hassanieh *et al.* [2012b], the spectrum of the input is randomly permuted and then run through a low-pass filter to isolate and identify frequencies which carry a large fraction of the signal's energy. This leads to randomized algorithms that fail on a non-negligible set of possible inputs. On the other hand, Iwen [2010] takes advantage of the combinatorial properties of *aliasing* in order to identify the significant frequencies. This leads to a deterministic algorithm with higher runtime and sampling requirements than the randomized algorithms mentioned. Both of these randomized and deterministic approaches have drawbacks. Randomized algorithms are not suitable for failure-intolerant applications, while the process used to reconstruct significant frequencies in Iwen [2010] relies on the CRT, which is highly unstable to errors in the residues. While there do exist algorithms for "noisy Chinese Remaindering" [Goldreich *et al.* (2000); Boneh (2002); Shparlinski and Steinfeld (2004)] these have thus far not found application to the sparse DFT problem, and we leave this as future work.

As this paper was being prepared, the authors became aware of an independent work using very similar methods for frequency estimation in the noiseless case [Hassanieh *et al.* (2012a)]. Both methods consider the phase difference between Fourier samples to extract frequency information, but are based on different techniques for binning significant frequencies. The authors of Hassanieh *et al.* [2012a] use random dilations and efficient filters of Hassanieh *et al.* [2012b], whereas we use different sample lengths in the spirit of Iwen [2010]. We believe both contributions are of interest, and reinforce the notion that exploiting phase information is critical for developing fast, robust algorithms for the sparse Fourier transform problem.

1.2. *Relationship to compressed sensing*

The term "compressed sensing" refers to a new paradigm in signal processing which seeks to recover a compressible signal from a number of linear measurements roughly proportional to its information content, rather than its nominal dimension. While this paper does not make explicit use of the results or algorithms of compressed sensing, there are parallels in the approaches used. The purpose of this section is to clarify the relationship between the two.

All algorithms for the sparse Fourier transform problem take a small number of samples of the input x, either at random or in a deterministic fashion. These samples are then processed in a highly non-linear, algorithm-dependent manner to produce a k-term Fourier representation of x – that is, a list $\{(\widetilde{\omega}_\ell, \widetilde{a}_\ell)\}_{\ell=1}^{k}$ of significant frequency/coefficient pairs. In other words, these algorithms approximately solve the severely underdetermined system $\boldsymbol{R}\boldsymbol{F}^*\widehat{x} = \boldsymbol{R}x$, where $\boldsymbol{R}$ is the restriction to the samples used by the algorithm, $\boldsymbol{F}^*$ is the adjoint of the $N \times N$ discrete Fourier

matrix with entries

$$F_{jk} = \frac{1}{\sqrt{N}} \mathrm{e}^{-2\pi \mathrm{i} jk/N}, \quad 0 \leq j, \;\; k < N \tag{1}$$

and $\widehat{x}$ is the DFT of x.

In Candès *et al.* [2006], Candès, Romberg, and Tao considered the dual problem — that of recovering a given signal from highly incomplete Fourier measurements. Specifically, suppose that a signal x of length N is the superposition of k spikes at times $t = \tau_j$:

$$x[t] = \sum_{j=1}^{k} x[\tau_j] \delta(t - \tau_j). \tag{2}$$

The authors show that, with high probability, x can be recovered exactly from a randomly chosen set Ω of m frequencies from the DFT of x, provided

$$m \geq Ck \log N \tag{3}$$

for some constant C whose value depends on the desired probability of success. This can be viewed as the severely underdetermined linear system dual to the system described above: $\boldsymbol{RF}x = \boldsymbol{R}\widehat{x}$. The recovery algorithm in this case is the ℓ_1 minimization

$$g^* = \operatorname{argmin} \|g\|_1 \quad \text{subject to } \widehat{g}(\omega) = \widehat{f}(\omega) \;\; \text{for all } \omega \in \Omega. \tag{4}$$

The idea of using ℓ_1 minimization to recover sparse vectors has been studied extensively in a number of research communities, including seismic imaging [Santosa and Symes (1986)], image processing [Rudin *et al.* (1992)], and signal processing [Chen *et al.* (1998)], where it is commonly referred to as basis pursuit. The theoretical foundations of ℓ_1 approximation are treated in depth in the monograph [Pinkus (1989)].

Other sampling schemes and recovery algorithms have been studied for the compressed sensing problem. For example, in Rauhut [2007], ℓ_1 minimization is used with points sampled randomly from a *continuous* distribution, while in Xu [2011] a deterministic sampling scheme is analyzed with reconstruction through Orthogonal Matching Pursuit (OMP). Other works which analyze the performance of OMP in the compressed sensing setting include Tropp and Gilbert [2007] and Kunis and Rauhut [2008].

Sparse Fourier approximation and compressed sensing are therefore broadly similar in both their goals (sparse approximation of signals) and methods (in particular, the use of randomization.) There are, however, substantial differences between the two, which we now enumerate.

(1) Sampling requirements. The compressed sensing model requires measurement matrices to satisfy the Restricted Isometry Property, which has been shown to hold with high probability for random Gaussian, Bernoulli, and Fourier ensembles. Sparse Fourier algorithms, on the other hand, generally require more

structure in their sampling sets. This is obviously true for the deterministic algorithms, and also for some of the randomized versions — in particular, [Gilbert *et al.* (2005)] requires samples that lie on arithmetic progressions.

(2) Reconstruction costs. As mentioned above, the reconstruction of the target signal in the compressed sensing model is achieved by a convex optimization problem (which can be recast as a linear program). This is expected to incur a computational costs of $O(N^3)$ for a signal of length N. Most sparse Fourier transform algorithms have time complexity that is polylogarithmic in N, and so are exponentially faster.

(3) Allocation of resources. The balance between the two previous items is the major point of distinction. Indeed, we view the comparison of the two paradigms as an "apples-to-oranges" scenario: In the seismic imaging environment (where practitioners have recently implemented compressed sensing methods [Lin and Herrmann (2007); Demanet and Peyré (2011)]), high acquisition costs make long processing times on the back end more palatable. Sparse Fourier transform algorithms, however, were developed with data streaming applications in mind. In this area, low signal acquisition costs and enormous problem sizes necessitate fast algorithms with sparing use of memory resources.

1.3. *New results*

In this paper we describe a simple, deterministic algorithm that avoids reconstruction with the CRT. We are thus able to avoid two pitfalls associated with existing algorithms. Our method relies on sampling the signal in the time domain at slightly shifted points, and thus it assumes access to an underlying continuous-time signal. The shifted time samples allow us to determine the value of significant frequencies in sub-sampled FFTs and also indicate when two or more frequencies have been aliased in such a sub-sampled FFT. These two key facts allow us to significantly reduce (by up to two orders of magnitude) the average-case sampling and runtime complexity of the sparse FFT over a certain class of random signals. Our worst-case bounds improve by a constant factor those of prior deterministic algorithms. We present both adaptive and non-adaptive versions of our algorithms. If the application allows samples to be acquired adaptively (that is, dependent on previous samples), we are able to improve further on our average-case bounds.

The remainder of this paper is organized as follows. In Sec. 2, we introduce notation and prove the technical lemmas underlying our algorithms. In Sec. 3, we introduce randomized and deterministic versions of our algorithm. In Sec. 4, we prove that our algorithm has average-case runtime and sampling complexities of $\Theta(k \log(k))$ and $\Theta(k)$, respectively. In Sec. 5, we present the results of an empirical evaluation of our algorithm and compare its runtime and sampling requirements to competing algorithms. Finally in Sec. 6, we provide some concluding remarks and discuss ongoing work to appear in the future.

2. Mathematical Background

2.1. *Preliminaries*

Throughout this work we shall be concerned with frequency-sparse band-limited signals $S : [0, 1) \to \mathbb{C}$ of the form

$$S(t) = \sum_{j=1}^{k} a_j \mathrm{e}^{2\pi \mathrm{i} \omega_j t}, \tag{5}$$

where $\omega_j \in [-N/2, N/2) \cap \mathbb{Z}, a_j \in \mathbb{C}$, and $k \ll N$. The Fourier series of S is given by

$$\widehat{S}(\omega) = \int_0^1 S(t) \mathrm{e}^{-2\pi \mathrm{i} \omega t} \mathrm{d}t, \quad \omega \in \mathbb{Z}, \tag{6}$$

so that for signals of the form (5) we have $\widehat{S}(\omega_j) = a_j$ and $\widehat{S}(\omega) = 0$ for all other $\omega \in [-N/2, N/2) \cap \mathbb{Z}$. Given any finite sequence $\boldsymbol{S} = (s_0, s_1, \ldots, s_{p-1})$ of length p we define its DFT by

$$\widehat{\boldsymbol{S}}[h] = \sum_{j=0}^{p-1} s_j \mathrm{e}^{\frac{2\pi \mathrm{i} jh}{p}} = \sum_{j=0}^{p-1} \boldsymbol{S}[j] W_p^{jh}, \tag{7}$$

where $h = 0, 1, \ldots, p-1$, $\boldsymbol{S}[j] := s_j$ and $W_p := \mathrm{e}^{-\frac{2\pi \mathrm{i}}{p}}$ is the primitive pth root of unity. The FFT allows the computation of $\widehat{\boldsymbol{S}}$ in $\mathrm{O}(p \log p)$ steps.

We apply the DFT to discrete samples of $S(t)$ to compute the Fourier coefficients a_j of $S(t)$. For an integer p and real $\varepsilon > 0$ we form discrete arrays of samples of S of length p via

$$\boldsymbol{S}_p[j] = S\left(\frac{j}{p}\right), \quad \boldsymbol{S}_{p,\varepsilon}[j] = S\left(\frac{j}{p} + \varepsilon\right), \quad j = 0, 1, \ldots, p-1.$$

Now assume that all $\omega_j \pmod p$, $1 \le j \le k$ are distinct. It is a simple derivation to obtain

$$\widehat{\boldsymbol{S}}_p[h] = \begin{cases} p a_j & h \equiv \omega_j \pmod p \\ 0 & \text{otherwise.} \end{cases}$$

By examining the peaks of $\widehat{\boldsymbol{S}}_p[h]$ we will be able to determine $\{\omega_j \pmod p : 1 \le j \le k\}$. Previous approaches applied the CRT to reconstruct $\{\omega_j\}$ by taking a suitable number of p's, which must overcome the problem of registrations to match up each ω_j whenever a new p is used [see, e.g. Iwen (2010, 2012)]. Our algorithm takes a different approach using the shifted sub-samples. Note that

$$\widehat{\boldsymbol{S}}_{p,\varepsilon}[h] = \begin{cases} p a_j e^{2\pi \mathrm{i} \varepsilon \omega_j} & h \equiv \omega_j \pmod p \\ 0 & \text{otherwise.} \end{cases}$$

It follows that in this setting, for $h \equiv \omega_j \pmod{p}$ we have $\frac{\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]}{\widehat{\boldsymbol{S}}_p[h]} = \mathrm{e}^{2\pi \mathrm{i}\varepsilon\omega_j}$. Hence

$$2\pi\varepsilon\omega_j \equiv \mathrm{Arg}\left(\frac{\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]}{\widehat{\boldsymbol{S}}_p[h]}\right) \pmod{2\pi}, \tag{8}$$

where $\mathrm{Arg}(z)$ denotes the phase angle of the complex number z in $[-\pi, \pi)$. Assume that we take $|\varepsilon| \leq \frac{1}{N}$. Then ω_j is completely determined by (8) as there will be no wrap-around aliasing, and

$$\omega_j = \frac{1}{2\pi\varepsilon}\,\mathrm{Arg}\left(\frac{\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]}{\widehat{\boldsymbol{S}}_p[h]}\right). \tag{9}$$

In fact, more generally, if we have an estimate of ω_j, say $|\omega_j| < \frac{L}{2}$, then by taking $|\varepsilon| \leq \frac{1}{L}$ the same reconstruction formula (9) holds. Note that even though the denominator of (9) contains a very small number ε, it can be verified through Taylor expansion that the numerator is of the same order, so that the ratio is well-behaved in the noiseless case, at least for ω sufficiently far from $\pm\pi$. The observation that by taking slightly shifted samples will allow us to identify frequencies in $S(t)$ underlies the algorithms which follow, and the bulk of this paper analyzes various aspects of the proposed algorithms, such as efficiency and robustness.

One of the problems is that when $p < N$, it is possible that two or more distinct frequencies will have the same remainder modulo p. In this case, we say the frequencies are *aliased* or *collide* $\pmod{p}$. In general, for $h \in \{0, \ldots, p-1\}$ and the given signal $S(t)$ let $I(S, h; p) := \{j : \omega_j \equiv h \pmod{p}\}$. Then we have

$$\widehat{\boldsymbol{S}}_p[h] = \sum_{\omega \equiv h \pmod{p}} \widehat{\boldsymbol{S}}(\omega) = p \sum_{j \in I(S,h;p)} a_j. \tag{10}$$

When aliasing occurs reconstruction via (9) is no longer valid. The aliasing phenomenon presents a serious challenge for any method with sub-linear sampling complexity. In the next section, we develop a simple test to determine whether or not aliasing has occurred in a p-length DFT, which then allows us to effectively overcome this challenge and develop provably correct sub-linear algorithms.

2.2. *Technical lemmas*

To effectively apply the sub-sampling idea in a Fourier algorithm, one must first overcome the aliasing challenge. Using shifted sub-samples gives us a simple yet extremely effective criterion to determine whether or not aliasing has occurred at a given location in a p-length DFT without resorting to complicated combinatorial techniques. Observe that complementing (10) we have

$$\widehat{\boldsymbol{S}}_{p,\varepsilon}[h] = p \sum_{j \in I(S,h;p)} a_j \mathrm{e}^{2\pi \mathrm{i}\varepsilon\omega_j}. \tag{11}$$

It follows that

$$|\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]|^2 - |\widehat{\boldsymbol{S}}_p[h]|^2 = p^2 \sum_{j,l\in I(S,h;p)} a_j\overline{a_l}\mathrm{e}^{2\pi\mathrm{i}\varepsilon(\omega_j-\omega_l)} - p^2\left|\sum_{j\in I(S,h;p)} a_j\right|^2. \quad (12)$$

Lemma 1. *Let $p > 1$ and $h \in \{0, 1, \ldots, p-1\}$. Assume that $q = |I(S,h;p)| > 1$, i.e. $\omega_j \equiv h \,(\mathrm{mod}\, p)$ for more than one j in $S(t)$. Then we have the following*:

(A) *Let $\varepsilon > 0$ and $E := \{\omega_j - \omega_m : j, m \in I(S,h;p)\}$. Suppose that all elements of εE are distinct $(\mathrm{mod}\, 1)$. Then $|\widehat{\boldsymbol{S}}_{p,m\varepsilon}[h]| \neq |\widehat{\boldsymbol{S}}_p[h]|$ for some $1 \leq m \leq q^2 - q$.*
(B) *For almost all $\varepsilon > 0$ we have $|\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]| \neq |\widehat{\boldsymbol{S}}_p[h]|$.*

Proof. The proof of part (B) is immediate from (12). Observe that $f(\varepsilon) := |\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]|^2 - |\widehat{\boldsymbol{S}}_p[h]|^2$ is trigonometric polynomial in ε, and it is not identically 0 given that $q = |I(S,h;p)| > 1$. Thus, it has at most finitely many zeros for $\varepsilon \in [0,1)$, and hence (B) is clearly true.

We resort to the Vandermonde matrix to prove part (A). For simplicity we write $f(t) = \sum_{\alpha\in E} c_\alpha \mathrm{e}^{2\pi\mathrm{i}\alpha t}$. Set $r_\alpha := \mathrm{e}^{2\pi\mathrm{i}\alpha\varepsilon}$ where ε satisfies the hypothesis of the lemma, which implies that all r_j are distinct. Assume the claim of part (A) is false. Then we have $f(m\varepsilon) = 0$ for all $0 \leq m \leq q^2 - q$. Here, $f(0) = 0$ is automatic because $\boldsymbol{S}_{p,0} = \boldsymbol{S}_p$. Thus we have

$$\sum_{\alpha\in E} c_\alpha r_\alpha^m = 0, \quad m = 0, 1, \ldots, q^2 - q. \quad (13)$$

But the cardinality of E is at most $q^2 - q + 1$, which means that there are at most $q^2 - q + 1$ terms in the sum in (13). Because all r_α are distinct the matrix $[r_\alpha^m]$ is a non-singular Vandermonde matrix, and for (13) to hold all c_α must be zero. This is clearly not the case, and a contradiction. □

Remark. Any irrational ε or $\varepsilon = \frac{a}{b}$ with a, b coprime and $b \geq 2N$ will satisfy the hypothesis of part (A) of Lemma 1. It is also easy to show that in the special case where all coefficients a_j are real and $|I(S,h;p)| = 2$, we have $|\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]| \neq |\widehat{\boldsymbol{S}}_p[h]|$ for any $\varepsilon = \frac{a}{b}$ with a, b coprime and $b \geq N$.

Lemma 1 allows us to determine whether aliasing has occurred by whether $|\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]|/|\widehat{\boldsymbol{S}}_p[h]| = 1$ for a few values of ε. It offers both a deterministic (part (B)) and a random (part (A)) procedure to identify aliasing in the sub-sampled DFTs. In practice, we need to set a tolerance τ in order to accept or reject frequencies according to the criterion

$$\left|\frac{|\widehat{\boldsymbol{S}}_{p,\varepsilon}[h]|}{|\widehat{\boldsymbol{S}}_p[h]|} - 1\right| \leq \tau. \quad (14)$$

We typically choose $\varepsilon = 1/cN$ for some small constant $c \geq 2$, which would satisfy the hypothesis of part (A) of Lemma 1. A tolerance on the order of p/N works well in general, which is what we use in our experiments in Sec. 5 below.

In our algorithms, we will take a number of sub-sampled DFTs of an input signal $S(t)$ of the form (5), whose lengths we denote p_ℓ. Lemma 1 allows us to determine whether or not two or more frequencies are aliased, so that we only add the non-aliased term to our representation. Since it is unlikely that two or more frequencies are aliased modulo two different sampling rates, using a different p_ℓ in a subsequent iteration lets us quickly discover all frequencies present in $S(t)$. Lemma 2 gives a worst-case bound on the number of p_ℓ's required by our deterministic algorithm to identify all k frequencies in a given Fourier-sparse signal. It is similar to Iwen [2010, Lemma 1], but with a smaller constant. In its proof we use the CRT, which we quote here for completeness [see, e.g. Niven *et al.* (1991)].

Theorem 1 (Chinese Remainder Theorem). *Any integer n is uniquely specified modulo N by its remainders modulo m pairwise relatively prime numbers p_ℓ, provided $\prod_{\ell=1}^{m} p_\ell \geq N$.*

Lemma 2. *Let $M > 1$. It suffices to take $1 + (k-1)\lfloor \log_M N \rfloor$ pairwise relatively prime p_ℓ's with $p_\ell \geq M$ to ensure that each frequency ω_j is isolated (i.e. not aliased) $(\mathrm{mod}\, p_\ell)$ for at least one ℓ.*

Proof. Assume otherwise, namely that given p_ℓ for $\ell = 1, 2, \ldots, L$ with $L > k\lfloor \log_M N \rfloor$ there exists some ω_j such that ω_j is aliased $(\mathrm{mod}\ p_\ell)$. By the Pigeon Hole Principle there exists at least one $\omega_m \neq \omega_j$ such that $\omega_j - \omega_m \equiv 0 \pmod{p_\ell}$ at least q times, where $q > \lfloor \log_M N \rfloor$. Without loss of generality we assume that $\omega_j - \omega_m \equiv 0 \pmod{p_\ell}$ for $\ell = 1, 2, \ldots, q$. Now by the fact that $p_\ell \geq M$, we have

$$\prod_{\ell=1}^{q} p_\ell \geq M^q \geq N.$$

By the CRT we would then have $\omega_j \equiv \omega_m \pmod{N}$, a contradiction. □

We remark that the algorithm in Iwen [2010] requires taking $1 + 2k \log_k N$ co-prime sample lengths, since that algorithm requires each ω to be isolated in at least half of the DFTs of length p_ℓ. This requirement stems from the fact that that algorithm cannot distinguish between aliased and non-aliased frequencies in a given sub-sampled DFT. Our worst-case bound is approximately a factor of two better, though in practice our algorithms never use all those sample lengths on random input. The fact that we can tell which frequencies are "good" for a given p_ℓ allows us to construct our Fourier representation one term at a time, and quit when we have achieved a prescribed stopping criterion.

3. Algorithms

Both of our algorithms proceed along a similar course; in fact they differ only in the choice of the sample lengths p_ℓ. We assume that we are given access to the continuous-time signal $S(t)$ whose Fourier coefficients we would like to determine, and further that we can sample from S at arbitrary points t in unit time. This is an

appropriate model for analog signals, but not for discrete ones. In the discrete case, one could interpolate between given samples to approximate the required S-values, though we have not implemented or analyzed this case. (The same assumptions hold for the algorithms in Iwen [2010], while those in Gilbert *et al.* [2002, 2005] and Hassanieh *et al.* [2012b] are formulated purely in the discrete realm.) In this paper, mainly limit ourselves to the noiseless case. Though this is a highly unrealistic assumption, it permits a simple description of the underlying algorithm. In Sec. 3.3, we discuss some of the problems associated with noisy signals and give a minor modification of our algorithm for low-level noise. A second manuscript in preparation addresses the issue of noise specifically, with more significant modifications to the algorithms described below.

3.1. *Non-adaptive*

Our algorithms start by choosing a sample length p_1 such that $p_1 \geq ck$ for some constant $c > 1$. For a fixed $\varepsilon \leq 1/N$, we then compute $\widehat{\boldsymbol{S}}_p$ and $\widehat{\boldsymbol{S}}_{p,\varepsilon}$, sort the results by magnitude, and compute frequencies ω via (9) for the k largest coefficients in absolute value. We then check whether or not each of those frequencies is aliased via (10), and if it is not, we add it to our list. The coefficient is given by the unshifted sample value $\widehat{\boldsymbol{S}}_p[h]$ at that frequency. After this, we combine terms with the same frequency and prune small coefficients from the list. We then iterate until a stopping criterion is reached. In the empirical study described in Sec. 5, we stopped when the number of distinct frequencies in our list equalled the desired sparsity.

Our deterministic algorithm chooses p_ℓ to be the ℓth prime greater than ck. This ensures that all samples lengths are co-prime, at the expense of taking slightly more samples than necessary. By Lemma 2, $1+(k-1)\lfloor \log_{ck} N \rfloor$ such p_ℓs suffice to isolate every ω at least once. This gives us worst-case sampling and runtime complexity on the same order as Iwen [2010], though the results in Sec. 5 indicate that on average we significantly outperform those pessimistic bounds.

Our Las Vegas algorithm chooses p_ℓ uniformly at random from the interval $[c_1 k, c_2 k]$ for constants $1 < c_1 < c_2$. In this case we cannot make a worst-case guarantee on the number of iterations needed by the algorithm to converge. However, the results in Sec. 5 indicate that the Las Vegas version performs similarly to the deterministic version on the class of signals tested.

3.2. *Adaptive*

The algorithms can also be implemented in an adaptive fashion, by which we mean that the size of the current representation is taken into account in subsequent iterations. In particular, if $\boldsymbol{R}$ is our current representation, we let $k^* = k - |\boldsymbol{R}|$ and choose the next p_ℓ with respect to k^* instead of k. Moreover, before taking DFTs, we subtract off the contribution from the current representation, so that effort is not expended re-identifying portions of the spectrum already discovered. This idea is similar to that in Gilbert *et al.* [2002, 2005], though in our empirical studies

the evaluation of the representation is done directly, rather than as an unequally-spaced FFT. This gives our algorithms asymptotically slower runtime, but the effect is negligible for the values of k studied in Sec. 5. A formal description appears below in Algorithm 1.

Algorithm 1. PHASESHIFT

Input: function pointer S, integers c_1, c_2, k, N, real ε
Output: $\boldsymbol{R}$, a sparse representation for $\widehat{S}$
$\boldsymbol{R} \leftarrow \emptyset$, $\varepsilon_0 \leftarrow 0, \varepsilon_1 \leftarrow \varepsilon$, $\ell \leftarrow 1$
while $|\boldsymbol{R}| < k$ **do**
  $k^* \leftarrow k - |\boldsymbol{R}|$ {or k if non-adaptive}
  $p_\ell \leftarrow$ first prime $\geq c_1 k^*$
    {or UNIFORM$(c_1 k^*, c_2 k^*)$ if Las Vegas}
  for $m = 0$ to 1 **do**
    for $j = 0$ to $\ell - 1$ **do**
      $\boldsymbol{S}_{\ell,m}[j] \leftarrow S\left(\dfrac{j}{p_\ell} + \varepsilon_m\right)$
      $\boldsymbol{S}_{\text{rep}}[j] \leftarrow \displaystyle\sum_{(\omega, c_\omega) \in \boldsymbol{R}} c_\omega \mathrm{e}^{2\pi \mathrm{i} \omega (j/p_\ell + \varepsilon_m)}$
        {omit if non-adaptive}
    end for
    $\widehat{\boldsymbol{S}}_{\ell,m} \leftarrow \mathrm{FFT}(\boldsymbol{S}_{\ell,m} - \boldsymbol{S}_{\text{rep}})$
    $\widehat{\boldsymbol{S}}^{\text{sort}}_{\ell,m} \leftarrow$ SORT$(\widehat{\boldsymbol{S}}_{\ell,m})$
    for $j = 1$ to k^* **do**
      $\omega_{j,\ell} \leftarrow \dfrac{1}{2\pi\varepsilon} \operatorname{Arg}\left(\dfrac{\widehat{\boldsymbol{S}}^{\text{sort}}_{\ell,1}[j]}{\widehat{\boldsymbol{S}}^{\text{sort}}_{\ell,0}[j]}\right)$
    end for
  end for
  for $j = 1$ to k^* **do**
    if $\left|\dfrac{\left|\widehat{\boldsymbol{S}}^{\text{sort}}_{\ell,0}[j]\right|}{\left|\widehat{\boldsymbol{S}}^{\text{sort}}_{\ell,1}[j]\right|} - 1\right| < \dfrac{p_\ell}{N}$ **then**
      $\boldsymbol{R} \leftarrow \boldsymbol{R} \cup \left\{\left(\omega_{j,\ell}, \widehat{\boldsymbol{S}}_{\ell,0}[\omega_{j,\ell}]\right)\right\}$
    end if
  end for
  collect terms in $\boldsymbol{R}$ with same ω
  prune small coefficients from $\boldsymbol{R}$
  $\ell \leftarrow \ell + 1$
end while

3.3. *Modifications in the presence of noise*

In the noiseless versions of the algorithms described in this paper, a test for aliasing is implemented by considering the ratio of magnitudes of shifted and unshifted peaks. When the samples are corrupted by noise, there will be two challenges. The first challenge is that the reconstruction of frequencies from shifts will be corrupted by noise. The second challenge is that there will be variations among the magnitudes even for non-aliased terms, so a higher threshold that depends on the size of the noise must be set. When this threshold is too large it affects the ability to distinguish aliased terms as there will be an increased number of false negatives. On the other hand, lower thresholds that reduce false negatives will lead to an increased number of false positives.

The first challenge can be addressed effectively through a combination of using larger p_j's, multiple shifts and a multiscale unwrapping. The idea of using larger p_j's is rather straightforward yet effective. For any given p_j the DFT detects the location of the frequencies modulo p_j rather accurately even with substantial noise. Furthermore, the reconstructed frequencies will still tend to cluster around the true value. Suppose, that we sample the signal and compute DFTs of length p_j on these samples. The locations of the peaks in these short DFTs tell us the accurate value of $\omega \bmod p_j$ for each unaliased frequency ω appearing in the signal. Writing $\omega = ap_j + b$ with $a, b \in \mathbb{Z}$, we now know b and must determine a.

With a small amount of noise the reconstructed frequencies $\widetilde{\omega}$ using (9) will be close to the true ω. We can thus round $\widetilde{\omega}$ to the nearest integer of the form $ap_j + b$, which will recover the true frequency ω as long as $|\widetilde{\omega} - \omega| < p_j/2$. For high noise levels, it is possible that the $\widetilde{\omega}$ will deviate by more than $p_j/2$ from ω, so that the value for a given by rounding will be incorrect. By choosing larger p_j (i.e. increasing the parameter c_1) one can alleviate the problem somewhat, provided that the noise level is not too high. When the noise level is so high that taking a large p_j is no longer economical, a potential solution is to take multiple shifts and employ a multiscale unwrapping technique. We are still at the preliminary stage in our study of these new techniques, but early results are very encouraging.

The second challenge poses a bigger problem, but again it can be addressed in several ways. The multiscale unwrapping method will repeatedly check for aliasing at each stage, which makes it highly unlikely that an aliased frequency will pass through all the tests. Even in the unlikely even that it does, our algorithm allows false positives. Since each mode is subtracted from the original signal in our algorithm, a false positive frequency will lead to an extra mode in the new signal. As the process continues it will be extracted and cancel out the false frequency extracted earlier.

4. Average-Case Analysis

In this section, we prove that the average-case runtime and sampling complexity of our algorithm are $\Theta(k \log k)$ and $\Theta(k)$, respectively. This is shown over a class

of random signals described in Sec. 4.2. Before giving this result on the expected runtime and sampling complexity, in Sec. 4.1 estimate the costs of a single iteration of the while loop in Algorithm 1, lines 5–25. We then describe in Sec. 4.2, the random signal model over which we prove our average-case bounds. In Sec. 4.3, we prove that the expected number of iterations of the while loop is constant, and in Sec. 4.4, we use this result to prove our average-case bounds.

4.1. *While loop runtime and sampling complexity*

The computational cost of the while loop in Algorithm 1, lines 5–25 is dominated by three operations. The first is the evaluation of the current representation $\boldsymbol{R}$ of $k - k*$ terms at the $\mathrm{O}(k^*)$ points j/p_ℓ in line 10. In our implementation, we simply calculated this directly, looping over both the sample points and the terms in the representation. The complexity of this implementation is $\mathrm{O}(p_\ell(k - k^*)) = \mathrm{O}(k^*(k - k^*)) = \mathrm{O}(k^2)$, and while non-equispaced FFT [Dutt and Rokhlin (1993); Anderson and Dahleh (1996)] yield an asymptotically faster runtime of $\mathrm{O}(k \log(k))$, they also incur large overhead costs. For the values of k considered in this paper, the direct evaluation seems to have little effect on the overall runtime. The other two dominant computational tasks in the inner loop are the FFTs of $\mathrm{O}(k)$ samples and the subsequent sorting of these DFT coefficients. It is well-known that both of these operations can be done in time $\Theta(k \log(k))$ [Cormen *et al.* (2001)]. Thus the inner loop has overall time complexity $\Theta(k \log(k))$, assuming the use of non-equispaced FFTs.

4.2. *Random signal model*

For both the average-case analysis and for the empirical evaluation described in Sec. 5, we considered test signals with uniformly random phase over the bandwidth and coefficients chosen uniformly from the complex unit circle. In other words, given k and N, we choose k frequencies ω_j uniformly at random (without replacement) from $[-N/2, N/2) \cap \mathbb{Z}$. The corresponding Fourier coefficients a_j are of the form $\mathrm{e}^{2\pi \mathrm{i} \theta_j}$, where θ_j is drawn uniformly from $[0, 1)$. The signal is then given by

$$S(t) = \sum_{j=1}^{k} a_j \mathrm{e}^{2\pi \mathrm{i} \omega_j t}. \tag{15}$$

This is the standard signal model considered in previous empirical evaluations of sub-linear Fourier algorithms [Iwen *et al.* (2007); Iwen (2010); Hassanieh *et al.* (2012b)].

4.3. *Markov analysis of collisions*

In order to analyze the expected runtime and sampling complexity of our algorithms, we must estimate the expected number of collisions among frequencies

modulo the sample lengths used by the algorithms. Recall that in the noiseless case, our algorithms are able to detect when a collision between two or more frequencies has occurred, and for those that are not aliased we are able to calculate the value of the frequency. Thus we seek to estimate the expected fraction of frequencies that are aliased modulo a given sample length p, since this determines how many passes the algorithm makes. In this section, we derive bounds on the expected value of this quantity and discuss how the stopping criteria used in the algorithm affect its average-case performance.

In the random signal model considered in Sec. 5, we assume the k frequencies are uniformly distributed over the bandwidth $[-N/2, N/2)$, and so the residues $\omega \bmod p$ are also uniformly distributed over $[0, p-1]$. Our problem then becomes a classical occupancy problem: The number of collisions among the frequencies is equivalent to the number of multiple-occupancy bins when k balls are thrown uniformly at random into p bins. Define X_m to be the number of single-occupancy bins after m balls are thrown, Y_m to be the number of multiple-occupancy bins after m balls are thrown, and Z_m to be the number of zero-occupancy bins after m balls are thrown. Since p is constant, we have the trivial relationship $Z_m = p - X_m - Y_m$, so it suffices to consider only the pair (X_m, Y_m). When the $(m+1)$st ball is thrown, we have the following possibilities:

- it lands in an unoccupied bucket, with probability $Z_m/p = 1 - (X_m + Y_m)/p$;
- it lands in a single-occupancy bucket, with probability X_m/p;
- it lands in a multiple-occupancy bucket, with probability Y_m/p.

In the first case, we have $X_{m+1} = X_m + 1, Y_{m+1} = Y_m$; in the second case, we have $X_{m+1} = X_m - 1, Y_{m+1} = Y_m + 1$; and in the third case, we have $X_{m+1} = X_m, Y_{m+1} = Y_m$. Conditioning on the values of X_m, Y_m we have

$$\mathbb{E}\left(\begin{bmatrix} X_{m+1} \\ Y_{m+1} \end{bmatrix} \middle| \begin{bmatrix} X_m \\ Y_m \end{bmatrix}\right) = \begin{bmatrix} 1 - 2/p & -1/p \\ 1/p & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \tag{16}$$

so that the system forms a Markov chain. By recursively conditioning on the values of X_{m-1}, Y_{m-1}, we can calculate the expected values of X_k, Y_k for any $k > 0$ using the initial condition $X_1 = 1, Y_1 = 0$. Denoting by A the matrix in the right-hand side of Eq. (16), we have

$$\mathbb{E}\left(\begin{bmatrix} X_k \\ Y_k \end{bmatrix}\right) = \sum_{m=0}^{k-1} \left(A^m \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right) = \left(\sum_{m=0}^{k-1} A^m\right) \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{17}$$

Since $\rho(A) = 1 - 1/p < 1$, where ρ is the spectral radius, the geometric matrix series can be written

$$\sum_{m=0}^{k-1} A^m = (I - A)^{-1}(I - A^k). \tag{18}$$

After some linear algebra, we obtain

$$\mathbb{E}\left(\begin{bmatrix} X_k \\ Y_k \end{bmatrix}\right) = \begin{bmatrix} k\left(1-\frac{1}{p}\right)^{k-1} \\ p\left(1-\left(1-\frac{1}{p}\right)^{k}\right) - k\left(1-\frac{1}{p}\right)^{k-1} \end{bmatrix}. \tag{19}$$

Since $Z_k = p - X_k - Y_k$, we have $\mathbb{E}(Z_k) = p(1-1/p)^k$.

In our algorithms, we choose $p = ck$ for some small integer c. Using this and the approximation $(1+\frac{x}{n})^n \approx \mathrm{e}^x$, we have

$$\mathbb{E}\left(\begin{bmatrix} X_k \\ Y_k \end{bmatrix}\right) \approx \begin{bmatrix} k\mathrm{e}^{-1/c} \\ ck(1-\mathrm{e}^{-1/c}) - k\mathrm{e}^{-1/c} \end{bmatrix}. \tag{20}$$

This gives a non-linear equation for the expected number of collisions among k frequencies as a function of the parameter c. Newton's method can then be used to determine the value c required to ensure a desired fraction of the frequencies are not aliased. For example, to ensure that 90% of frequencies are isolated on average, it suffices to take $c = 5$; this value for the parameter c had already been found to give good performance in our empirical evaluation of the algorithms.

4.4. *Average-case runtime and sampling complexity*

In this section, we will use a probabilistic recurrence relation due to Karp [Karp (1994); Dubhashi and Panconesi (2009)] to give average-case performance bounds and concentration results for the case when the algorithm is halted after identifying k or more terms. In particular, we use the following theorem for recurrences of the form

$$T(k) = a(k) + T(H(k)), \tag{21}$$

where $T(k)$ denotes the time required to solve an instance of size k, $a(k)$ is the amount of work done on a problem of size k, and $0 \leq H(k) \leq k$ is a random variable denoting the size of the subproblem generated by the algorithm.

Theorem 2. [Karp (1994, Theorem 1.2)] *Suppose $a(k)$ is non-decreasing, continuous, and strictly increasing on $\{x : a(x) > 0\}$, and that $\mathbb{E}[H(k)] \leq m(k)$ for a non-decreasing continuous function $m(k)$ such that $m(k)/k$ is also non-decreasing. Denote by $u(k)$ the solution to the deterministic recurrence*

$$u(k) = a(k) + u(m(k)). \tag{22}$$

Then for $k > 0$ and $t \in \mathbb{N}$,

$$\mathbb{P}[T(k) > u(k) + ta(k)] \leq \left(\frac{m(k)}{k}\right)^t. \tag{23}$$

Our algorithm does work $a(k) = \Theta(k \log(k))$ on input of size k and generates a subproblem whose average size is $m(k) = k/10$. (Recall from Sec. 4.3 that with the parameter $c = 5$, on average over 90% of the frequencies were not aliased modulo $p = \mathrm{O}(ck)$.) The associated deterministic recurrence is then

$$u(k) = \Theta(k \log(k)) + u(k/10), \tag{24}$$

whose solution is $u(k) = \Theta(k \log(k))$ [see, e.g. Cormen *et al.* (2001)]. A straightforward application of Theorem 2 yields the following

Theorem 3 (Runtime bound). *Let $T(k)$ denote the runtime of Algorithm* 1 *on a random signal drawn from the class in Sec.* 4.2. *Then $\mathbb{E}[T(k)] = \Theta(k \log(k))$ and*

$$\mathbb{P}[T(k) > \Theta(k \log(k)) + tk \log(k)] \leq 10^{-t}. \tag{25}$$

The sampling complexity $S(k)$ can be handled in an analogous manner, since in this case $a(k) = \Theta(k)$ and $m(k) = k/10$ as before. The associated deterministic recurrence becomes

$$u(k) = \Theta(k) + u(k/10), \tag{26}$$

whose solution is $u(k) = \Theta(k)$. Applying Theorem 2 again we have the following

Theorem 4 (Sampling bound). *Let $S(k)$ denote the number of samples used by Algorithm* 1 *on a random signal drawn from the class in Sec.* 4.2. *Then $\mathbb{E}[S(k)] = \Theta(k)$ and*

$$\mathbb{P}[S(k) > \Theta(k) + tk] \leq 10^{-t}. \tag{27}$$

5. Empirical Evaluation

In this section, we describe the results of an empirical evaluation of the *adaptive* deterministic and Las Vegas variants of the Phaseshift algorithm described above. Both algorithms were implemented in C++ using FFTW 3.0 [Frigo and Johnson (2005)] for the FFTs, using `FFTW_ESTIMATE` plans since the sample lengths are not known in advance for the Las Vegas variant. For comparison, we also ran the same tests on the four variants of GFFT as well as on AAFFT and FFTW itself. The FFTW runs utilized the `FFTW_PATIENT` plans with wisdom enabled, and so are highly optimized. The experiments were run on a single core of an Intel Xeon E5620 CPU with a clock speed of 2.4 GHz and 24 GB of RAM, running SUSE Linux with kernel 2.6.16.60-0.81.2-smp for x86_64. All code was compiled with the Intel compiler using the `-fast` optimization. As in Iwen [2012], timing is reported in CPU ticks using the `cycle.h` file included with the source code for FFTW.

In the following sections, we refer to our algorithm as "Phaseshift", since by taking shifted time samples of the input signal we also shift the phase of the Fourier coefficients. To keep the plots readable, we only show data for the adaptive,

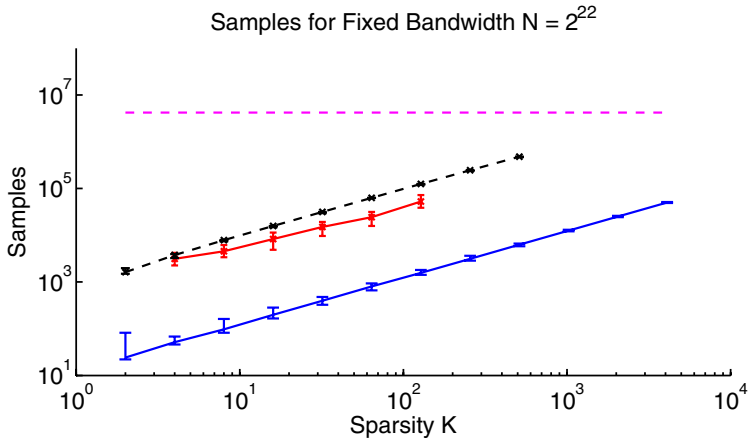
deterministic variant of our algorithm; the other variants perform similarly the algorithms of Iwen [2012] are denoted GFFT-XY, where X ∈ {D,R} and Y ∈ {F,S}. The D/R stands for deterministic or randomized, while the F/S stands for fast or slow. The fast variants use more samples but less runtime while the slow variants use fewer samples but more runtime. In the plots below, we always show the GFFT variant with the most favorable sampling or runtime complexity. Finally, AAFFT denotes the algorithm of Gilbert *et al.* [2005]. The implementations tested are summarized in Table 1 along with the average-case sampling and runtime complexities, and the associated references.

5.1. *Setup*

Each data point in Figs. 1–2 is the average of 100 independent trials of the associated algorithm for the given values of the bandwidth N and the sparsity k. The lower and upper bars associated with each data point represent the minimum and maximum number of samples or runtime of the algorithm over the 100 test functions. The

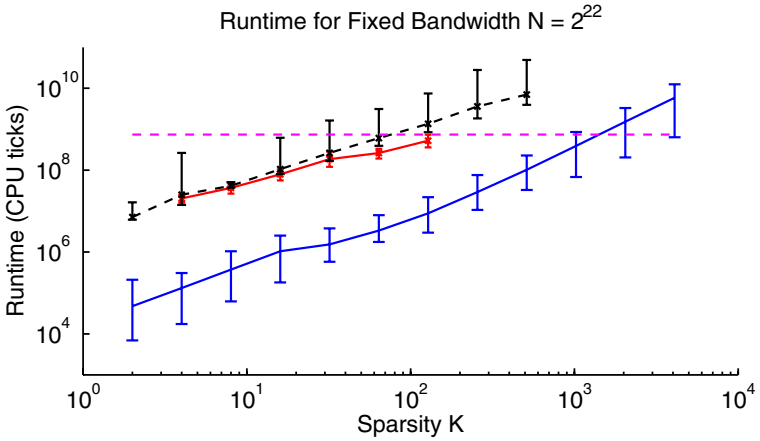
Table 1. Implementations used in the empirical evaluation.

Algorithm	R/D	Samples	Runtime	Reference
PS-Det	D	k	$k \log k$	Section 4
PS-LV	R	k	$k \log k$	Section 4
GFFT-DF	D	$k^2 \log^4 N$	$k^2 \log^4 N$	[Iwen (2012)]
GFFT-DS	D	$k^2 \log^2 N$	$Nk \log^2 N$	[Iwen (2012)]
GFFT-RF	R	$k \log^4 N$	$k \log^5 N$	[Iwen (2012)]
GFFT-RS	R	$k \log^2 N$	$N \log N$	[Iwen (2012)]
AAFFT	R	$k \log^c N$	$k \log^c N$	[Gilbert *et al.* (2005)]
FFTW	D	N	$N \log N$	[Frigo and Johnson (2005)]



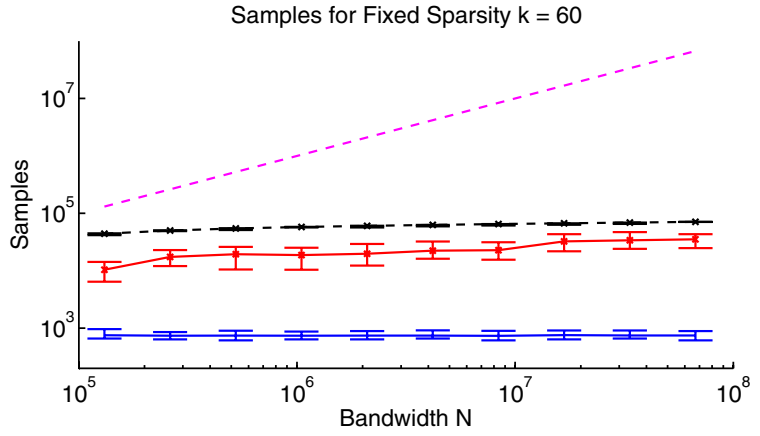
(a)

Fig. 1. (Color online) (a) Sampling complexity with fixed bandwidth $N = 2^{22}$ for PS-Det (blue solid line), GFFT-RS (red solid line), AAFFT (black dashed line), and FFTW (magenta dashed line). (b) Runtime complexity with fixed bandwidth $N = 2^{22}$ for PS-Det (blue solid line), GFFT-RF (red solid line), AAFFT (black dashed line), and FFTW (magenta dashed line).

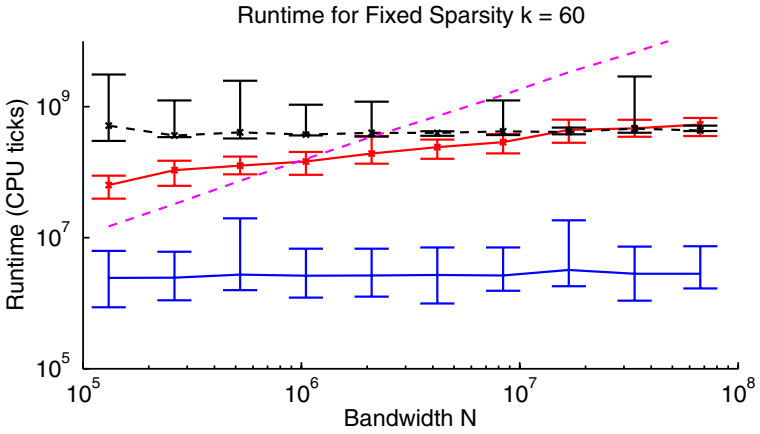


(b)

Fig. 1. (*Continued*)



(a)



(b)

Fig. 2. (Color online) (a) Sampling complexity with fixed sparsity $k = 60$ for PS-Det (blue solid line), GFFT-RS (red solid line), AAFFT (black dashed line), and FFTW (magenta dashed line). (b) Runtime complexity with fixed sparsity $k = 60$ for PS-Det (blue solid line), GFFT-RF (red solid line), AAFFT (black dashed line), and FFTW (magenta dashed line).

values of k tested were $2, 4, 8, \ldots, 4096$, while the values of N were $2^{17}, 2^{18}, \ldots, 2^{26}$. For larger values of k, the slow GFFT variants and AAFFT took too long to complete on our hardware, so we only present partial data for these algorithms. Nevertheless, the trend seen in the plots below continues for higher values of the sparsity. The test signals were generated according to the signal model described in Sec. 4.2.

The Phaseshift and deterministic GFFT variants will always recover such signals exactly. The randomized GFFT variants are Monte Carlo algorithms, and so, when they succeed, will also recover the signal exactly. AAFFT, on the other hand, is an approximation algorithm which will fail on a non-negligible set of input signals. However, for the runs depicted in Figs. 1–2, AAFFT always produced an answer with ℓ_2 error less than 10^{-4}. The randomized GFFT variants failed a total of 7 times out of 2,200 test signals, a relatively small amount that can be reduced by parameter tuning. For the Phaseshift variants, we chose the parameters $c_1 = 5, c_2 = 10$, and took the shift ε to be $1/2N$. Finally, for the randomized GFFT variants, we chose the Monte Carlo parameter to be 1.2.

5.2. *Sampling complexity*

In Fig. 1(a), we compare the average number of samples of the input signal S required by each algorithm when the bandwidth N fixed at 2^{22}. The sparsity of the test signal is varied from 2 to 4096 by powers of two. We can see that the Phaseshift variants require over an order of magnitude fewer samples than GFFT-RS, the GFFT variant with the lowest sampling requirements. Both Phaseshift variants also require over an order of magnitude fewer samples than AAFFT. The comparison with the deterministic GFFT variants is even starker; Phaseshift-Det requires two orders of magnitude fewer samples than GFFT-DS (not shown), and four orders of magnitude fewer samples than GFFT-DF (not shown).

In Fig. 2(a), we compare the average number of samples of the input signal S required by each algorithm when the sparsity k is fixed at 60. The bandwidth N was varied from 2^{17}–2^{26} by powers of two. Using powers of two for the bandwidth allows the best performance for both FFTW and AAFFT, though this fact is more relevant for the runtime comparisons in the following section. We can see that the Phaseshift variants require many fewer samples than all four GFFT variants as well as AAFFT and FFTW, for all values of N tested. The Phaseshift variants exhibit almost no dependence on the bandwidth for all values of N, a feature not shared by the other deterministic algorithms. We note here that in future work we plan to replace the $1/2N$ shift by two or more larger shifts with co-prime denominators to obtain an equivalent shift, as in Wang and Zhou [1998]. This should lead to more robustness at high values of N.

5.3. *Runtime complexity*

In Fig. 1(b), we compare the average runtime of each algorithm over 100 test signals when the bandwidth N is fixed at 2^{22}. The range of sparsity k considered is the

same as in Sec. 5.2. For all values of k the Phaseshift variants are faster than GFFT-RF (the fastest GFFT variant) and AAFFT by more than an order of magnitude. When compared to GFFT-RS (not shown), GFFT-DS (not shown), and FFTW, the difference in runtime is closer to three orders of magnitude.

In Fig. 2(b), we compare the average runtime of each algorithm over 100 test signals when the sparsity k is fixed at 60. The range of bandwidth considered is the same as in Sec. 5.2. The Phaseshift variants are the only algorithms that outperform FFTW for all values of N tested. The other implementations tested only become competitive with the standard FFT for $N \gtrsim 2^{20}$, while ours are faster even for modest N.

5.4. *Noisy case*

We report here on a preliminary study of the performance of the deterministic algorithm in the presence of noise. Our noisy signals were of the same form as in the previous section, but with complex white gaussian noise of standard deviation σ added to each measurement. As described in Sec. 3.3, the simplest way to deal with low-level noise is to simply round the reconstructed frequencies to the nearest integer of the form $ap_j + b$, where $b \equiv \omega \bmod p_j$ is the location of the peak in a length-p_j DFT. This modification doesn't change the runtime or sampling complexity significantly, so in this section we focus on the error in the approximation as a function of the noise level σ and the parameter c_1.

In the existing literature on the sparse Fourier transform, the ℓ_2 norm is most often used to assess the quality of approximation. There are many reasons for this choice, with the two most convincing perhaps being the completeness of the complex exponentials with respect to the ℓ_2 norm and Parseval's theorem. For certain applications, however, this choice of norm is inappropriate. For example, in wide-band spectral estimation and radar applications, one is interested in identifying a set of frequency intervals containing active Fourier modes. In this case, an estimate $\widetilde{\omega}$ of the true frequency ω with $|\widetilde{\omega} - \omega| \ll N$ is useful, but unless $\widetilde{\omega} = \omega$ the ℓ_2 metric will report an O(1) error. Furthermore, when considering non-periodic signals (equivalently, non-integer ω's) the same precision problem appears when using the ℓ_2 metric.

For these reasons, we propose measuring the approximation error of sparse Fourier transform problems with the Earth Mover Distance (EMD) [Rubner *et al.* (2000)]. Originally developed in the context of content-based image retrieval, EMD measures the minimum cost that must be paid (with a user-specified cost function) to transform one distribution of points into another. EMD can be calculated efficiently as the solution of a linear program corresponding to a certain flow minimization problem. In our situation, we consider the cost to move a set of estimated Fourier modes and coefficients $\{(\widetilde{\omega}_j, c_{\widetilde{\omega}_j})\}_{j=1}^{\widetilde{k}}$ to the true values $\{(\omega_i, c_{\omega_j})\}_{j=1}^{k}$ under the cost function

$$d_1((\omega, c_\omega), (\widetilde{\omega}, c_{\widetilde{\omega}}); N) \stackrel{\text{def}}{=} \frac{|\omega - \widetilde{\omega}|}{N} + |c_\omega - c_{\widetilde{\omega}}|. \tag{28}$$

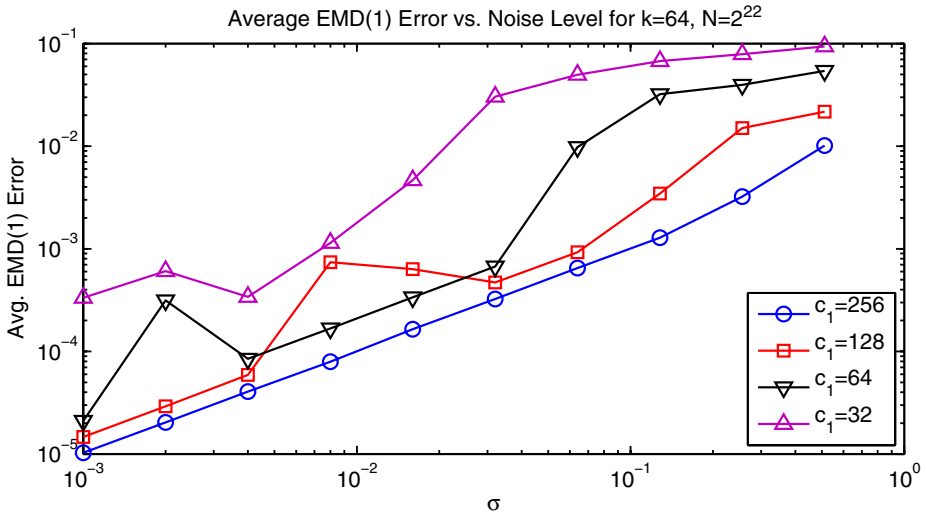


Fig. 3. EMD(1) error as a function of the noise level σ for various choices of the parameter c_1. The sparsity and bandwidth are fixed at $k = 64$, $N = 2^{22}$, respectively.

This choice of cost function strikes a balance between the fidelity of the frequency estimate (as a fraction of the bandwidth) and that of the coefficient estimate. We denote the EMD using d_1 for the cost function by EMD(1) below.

In Fig. 3, we report the average EMD(1) error over 100 test signals as a function of the input noise level σ, for various choices of the parameter c_1. In this experiment, the sparsity and bandwidth are fixed at $k = 64$ and $N = 2^{22}$, respectively. As expected, the error decreases as c_1 increases, since the rounding procedure described in Sec. 3.3 is more likely to result in the true frequency. Moreover, the error increases linearly with the noise level, indicating the procedure's robustness in the presence of noise.

We remark that in the noiseless case the choice $c_1 = 5$ was found to be sufficient, while Fig. 3 indicates that the much larger value $c_1 \approx 256$ is necessary for good approximation in the EMD(1) metric. The larger sample lengths imply an increase in both the runtime and sampling complexity, and indicate that the rounding procedure of Sec. 3.3 should be complemented by other modifications. This is the purpose of a second manuscript under preparation, in which we combine the rounding procedure with the use of larger shifts ε_j in a multiscale approach to frequency estimation.

6. Conclusion

In this paper, we have presented a deterministic and Las Vegas algorithm for the sparse Fourier transform problem that empirically outperform existing algorithms in average-case sampling and runtime complexity. While our worst-case bounds do not improve the asymptotic complexity, we are able to extend by an order of magnitude the range of sparsity for which our algorithm is faster than FFTW in the average case. The improved performance of our algorithm can be attributed

to two major factors: adaptivity and ability to detect aliasing. In particular, we are able to extract more information from a small number of function samples by considering the *phase* of the DFT coefficients in addition to their magnitudes. This represents a significant improvement over the current state of the art for the sparse Fourier transform problem.

We have developed a multiresolution approach to handle the noisy case, in which we learn the value of a frequency from most to least significant bit by increasing the size of the shift ε. Finally, we are exploring the extension of these methods to handle non-integer frequencies, which would represent the first such result in the sparse Fourier transform context.

Acknowledgments

We thank the anonymous referees for useful comments to improve the exposition of this paper. We would like to thank Mark Iwen and I. Ben Segal for making available the source code to the AAFFT and GFFT algorithms, Yossi Rubner for making available the source code for the Earth Mover Distance, and Piotr Indyk and Eric Price for sharing a preprint and source code for the sFFT algorithm. We also acknowledge helpful discussions with Anna Gilbert and Martin Strauss.

References

Ajtai, M., Iwaniec, H., Komlós, J., Pintz, J. and Szemerédi, E. (1990). Construction of a thin set with small Fourier coefficients. *Bull. London Math. Soc.*, **22**: 583–590.

Akavia, A. (2010). Deterministic sparse Fourier approximation via fooling arithmetic progressions. *Conference on Learning Theory* (*CoLT*).

Akavia, A., Goldwasser, S. and Safra, S. (2003). Proving hard-core predicates using list decoding. *Annual Symposium on Foundations of Computer Science*, pp. 146–159.

Anderson, C. and Dahleh, M. D. (1996). Rapid computation of the discrete Fourier transform. *SIAM J. Sci. Comput.*, **17**: 913–919.

Boneh, D. (2002). Finding smooth integers in short intervals using crt decoding. *J. Comput. Syst. Sci.*, **64**: 768–784.

Candès, E. J., Romberg, J. K. and Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, **52**: 489–509.

Chen, S., Donoho, D. and Saunders, M. (1998). Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, **20**: 33–61.

Cohen, A., Dahmen, W. and DeVore, R. (2009). Compressed sensing and best k-term approximation. *J. AMS*, **22**: 211–231.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C. (2001). *Introduction to Algorithms*, 2nd edn. MIT Press, Cambridge, MA.

Demanet, L. and Peyré, G. (2011). Compressive wave computation. *Found. Comput. Math.*, **11**: 257–303.

Dongarra, J. and Sullivan, F. (2000). Guest editors' introduction: The top 10 algorithms. *Comput. Sci. Eng.*, **2**(1): 22–23.

Dubhashi, D. P. and Panconesi, A. (2009). *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, Cambridge.

Dutt, A. and Rokhlin, V. (1993). Fast Fourier transforms for nonequispaced data. *SIAM J. Sci. Comput.*, **14**: 1368–1393.

Frigo, M. and Johnson, S. G. (2005). The design and implementation of FFTW3. *Proceedings of the IEEE*, **93**: 216–231, special issue on "Program Generation, Optimization, and Platform Adaptation".

Gilbert, A., Guha, S., Indyk, P., Muthukrishnan, S. and Strauss, M. (2002). Near-optimal sparse Fourier representations via sampling. *Symposium on Theory of Computing*, pp. 152–161.

Gilbert, A., Muthukrishnan, S. and Strauss, M. (2005). Improved time bounds for near-optimal sparse Fourier representations. *SPIE Wavelets XI*.

Gilbert, A., Strauss, M. and Tropp, J. (2008). A tutorial on fast Fourier sampling. *IEEE Sig. Process. Magaz.*, **25**: 57–66.

Goldreich, O., Ron, D. and Sudan, M. (2000). Chinese remaindering with errors. *IEEE Trans. Inform. Theory*, **46**: 1330–1338.

Hassanieh, H., Indyk, P., Katabi, D. and Price, E. (2012a). Nearly optimal sparse fourier transform. *Proceedings of 44th ACM Symposium on Theory of Computing* (*STOC*), pp. 563–578.

Hassanieh, H., Indyk, P., Katabi, D. and Price, E. (2012b). Simple and practical algorithms for sparse Fourier transform. *Proceedings of 23rd ACM-SIAM Symposium on Discrete Algorithms* (*SODA*), pp. 1183–1194.

Iwen, M. (2008). A deterministic sub-linear time sparse Fourier algorithm via non-adaptive compressed sensing methods. *Proceedings of the 19th annual ACM-SIAM Symposium on Discrete Algorithms* (*SODA*), Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, pp. 20–29.

Iwen, M. (2010). Combinatorial sublinear-time Fourier algorithms. *Found. Comput. Math.*, **10**: 303–338.

Iwen, M. (2012). Improved approximation guarantees for sublinear-time Fourier algorithms. *Appl. Comp. Harm. Anal.*, **24**(1): 57–82.

Iwen, M., Gilbert, A. and Strauss, M. (2007). Empirical evaluation of a sub-linear time sparse DFT algorithm. *Commun. Math. Sci.*, **5**: 981–998.

Karp, R. M. (1994). Probabilistic recurrence relations. *J. Assoc. Comput. Mach.*, **41**: 1136–1150.

Katz, N. (1989). An estimate for character sums. *J. Amer. Math. Soc.*, **2**: 197–200.

Kunis, S. and Rauhut, H. (2008). Random sampling of sparse trigonometric polynomials. II. Orthogonal matching pursuit versus basis pursuit. *Found. Comput. Math.*, **8**: 737–763.

Kushilevitz, E. and Mansour, Y. (1993). Learning decision trees using the Fourier spectrum. *SIAM J. Comput.*, **22**: 1331–1348.

Lin, T. and Herrmann, F. (2007). Compressed wavefield extrapolation. *Geophysics*, **72**: SM77–SM93.

Linial, N., Mansour, Y. and Nisan, N. (1993). Constant depth circuits, Fourier transform, and learnability. *J. Assoc. Comput. Mach.*, **40**: 607–620.

Mansour, Y. (1995). Randomized interpolation and approximation of sparse polynomials. *SIAM J. Comput.*, **24**: 357–368.

Niven, I., Zuckerman, H. and Montgomery, H. (1991). *An Introduction to the Theory of Numbers*, 5th edn. John Wiley & Sons Inc., New York.

Pinkus, A. (1989). *On L^1-approximation*, Cambridge Tracts in Mathematics, Vol. 93, Cambridge University Press, Cambridge.

Rauhut, H. (2007). Random sampling of sparse trigonometric polynomials. *Appl. Comput. Harmon. Anal.*, **22**: 16–42.

Rubner, Y., Tomasi, C. and Guibas, L. (2000). The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vis.*, **40**: 99–121.

Rudin, L., Osher, S. and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlin. Phenom.*, **60**: 259–268.

Santosa, F. and Symes, W. (1986). Linear inversion of band-limited reflection seismograms. *SIAM J. Sci. Statist. Comput.*, **7**: 1307–1330.

Shparlinski, I. and Steinfeld, R. (2004). Noisy chinese remaindering in the Lee norm. *J. Complex.*, **20**: 423–437.

Tao, T. and Vu, V. (2006). *Additive Combinatorics*, Cambridge Studies in Advanced Mathematics, Vol. 105, Cambridge University Press, Cambridge.

Tropp, J. A. and Gilbert, A. C. (2007). Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inform. Theory*, **53**: 4655–4666.

Wang, Y. and Zhou, G. (1998). On the use of high-order ambiguity function for multi-component polynomial phase signals. *Sig. Process.*, **65**: 283–296.

Xu, Z. (2011). Deterministic sampling of sparse trigonometric polynomials. *J. Complex.*, **27**: 133–140.

MPI–OpenMP algorithms for the parallel space–time solution of Time Dependent PDEs

Ronald D. Haynes[1] and Benjamin W. Ong[2]

1 Introduction

Modern high performance computers offer hundreds of thousands of processors that can be leveraged, in parallel, to compute numerical solutions to time dependent partial differential equations (PDEs). For grid-based solutions to these PDEs, domain decomposition (DD) is often employed to add spatial parallelism [19].

Parallelism in the time variable is more difficult to exploit due to the inherent causality. Recently, researchers have explored this issue as a means to improve the scalability of existing parallel spatial solvers applied to time dependent problems. There are several general approaches to combine temporal parallelism with spatial parallelism. Waveform relaxation [15] is an example of a "parallel across the problem" method. The "parallel across the time domain" approaches include the parareal method [11, 17, 16]. The parareal method decomposes a time domain into smaller temporal subdomains and alternates between applying a coarse (relatively fast) sequential solver to compute an approximate (not very accurate) solution, and applying a fine (expensive) solver on each temporal subdomain in parallel. Alternatively, one can consider "parallel across the step" methods. Examples of such approaches include the computation of intermediate Runge–Kutta stage values in parallel [18], and Revisionist Integral Deferred Correction (RIDC) methods, which are the family of parallel time integrators considered in this paper. Parallel across the step methods allow for "small scale" parallelism in time. Specifically, we will show that if a DD implementation scales to N_x processors, a RIDC-DD parallelism will scale to $N_t \times N_x$ processors, where $N_t < 12$ in practice. This contrasts with parallel across the time domain approaches, which can potentially utilize $N_t \gg 12$.

This paper discusses the implementation details and profiling results of the parallel space–time RIDC-DD algorithm described in [5]. Two hybrid OpenMP – MPI frameworks are discussed: (i) a more traditional fork-join approach of combining threads before doing MPI communications, and (ii) a threaded MPI communications framework. The latter framework is highly desirable because existing (spatially parallel) legacy software can be easily integrated with the parallel time integrator. Numerical experiments measure the communication overhead of both frameworks, and demonstrate that the fork-join approach scales well in space and time. Our results indicate that one should strongly consider temporal parallelization for the solution of time dependent PDEs.

[1] Memorial University of Newfoundland, St. John's, Newfoundland, Canada e-mail: rhaynes@mun.ca ·[2] Michigan State University, Institute for Cyber-Enabled Research, East Lansing, MI, USA e-mail: ongbw@msu.edu

2 Review

This paper is interested in parallel space-time solutions to the linear heat equation. We describe the application of our method to the linear heat equation in one spatial dimension $x \in [0,1]$ and $t \in [0,T]$,

$$u_t = u_{xx},\, u(t,0) = g_0(t),\, u(t,1) = g_1(t),\, u(0,x) = u_0(x). \tag{1}$$

The actual numerical results in §4 are presented for the 2D heat equation.

2.1 RIDC

RIDC methods [6, 7] are a family of parallel time integrators that can be broadly classified as predictor corrector algorithms [10, 2]. The basic idea is to simultaneously compute solutions to the PDE of interest and associated error PDEs using a low-order time integrator. We first review the derivation of the error equation.

Suppose $v(t,x)$ is an approximate solution to (1), and $u(t,x)$ is the (unknown) exact solution. The error in the approximate solution is $e(t,x) = u(t,x) - v(t,x)$. We define the residual as $\varepsilon(t,x) = v_t(t,x) - v_{xx}(t,x)$. Then the time derivative of the error satisfies $e_t = u_t - v_t = u_{xx} - (v_{xx} + \varepsilon)$. The integral form of the error equation,

$$\left[e + \int_0^t \varepsilon(\tau,x)\, d\tau \right]_t = (v+e)_{xx} - v_{xx}, \tag{2}$$

can then be solved for $e(t,x)$ using the initial condition $e(0,x) = 0$. The correction $e(t,x)$ is combined with the approximate solution $v(t,x)$ to form an improved solution. This improved solution can be fed back in to the error equation (2) and the process repeated until a sufficiently accurate solution is obtained. It has been shown that each application of the error equation improves the order of the overall method, provided the integral is approximated with sufficient accuracy using quadrature [8].

We introduce some notation to identify the sequence of corrected approximations. Denote $v^{[p]}(t,x)$ as the approximate solution which has error $e^{[p]}(t,x)$, which is obtained by solving

$$\left[e^{[p]} + \int_0^t \varepsilon^{[p]}(\tau,x)\, d\tau \right]_t = (v^{[p]} + e^{[p]})_{xx} - v^{[p]}_{xx}, \tag{3}$$

where $v^{[0]}(t,x)$ denotes the initial approximate solution obtained by solving the physical PDE (1) using a low-order integrator. In general, the error from the pth correction equation is used to construct the $(p+1)st$ approximation, $v^{[p+1]}(t,x) = v^{[p]}(t,x) + e^{[p]}(t,x)$. Hence, equation (3) can be expressed as

$$\left[v^{[p+1]} - \int_0^t v^{[p]}_{xx}(\tau,x)\, d\tau \right]_t = v^{[p+1]}_{xx} - v^{[p]}_{xx}. \tag{4}$$

We compute a low-order prediction, $v^{[0],n+1}$, for the solution of (1) at time t_{n+1} using a first-order backward Euler discretization (in time):

$$v^{[0],n+1} - \Delta t\, v_{xx}^{[0],n+1} = v^{[0],n},\ v^{[0],n+1}(a) = g_0(t^{n+1}),\ v^{[0],n+1}(b) = g_1(t^{n+1}), \quad (5)$$

with $v^{[0],0}(x) = u_0(x)$. With some algebra, a first-order backward Euler discretization of equation (4) gives the update, $v^{[p+1],n+1}$, as

$$v^{[p+1],n+1} - \Delta t\, v_{xx}^{[p+1],n+1} = v^{[p+1],n} - \Delta t\, v_{xx}^{[p],n+1} + \int_{t^n}^{t_{n+1}} v_{xx}^{[p]}(\tau, x)\, d\tau, \quad (6)$$

with $v^{[p+1],n+1}(a) = g_0(t^{n+1})$ and $v^{[p+1],n+1}(b) = g_1(t^{n+1})$. The integral in equation (6) is approximated using a sufficiently high-order quadrature rule [8].

Parallelism in time is possible because the PDE of interest (1) and the error PDEs (4) can be solved simultaneously, after initial startup costs. The idea is to fill out the memory footprint, which is needed so that the integral in equation (6) can be approximated by high-order quadrature, before marching solutions to (5) and (6) in a pipe–line fashion. See Figure 1 for a graphical example, and [6] for more details.

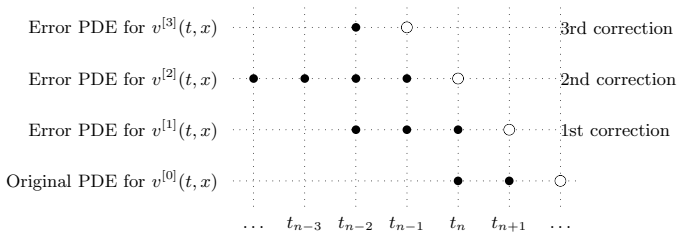


Fig. 1 The black dots represent the memory footprint that must be stored before the white dots can be computed in a pipe. In this figure, $v^{[0],n+2}(x)$, $v^{[1],n+1}(x)$, $v^{[2],n}(x)$ and $v^{[3],n-1}(x)$ are computed simultaneously.

2.2 RIDC–DD

The RIDC–DD algorithm solves the predictor (5) and corrections (6) using DD algorithms in space. The key observation is that (5) and (6) are **both** elliptic PDEs of the form $(1 - \Delta t\, \partial_{xx})z = f(x)$. The function $f(x)$ is known from the solution at the previous time step and previously computed lower-order approximations. DD algorithms for solving elliptic PDEs are well known [3, 4]. The general idea is to replace the PDE by a coupled system of PDEs over some partitioning of the spatial domain using overlapping or non–overlapping subdomains. The coupling is provided by necessary transmission conditions at the subdomain boundaries. These transmission conditions are chosen to ensure the DD algorithm converges and to optimize the con-

vergence rate. In [5], as a proof of principle, (5-6) are solved using a classical parallel Schwarz algorithm, with overlapping subdomains and Dirichlet transmission conditions. Optimized RIDC–DD variants are possible using an optimized Schwarz DD method [13, 12, 9], to solve (5-6). The solution from the previous time step can be used as initial subdomain solutions at the interfaces. We will use RIDCp–DD to refer to a pth-order solution obtained using $p-1$ RIDC corrections in time and DD in space.

3 Implementation Details

We view the parallel time integrator reviewed in §2.1 as a simple yet powerful tool to add further scalability to a legacy MPI or modern MPI–CUDA code, while improving the accuracy of numerical solution. The RIDC integrators benefit from access to shared memory because solving the correction PDE (6) requires both the solution from the previous time step and previously computed lower-order subdomain solution. Consequently, we propose two MPI-OpenMP hybrid implementations which map well to multi-core, multi-node compute resources. In the upcoming MPI 3.0 standard [1], shared memory access within the MPI library will provide alternative implementations.

Implementation #1: The RIDC-DD algorithm can be implemented using a traditional fork join approach, as illustrated in Program 1. After boundary information is exchanged, each MPI task spawns OpenMP threads to perform the linear solve. The threads are merged back together before MPI communication is used to check for convergence. The drawback to this fork-join implementation, is that the parallel space-time algorithm becomes tightly integrated, making it difficult to leverage an existing spatially parallel DD implementation.

```
1.  MPI Initialization
2.  ...
3.    for each time step
4.       for each Schwarz iteration
5.          MPI Comm (exchange boundary info)
6.             OMP Parallel for each prediction/correction
7.                linear solve
8.             end parallel
9.          MPI Comm (check for convergence)
10.      end
11.   end
12. ...
13. MPI Finalize
```

Program 1: RIDC-DD implementation using a fork-join approach. The time parallelism occurs *within* each Schwarz iteration, requiring a tight integration with an existing spatially parallel DD implementation.

Implementation #2: To leverage an existing spatially parallel DD implementation, a non-traditional hybrid approach must be considered. By changing the order of the loops, the Schwarz iterations for the prediction and the correction loops can be evaluated independently of each other. This is realized by spawning individual OpenMP threads to solve the prediction and correction loops on each subdomain; the Schwarz iterations for the prediction/correction step run independently of each other until convergence. This implementation (Program 2) has several consequences: (i) a thread safe version of MPI supporting `MPI_THREAD_MULTIPLE` is required. (ii) In addition, we required a thread-safe, thread-independent version of `MPI_BARRIER`, `MPI_BROADCAST` and `MPI_GATHER`. To achieve this, we wrote our own wrapper library using the thread safe `MPI_SEND`, `MPI_RECV` and `MPI_SENDRECV` provided by (i).

```
1.  MPI Initialization
2.  ...
3.    for each time step
4.       OMP Parallel for each prediction/correction level
5.          for each Schwarz iteration
6.             MPI Comm (exchange boundary info)
7.             linear solve
8.             MPI Comm (check for convergence)
9.          end
10.      end parallel
11.   end
12. ...
13. MPI Finalize
```

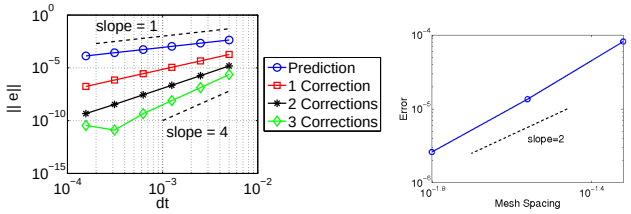
Program 2: RIDC-DD implementation using a non-traditional hybrid approach. Notice that lines 5-9 are the Schwarz iterations that one would find in an existing spatially parallel DD implementation. Hence, provided the DD implementation is thread-safe, one could wrap the time parallelization around an existing parallel DD implementation.

4 Numerical Experiments

We show first that RIDC-DD methods converge with the designed orders in space and time. Then, we profile communication costs using TAU [14]. Finally, we show strong scaling studies for the RIDC-DD algorithm. We compute solutions to the heat equation in $\mathbb{R}^2$, where centered finite differences are used to approximate the second derivative operator. Errors are computed using the known analytic solution. The computations are performed at the High Performance Computing Cenrer at Michigan State University, where nodes (consisting of two quad core Intel Westmere processors) are interconnected using infiniband and a high speed Lustre file system.

4.1 Convergence Studies and Profile Analysis

In Figure 2, the convergence plots show that our classical Schwarz RIDC-DD algorithm converges as expected in space and time. In general, one would balance the orders of the errors in space and time appropriately for efficiency. Here we pick RIDC4 since it mapped well to our available four core sockets and to demonstrate the scalability of our algorithm in time. We could, of course, used a fourth order method in space. The Schwarz iterations are iterated until a tolerance of 10^{-12} is reached for the predictors and correctors (which explains why the error in the fourth-order approximation levels out as the time step becomes small).



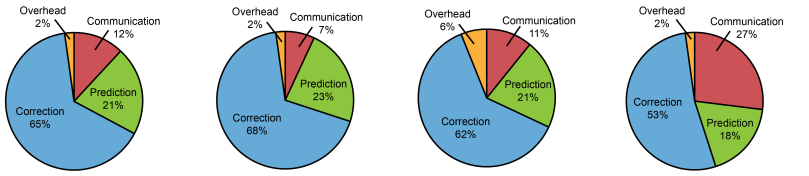
(a) Time Convergence (b) Space Convergence

Fig. 2 (a) Classical Schwarz RIDCp-DD algorithms, $p = 1,2,3,4$, converge to the reference solution with the designed orders of accuracy. Here Δx is fixed while Δt is varied. (b) Second-order convergence in space is demonstrated for the fourth-order RIDC-DD algorithm. Here, Δt is fixed while Δx is varied.

The communication costs for our two implementations of RIDC4-DD are profiled using TAU [14]. We see in Figure 3, communication costs are minimal for implementation #1, and scales nicely as the number of nodes is increased, but the communication cost is significant for implementation #2. In Figure 3(a,c), the domain is discretized into 180×180 grid nodes, which are split into a 3×3 configuration of subdomains. In Figure 3(b,d), the domain is discretized into 360×360 grid nodes, which are split into a 6×6 configuration of subdomains. This keeps the number of grid points per subdomain constant so that the computation time for the matrix factorization and linear solve are the same.

4.2 Characterizing Parallel Performance

Due to the better communication profile, we use framework #1 for our experiments. We fix $\Delta x = \frac{1}{180}$, $\Delta y = \frac{1}{180}$, $\Delta t = \frac{1}{1000}$, and TOL=$10^{-12}$ (the Schwarz iteration tolerance). We consider three configurations of subdomains: 2×2, 4×4 and 6×6. For each configuration we illustrate the speedup and efficiency due to the time parallelism in Figure 4. We choose to fix the ratio between the overlap and subdomain



(a) Implementation #1 3×3 domain (b) Implementation #1 6×6 domain (c) Implementation #2 3×3 domain (d) Implementation #2 6×6 domain

Fig. 3 Profile of the RIDC4-DD algorithm using both implementations. Overhead and communication costs are reasonable for implementation #1, but are high for implementation #2.

size to ensure the number of unknowns on each subdomain scales appropriately as the number of subdomains is increased.

In Figure 4 we show three curves corresponding to a 2×2, 4×4 and a 6×6 configuration of subdomains. For each configuration we compute a fourth order solution in time using $1, 2$ and 4 threads. The 6×6 configuration of subdomains with 4 threads uses a total of 144 cores. We plot the efficiency (with respect to the one thread run) as a function of the number of threads. Speedup is evident as temporal parallelization is improved, however, efficiency decreases as the number of subdomains increases.

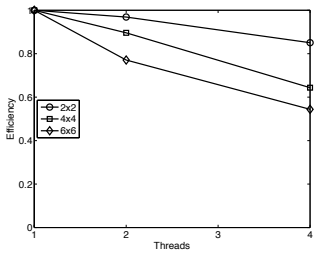


Fig. 4 Scaling study (in time) for a RIDC4-DD algorithm.

5 Conclusions

This paper has presented the implementation details and first reported profiling results for a newly proposed space–time parallel algorithm for time dependent PDEs. The RIDC–DD method combines traditional domain decomposition in space with a new family of deferred correction methods designed to allow parallelism in time. Two possible implementations are described and profiled. The first, a traditional hybrid OpenMP–MPI implementation, requires potentially difficult modifications of an existing parallel spatial solver. Numerical experiments verify that the algorithm achieves its designed order of accuracy and scales well. The second strategy allows a relatively easy reuse of an existing parallel spatial solver by using OpenMP to spawn threads for the simultaneous prediction and correction steps. This non–traditional hybrid use of OpenMP and MPI currently requires writing of custom thread–safe and thread–independent MPI routines. Profile analysis shows that our non-traditional use of OpenMP–MPI suffers from higher communication costs than the standard use of OpenMP-MPI. An inspection of the prediction and correction

equations indicates that optimized Schwarz variants of the algorithm are possible and will enjoy nice load balancing. This work is ongoing.

Acknowledgements This work was supported by the Institute for Cyber-Enabled Research (iCER) at MSU, NSERC Discovery Grant 311796, and AFOSR Grant FA9550-12-1-0455.

References

1. Mpi 3.0 standardization effort. `http://meetings.mpi-forum.org/MPI_3.0_main_page.php`. Accessed 10/25/2012
2. Böhmer, K., Stetter, H.: Defect correction methods. Theory and applications. Computing Supplementum, 5 (1984)
3. Cai, X.C.: Additive Schwarz algorithms for parabolic convection-diffusion equations. Numer. Math. **60**(1), 41–61 (1991)
4. Cai, X.C.: Multiplicative Schwarz methods for parabolic problems. SIAM J. Sci. Comput. **15**(3), 587–603 (1994)
5. Christlieb, A., Haynes, R., Ong, B.: A parallel space-time algorithm. SIAM J. Sci. Comput. **34**(5), 233–248 (2012)
6. Christlieb, A., Macdonald, C., Ong, B.: Parallel high-order integrators. SIAM J. Sci. Comput. **32**(2), 818–835 (2010)
7. Christlieb, A., Ong, B.: Implicit parallel time integrators. J. Sci. Comput. **49**(2), 167–179 (2011)
8. Christlieb, A., Ong, B., Qiu, J.M.: Comments on high order integrators embedded within integral deferred correction methods. Comm. Appl. Math. Comput. Sci. **4**(1), 27–56 (2009)
9. Dubois, O., Gander, M., Loisel, S., St-Cyr, A., Szyld, D.: The optimized Schwarz method with a coarse grid correction. SIAM J. Sci. Comput. **34**(1), A421–A458 (2012)
10. Dutt, A., Greengard, L., Rokhlin, V.: Spectral deferred correction methods for ordinary differential equations. BIT **40**(2), 241–266 (2000)
11. Gander, M., Vandewalle, S.: On the superlinear and linear convergence of the parareal algorithm. Lecture Notes in Computational Science and Engineering **55**, 291 (2007)
12. Gander, M.J.: Optimized Schwarz methods. SIAM J. Numer. Anal. **44**(2), 699–731 (2006)
13. Gander, M.J., Halpern, L.: Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. SIAM J. Numer. Anal. **45**(2), 666–697 (2007)
14. Koehler, S., Curreri, J., George, A.: Performance analysis challenges and framework for high-performance reconfigurable computing. Parallel Computing **34**(4), 217–230 (2008)
15. Lelarasmee, E., Ruehli, A.E., Sangiovanni-Vincentelli, A.L.: The waveform relaxation method for time–domain analysis of large scale integrated circuits. IEEE Trans. on CAD of IC and Syst. **1**, 131–145 (1982)
16. Lions, J., Maday, Y., Turinici, G.: A "parareal" in time discretization of PDEs. Comptes Rendus de l'Academie des Sciences Series I Mathematics **332**(7), 661–668 (2001)
17. Minion, M., Williams, S.: Parareal and spectral deferred corrections. In: NUMERICAL ANALYSIS AND APPLIED MATHEMATICS: International Conference on Numerical Analysis and Applied Mathematics 2008. AIP Conference Proceedings, vol. 1048, pp. 388–391 (2008)
18. Nievergelt, J.: Parallel methods for integrating ordinary differential equations. Communications of the ACM **7**(12), 731–733 (1964)
19. Toselli, A., Widlund, O.: Domain decomposition methods—algorithms and theory, *Springer Series in Computational Mathematics*, vol. 34. Springer-Verlag, Berlin (2005)

1.

1. Report Type

Final Report

Primary Contact E-mail

Contact email if there is a problem with the report.

ongbw@mtu.edu

Primary Contact Phone Number

Contact phone number if there is a problem with the report

906-487-3367

Organization / Institution name

Michigan State University

Grant/Contract Title

The full title of the funded effort.

Fault Tolerant Paradigms

Grant/Contract Number

AFOSR assigned control number. It must begin with "FA9550" or "F49620" or "FA2386".

FA9550-12-1-0455

Principal Investigator Name

The full name of the principal investigator on the grant or contract.

Benjamin W. Ong

Program Manager

The AFOSR Program Manager currently assigned to the award

Jean Luc Cambier

Reporting Period Start Date

11/30/2012

Reporting Period End Date

11/29/2015

Abstract

This project had three principle aims:
1. Improving the scalability and efficiency of ``Ultra-scale" methods for grid-based solutions to time-dependent PDEs;
2. Sparse storage and reconstruction of information;
3. Build-in several levels of resiliencies to handle various hard faults in the system.

Progress was made in all three areas, leading to fifteen published refereed articles, five articles in review, one completed masters thesis, and one doctoral thesis in progress.

Distribution Statement

This is block 12 on the SF298 form.

Distribution A - Approved for Public Release

Explanation for Distribution Statement

If this is not approved for public release, please provide a short explanation. E.g., contains proprietary information.

SF298 Form

Please attach your SF298 form. A blank SF298 can be found here. Please do not password protect or secure the PDF

The maximum file size for an SF298 is 50MB.

CompletedSF298.pdf

Upload the Report Document. File must be a PDF. Please do not password protect or secure the PDF . The maximum file size for the Report Document is 50MB.

report-publication.pdf

Upload a Report Document, if any. The maximum file size for the Report Document is 50MB.

Archival Publications (published) during reporting period:

Changes in research objectives (if any):

Change in AFOSR Program Manager, if any:

Extensions granted or milestones slipped, if any:

AFOSR LRIR Number

LRIR Title

Reporting Period

Laboratory Task Manager

Program Officer

Research Objectives

Technical Summary

Funding Summary by Cost Category (by FY, $K)

	Starting FY	FY+1	FY+2
Salary			
Equipment/Facilities			
Supplies			
Total			

Report Document

Report Document - Text Analysis

Report Document - Text Analysis

Appendix Documents

2. Thank You

E-mail user

Feb 09, 2016 15:47:44 Success: Email Sent to: ongbw@mtu.edu