

PAPER • OPEN ACCESS

Programmable quantum simulation by dynamic Hamiltonian engineering

To cite this article: David Hayes *et al* 2014 *New J. Phys.* **16** 083027

View the [article online](#) for updates and enhancements.

Related content

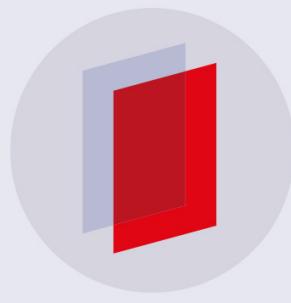
- [Effects of gate errors in digital quantum simulations of fermionic systems](#)
Jan-Michael Reiner, Sebastian Zanker, Iris Schwenk *et al.*

- [Quantum-circuit design for efficient simulations of many-body quantum dynamics](#)
Sadegh Raeisi, Nathan Wiebe and Barry C Sanders

- [Hamiltonian quantum simulation with bounded-strength controls](#)
Adam D Bookatz, Paweł Wocjan and Lorenza Viola

Recent citations

- [Decoherence Control of Nitrogen-Vacancy Centers](#)
Chao Lei *et al*



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

New Journal of Physics

The open access journal at the forefront of physics

Deutsche Physikalische Gesellschaft  | IOP Institute of Physics

Programmable quantum simulation by dynamic Hamiltonian engineering

David Hayes^{1,2}, Steven T Flammia¹ and Michael J Biercuk^{1,2}

¹ ARC Centre for Engineered Quantum Systems, School of Physics, The University of Sydney, NSW 2006, Australia

² National Measurement Institute, Bradfield Road, West Lindfield, NSW 2070, Australia
E-mail: michael.biercuk@sydney.edu.au

Received 18 March 2014, revised 17 June 2014

Accepted for publication 27 June 2014

Published 12 August 2014

New Journal of Physics **16** (2014) 083027

[doi:10.1088/1367-2630/16/8/083027](https://doi.org/10.1088/1367-2630/16/8/083027)

Abstract

Quantum simulation is a promising near term application for quantum information processors with the potential to solve computationally intractable problems using just a few dozen interacting qubits. A range of experimental platforms have recently demonstrated the basic functionality of quantum simulation applied to quantum magnetism, quantum phase transitions and relativistic quantum mechanics. However, in all cases, the physics of the underlying hardware restricts the achievable inter-particle interactions and forms a serious constraint on the versatility of the simulators. To broaden the scope of these analog devices, we develop a suite of pulse sequences that permit a user to efficiently realize average Hamiltonians that are beyond the native interactions of the system. Specifically, this approach permits the generation of all symmetrically coupled translation-invariant two-body Hamiltonians with homogeneous on-site terms, a class which includes all spin-1/2 XYZ chains, but generalized to include *long-range* couplings. Our work builds on previous work proving that universal simulation is possible using both entangling gates and single-qubit unitaries. We show that determining the appropriate ‘program’ of unitary pulse sequences which implements an arbitrary Hamiltonian transformation can be formulated as a linear program over functions defined by those pulse sequences, running in polynomial time and scaling efficiently in hardware resources. Our analysis extends from circuit model quantum information to



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](#). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

adiabatic quantum evolutions, representing an important and broad-based success in applying functional analysis to the field of quantum information.

Keywords: quantum simulation, quantum information, dynamical mapping, functional analysis

1. Introduction

Universal computation is derived from the ability to reduce complex and arbitrary calculations to a finite set of physical operations. While generic digital logic is well suited for many applications, *analog* simulation can provide much more efficient methods for performing certain types of computations. In this approach, computational operations of interest can be efficiently built into the underlying hardware, for instance, as analog circuits. Such approaches can greatly enhance performance at the cost of guaranteed error tolerance [1].

The same capabilities exist in the quantum regime, with the promise to realize extraordinary computational capabilities for a variety of challenging problems of interest. Analog quantum simulation may provide a near-term path to providing solutions to currently intractable computations at the scale of just a few dozen interacting quantum systems [2, 3, 4]. Quantum simulation leverages well-controlled quantum coherent devices in order to study the properties of poorly understood many-body systems: e.g. interacting spins and quantum magnetism. In this vein, a wide variety of experimental demonstrations have been realized to date [4–12] using precisely controlled atomic systems.

The efficiency benefits associated with analog simulators are partially offset by the challenging open question as to how one may efficiently implement a ‘program’ therein using a constrained set of basic resources, even when proofs of universality exist [13–15]. This is apparent in the case of experimental simulation of quantum magnetism; all experiments to date have been severely constrained by the native interactions achievable between the underlying quantum bits. Attempts to expand functionality have typically focused on exploiting the specific properties of a given hardware platform [16, 17] (e.g. varying the coupling of spins to different normal modes of motion in a trapped-ion simulator), rather than providing a generic, technology-independent approach. Producing a general, efficient and extensible framework for programming complex interactions and broadening the range of accessible simulations with limited hardware capabilities [18, 19] is thus a key requirement to expand the utility of mesoscale quantum simulation.

Here we describe a technology-independent technique to develop a hybrid quantum spin simulator with constrained resources, capable of efficiently realizing a broad class of interacting spin Hamiltonians. Beginning with a native long-range pairwise interaction between spins on a one-dimensional chain, the application of short sequences of single-spin Pauli operators can effectively generate any other Hamiltonian in the family by filtering the time-averaged relative weight of different pairwise interactions. The approach we describe frames the problem as one of functional synthesis where basis functions are generated by the pulse sequences that we define. This general methodology has met success in a variety of contexts for quantum information, including the design of algorithms for quantum memory [20, 21], robust-control techniques [22, 23], circuit-model algorithm synthesis [24], and even applications in quantum-enabled sensing [25].

Through sequential and concatenated application of the pulse sequences we define, the basis functions can be added and multiplied together and are shown to generate a complete, orthogonal set of functions that span the full space of Hamiltonians in the family. The challenge of determining the appropriate operations required to map the native interaction to any other is reduced to an efficiently soluble linear program, and moreover this solution is (in a certain precise sense) optimal. In addition to providing specific examples of useful filtering tasks, we show how this technique is extensible to adiabatic quantum simulation and permits the preparation of states that are otherwise inaccessible.

Specifically, we consider the family of one-dimensional Hamiltonians with translation-invariant symmetrically coupled two-body terms (i.e. XX-, YY-, and ZZ-type couplings) extended to include long-range couplings, and homogeneous transverse on-site terms. More precisely, we model a finite N -qubit chain on a one-dimensional lattice (figure 1(a)) with Hamiltonian $H = H_I + H_T$. Here H_I is a fully connected interaction Hamiltonian

$$H_I = \sum_{d=1}^{N-1} \Omega_d \hat{h}_d = \sum_{d=1}^{N-1} \Omega_d \cdot \sum_{j=1}^{N-d-1} \mathbf{S}_j \mathbf{S}_{j+d}, \quad (1)$$

where $\Omega_d = (\Omega_d^{(x)}, \Omega_d^{(y)}, \Omega_d^{(z)})$ and $\mathbf{S}_j \mathbf{S}_{j+d} = (X_j X_{j+d}, Y_j Y_{j+d}, Z_j Z_{j+d})$ represent the interaction strength and coupling in each Cartesian direction of two qubits separated by d lattice spacings. The dot product between these implements a generally anisotropic coupling. We also incorporate a homogeneous transverse field $H_T = B \sum_j Z_j$, where the magnitude of B is controllable.

Our first main result is that, starting with any such family with polynomially decaying long-range couplings Ω_d , there exists a short sequence of single-qubit Z pulses such that the time-averaged Hamiltonian is equal to *any other Hamiltonian in the class*, up to rescaling factor. We can quantify this statement as follows. The sequence of Z pulses contains at most $O(\log N)$ concatenation steps, requiring a pulse number of at most $O(N^2)$ with at most $O(N^3)$ individual spin flips, and the rescaling factor for the filtered interaction strength is at most $N^{-2.585}$, the exponent given by $-\log_2 6$. Thus the scheme can be fully implemented in polynomial time. Moreover, the *optimal* rescaling factor for our scheme can be obtained by efficiently solving a linear program. We also provide explicit demonstrations of our method for implementing power-law decay modulation in Ising spin chains and adiabatic state preparation, as well as explicit bounds on simulation error.

The remainder of this paper is organized as follows. We describe our general approach to Hamiltonian engineering in section 2, introducing an extensible family of pulse sequences that allow us to generate any Ising-type coupling (in the above sense). Through combination of filters we demonstrate how these pulse sequences are in fact universal, allowing arbitrary mappings within the Hamiltonian class with efficient resource scaling in section 3, with additional proof details in [appendix](#). We then show how filters may be efficiently compiled to achieve a desired ‘program’ for a quantum simulator in section 4, followed by examples of what can be achieved with the method in section 5. We then show how this can all be generalized to adiabatic evolutions (section 6) and consider the fully general Hamiltonians of equation (1) in section 7, including rigorous bounds on the simulation error. Our work concludes with a discussion of open questions in the literature and a summary of our results.

2. Hamiltonian engineering

2.1. General concepts

Owing its foundations to research in NMR [5, 26], the idea of altering the dynamics of interacting quantum systems has proven invaluable in the field of quantum control. For instance, in the context of dynamic error suppression [27–30] a quantum system may be effectively isolated from its environment, not by eliminating the physical interaction, but by inducing a dynamical response which acts to average out the coupling to the environment. This picture extends to interactions between quantum coherent systems as in spin decoupling and recoupling in NMR [5, 18, 19, 31] and have been used to provide universality proofs of quantum simulators using a set of operations similar to universal quantum computation [13–15].

The simplest example of this type of Hamiltonian engineering is the dynamic decoupling of two spins interacting via the Hamiltonian $H = \Omega X_1 X_2$, where X_j is the Pauli x operator for the j th particle. The application of phase-flip pulses (Z_j) at time intervals Δt produces time-dependent operators $X_j(t) = w_j(t)X_j$ in the Heisenberg picture. The control propagator $w_j(t) = \pm 1$ captures the discrete-time modulation with the sign changes occurring at the times of the applied pulses and time evolution operator is,

$$U(\Delta t) = \exp [-i\Omega_{\text{eff}} X_1 X_2 \Delta t], \quad (2)$$

with $\Omega_{\text{eff}} = \Omega \mathbf{w}_1 \cdot \mathbf{w}_2$ and $[\mathbf{w}_j]_l = w_j(l\Delta t)$. We thus see that the inner product of the discretized control propagators \mathbf{w}_j determines the effective coupling.

It has been demonstrated that the physics underlying this simple example can be generalized for application to a collection of qubits in order to realize arbitrary effective two-qubit interactions [18]. Unfortunately existing approaches entail tremendous complexity when employed for Hamiltonian engineering in quantum simulations. Our goal is to produce a simple extensible set of pulse-sequence constructions suitable for the task of quantum simulation. Our method takes advantage of the symmetry of a ubiquitous architecture in physical quantum simulators, the translational invariant one-dimensional spin chain with long-range coupling.

2.2. Filter construction

We now show how to build on this very simple example and define a suite of pulse sequences that may be used to efficiently program an effective long-range pairwise spin-coupling Hamiltonian in a multi-qubit system. Consider a Hamiltonian of the form in equation (1); this form of long-range spin interaction is extremely important for quantum-many-body physics studies of exotic forms of quantum magnetism and is commonly realized e.g. in trapped-ion spin simulators and optical lattice simulators [6, 32–35].

We begin by treating the special case of Ising-type interactions,

$$H_I \rightarrow \sum_{d=1}^{N-1} \Omega_d \hat{h}_d = \sum_{d=1}^{N-1} \Omega_d \sum_{j=1}^{N-d-1} X_j X_{j+d}, \quad (3)$$

and determine the relevant pulse sequences that permit our desired transformations on the interqubit couplings to be realized exactly. Later we will revisit the generalized anisotropic Heisenberg-type coupling with homogeneous on-site terms and discuss what modifications must be made in order to effectively treat the more general model. We look to craft a set of

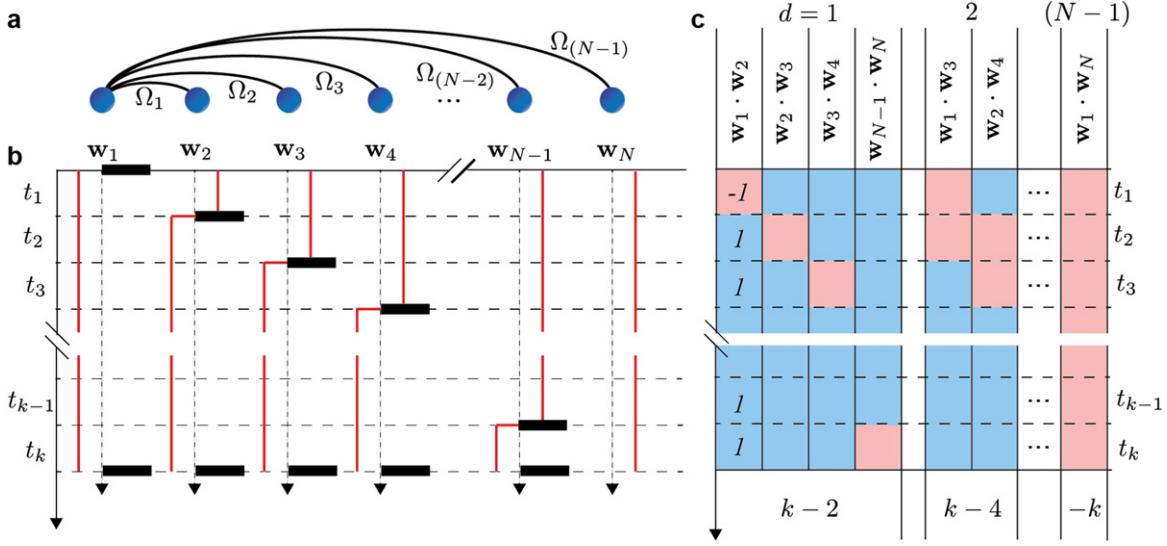


Figure 1. Illustration of translationally invariant dynamical filter generation for a linear chain of coupled qubits. (a) Schematic depiction of qubits on a linear chain with distance-dependent coupling. For an N -qubit chain the largest distance is $(N - 1)d$, where d is the primitive lattice constant. Only interactions between the qubit 1 and others in the chain are presented for clarity. (b) Pulse sequence and associated control propagators, $\{w_j^{(k)}\}$, for the N qubits. Time runs from top to bottom in k discrete steps. Black bars indicate π_Z pulses, causing a sign change in the associated control propagator. The particular map depicted generates $f_{k=1}^A(d)$. (c) Graphical representation of the vector dot-products, $w_j^{(k)} \cdot w_{j'}^{(k)}$ between spins entering into the time evolution operator. Color indicates the sign of the interaction in a discrete time bin. The sum over time bins gives the resulting prefactor in the effective Hamiltonian. All pairs of spins separated by distance d have a resulting overlap $k - 1$, all pairs separated by $2d$ have overlap $k - 2$, decreasing linearly with d until the qubits separated by the largest distance have overlap $-k$, indicating a sign change.

pulse sequences that permit the realization of an arbitrary effective Hamiltonian within this class by modifying the native interaction so that $\Omega_d \rightarrow \Omega_{df}(d)$. In this approach both the original and effective Hamiltonians are translationally invariant.

To simulate the time-evolution under a Hamiltonian for a time T , we break T into k segments of equal duration $\Delta t_k = T/k$, with pulses applied as required at the transitions between time segments [14]. This results in a family of control propagators $\{w_j^{(k)}\}$ labelled by k with j indexing the qubits. We define operators $\hat{P}_{k,l}$ that are applied both at the beginning and end of the l th time segment.³ Each is a product of Z_j operators on some subset of the N different qubits in the chain. The result of applying these pulse sequences is a modification of the time evolution during the l th time segment to $U_{k,l}(\Delta t_k) = \hat{P}_{k,l} \exp[-iH\Delta t_k] \hat{P}_{k,l}$. The $\hat{P}_{k,l}$ operators generate sign changes in the control propagators of individual qubits and the sign of $\hat{P}_{k,l} X_j \hat{P}_{k,l}$ is encoded by the l th element of $w_j^{(k)} \equiv [w_j^{(k)}]_l$ (figures 1(b)–(c)). The effective time-evolution is related to the individual control propagators as $U_{k,l}(\Delta t_k) = \exp[-iH_{k,l}^{\text{eff}} \Delta t_k]$ where $H_{k,l}^{\text{eff}} = \sum_d \Omega_d \sum_j [w_j^{(k)}]_l [w_{j+d}^{(k)}]_l X_j X_{j+d}$. The total evolution operator is given by the product

³ Thus at the boundary of time segments l and $l + 1$ we actually apply the operator $\hat{P}_k \hat{P}_{k+1}$. Defining $\hat{P}_{k,l}$ in this way greatly simplifies the analysis.

over all time periods as

$$U_k(T) = \prod_{l=1}^k U_{k,l}(\Delta t_k) = \exp \left[-i H_k^{\text{eff}} T \right], \quad (4)$$

$$H_k^{\text{eff}} = \sum_l H_{k,l}^{\text{eff}} \equiv \sum_d \Omega_d f_k(d) \hat{h}_d. \quad (5)$$

The dynamically modified coupling is represented through the function $f_k(d)$, which we define to be the vector dot product of the control propagators $\mathbf{w}_j^{(k)} \cdot \mathbf{w}_{j+d}^{(k)}$ over the discrete time segments.

With this framework we introduce the primary class of pulse sequence we construct, Λ_k , defined by the pulse operators

$$\left[\hat{P}_{k,j} \right]_{j=1}^k \quad \hat{P}_{k,j} = \prod_{i=1}^N Z_i^{\left\lfloor \frac{j-i-1}{k} \right\rfloor}. \quad (6)$$

Here, square brackets around the operators $P_{k,j}$ indicate that we are defining a set of operators acting simultaneously on all qubits in the chain. We also denote rounding x down (respectively, up) to the nearest integer by $\lfloor x \rfloor$ (respectively, $\lceil x \rceil$). In this construction the exponent on the i th Pauli Z operator is always an integer so that, depending on the parity, either Z_i or the identity is applied to the i th qubit and j denotes the relevant time bin. These sequences modify Ω_d by the factor

$$f_k^\Lambda(d) := (-1)^{\lfloor \frac{d}{k} \rfloor} \left(1 - 2 \left\{ \frac{d}{k} \right\} \right), \quad (7)$$

where $\{x\}$ denotes the fractional part of x . This filter varies as a triangle wave as a function of qubit distance (as shown in figure 2(a)), with period $2k$ and range $f_k^\Lambda(d) \in [-1, 1]$ which allows for considerable flexibility in the effective coupling. For example, it is possible to switch the sign of the interaction from ferromagnetic to antiferromagnetic, or visa versa, as a function of distance, d .

The second class of filters we introduce is defined as Γ_k :

$$\left[\hat{P}'_{k,j} \right]_{j=1}^k \quad \hat{P}'_{k,j} = \prod_{i=1}^N Z_i^{\left\lceil \left\{ \frac{k+i-j}{k} \right\} \right\rceil}. \quad (8)$$

The filter Γ_k is capable of providing a relative enhancement to the interaction strength of qubits separated by integer multiples of k lattice spacings. The Γ_k map modifies the system's evolution according to

$$f_k^\Gamma(d) = 1 - \frac{4}{k} \left\lceil \left\{ \frac{d}{k} \right\} \right\rceil. \quad (9)$$

Because of the selectivity of the filter's action in providing a periodic (in d) relative enhancement in coupling, we refer to this as a boost filter.

3. A complete set of interactions

The simulation of arbitrary Hamiltonians mandates much more complex functional dependences than simple triangle waves or periodic boosts. Previous work [36] has shown that triangle waves coupled with phase shifts form a complete basis for functional synthesis similar to sines and cosines. Despite being unable to generate phase shifted triangle waves, we show that it is possible to achieve arbitrary multiqubit interactions when we augment positive linear combination of filters with products of filters. Utilizing both operations overcomes the limitations arising from accessing only positive linear combinations and ‘cosine’-like triangle waves.

3.1. Universal filter space

A positive linear combination such as, $\Omega_d \left[T^{(1)}f^{(1)}(d) + T^{(2)}f^{(2)}(d) \right]$, is generated through sequential application of the pulse sequences with the relative weights being determined by the time taken to implement the pulse sequence; cf equation (4). Similarly, products of filters such as $\Omega_d \left[T^{(1)}T^{(2)}f^{(1)}(d)f^{(2)}(d) \right]$ are achieved via sequence concatenation, in which sequences are ‘nested’ within the free evolution periods of one another. These two operations are simple but extremely powerful; with these it is possible to engineer *arbitrary* translationally invariant qubit couplings within the family we consider.

Achieving this among N qubits implies the ability to perform functional synthesis in an $(N - 1)$ -dimensional space, where each dimension represents the coupling between qubits separated by a particular distance, d . A point in this space therefore represents an arbitrary combination of interaction strengths over the relevant values of d . In this space, we define C_N to be the convex set of filters $f_k(d)$ generated by both positive linear combination and concatenation of our basic filters. While this set takes a complex geometric form, we will show that it contains an $(N - 1)$ -dimensional hypercube around the origin, implying that these filters achieve an arbitrary $f(d)$ and hence an arbitrary $\Omega_{\text{eff}}(d)$. We call this cube ‘universal’, in analogy with universal computation, but here restricted to translation-invariant two-body spin Hamiltonians with on-site transverse fields. In figure 2, we show C_3 and C_4 , including the universal set contained therein.

3.2. Efficient resource scaling of the method

Proving access to a universal filter space still leaves open the question of how the actual control operations affect the scaling of the simulator’s performance. We prove that it is possible to efficiently generate an arbitrary desired $\Omega_{\text{eff}}(d)$ using linear combination and concatenation of the primary sets of filters we define. A key result of our method, described in detail in [appendix](#), is that the realization of an arbitrary interaction Hamiltonian within this universal space may be accomplished using at most $O(\log N)$ concatenation steps. These are the most resource-intensive operations we employ, requiring a pulse number of at most $O(N^2)$ with at most $O(N^3)$ individual spin flips. Arriving at this result is a complex exercise, and finding such an efficient resource scaling was not expected at the outset of this work.

Our ability to efficiently construct the universal filter space has significant impact on the *runtime* of filter implementation. The cost of universality is a rescaling of the interaction strength, reflected in the reduced volume of the hypercube relative to the extremal points of the convex set (see figures 2(b)–(c)). This reduced effective interaction strength within the universal

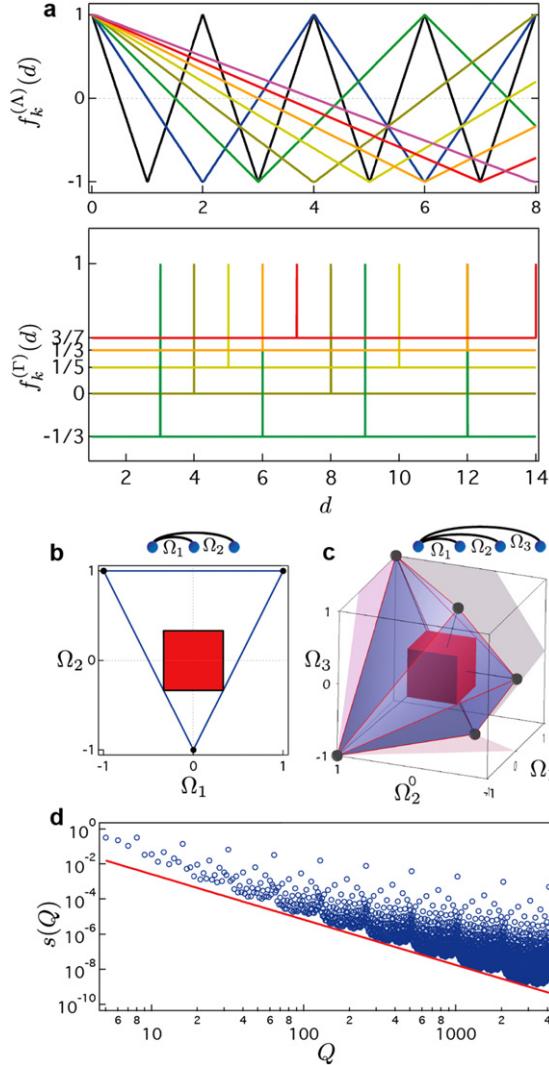


Figure 2. Filters achievable using dynamical mapping pulse sequences. (a) Upper panel: basis of triangle wave functions arising from $f_k^A(d)$. Traces plotted here have $k = 1$ (black, highest frequency) to $k = 8$ (violet, lowest frequency). Lower panel: basis of boost filters arising from $f_k^R(d)$, for $k = 3$ to $k = 6$ using the same color encoding as in the upper panel. (b) A representation of the set C_3 and the universal filter space contained within it, represented by the square confined in size to fit within the convex set defined along axes $\Omega_{d=1}$ and $\Omega_{d=2}$, the only possible values for three spins. Black points represent the extremal values of accessible filters $(1, 1) = \Lambda_0$, $(-1, 1) = \Lambda_1$ and $(0, -1) = \Lambda_2$ in the two dimensions, bounding a convex set denoted by the blue line. Only Λ_0 and Λ_2 are required to construct any other triangle wave where $k \geq 2$, since all higher order waves are a weighted average of these two maps when $d \leq 2$, and concatenation of sequences does not yield new filters: $\Lambda_2^2 = (\Lambda_0 + \Lambda_1)/2$, $\Lambda_1^2 = \Lambda_0$ and $\Lambda_1\Lambda_2 = \Lambda_2$. (c) A representation of the set C_4 and the universal filter space (cube) contained within it. Projections of the complex polytope onto planes in the basis of pairwise qubit coupling are represented using shading. In (b) and (c) schematic representations of possible interqubit couplings (giving axes on the filter space) are presented. (d) Calculated value of $s(Q)$, the strength of the filtered interaction for qubits separated by distance $Q > N/2$. Red line is a guide to the eye scaling as $Q^{-2.585}$.

filter space physically corresponds to an increase in the total evolution time required to achieve a target interaction form.

Naively, requiring $O(\log N)$ layers of concatenation in realizing the universal filter space would result in a bound of $N^{O(\log N)}$, which is only quasi-polynomial. To test this we calculate the quantity $s(Q)$, the ‘worst-case’ strength of the filtered interaction for qubits separated a distance $Q > N/2$, whose inverse sets the upper bound on runtime

$$s(Q) = \prod_{n=0}^m \left(f_{2^n}^\Lambda + \frac{1}{\left(1 - \frac{4}{Q}\right)} \right) \quad (10)$$

(see [appendix](#) for details of the construction of this equation). This expression is derived from the concatenated sequence described by equation [\(A.5\)](#), and it is simply the product of the largest reduction in strength for each of the m concatenations. This is equivalent to assuming the smallest value of the interaction strength reduction for each of the layers of concatenation, thus providing a lower bound on the final interaction strength. We note that we can focus only on the last half of the chain without loss of generality because the filter strength typically decreases as a function of d .

In figure [2\(d\)](#) we numerically evaluate this function for up to $N = 2^{20}$, revealing exotic fractal-like features due to a recursive structure in the definition in terms of the binary digits of Q . More importantly, we find that the special structure of our pulse sequences gives a much stronger runtime upper bound with *polynomial scaling*, than that suggested by the naive expectation. Examining the results, we conjecture that a lower bound for this function is given by $Q^{-2.585}$ (red line, figure [2\(d\)](#)), the exponent given by $-\log_2 6$. This is an exceptional observation, highlighting the practical relevance of this approach.

We may also consider the relevance of control errors in instances where complex concatenated sequences are employed. A chief source among these is the role of non-instantaneous control may be simply bounded by minimizing the ratio of τ_π/T for a given sequence. Since we have demonstrated that the worst-case pulse number required to enact an arbitrary transformation scales polynomially with qubit number, we may therefore also bound the growth of this error. Error-resistant compensating pulses may also be employed, as has been discussed in the literature [\[37, 38\]](#), or recent results assuming bounded-strength controls may be considered [\[39\]](#).

4. Efficient linear program for finding optimal pulse sequences

The existence of the universal filtering space guarantees that *some* pulse sequence exists in order to provide a desired transformation. The process of mapping a desired Hamiltonian for simulation to an appropriate and efficient ‘program’ of filters is accomplished as a problem in *linear programming*. The algorithm presented here actually *finds* a valid pulse sequence (if one exists given the input filters) and minimizes the total amount of time spent applying pulses. Our results show that the problem of compiling pulse sequences can be solved efficiently in N , the number of spins, subject to some mild caveats. Additionally, many fast, practical implementations of linear programming algorithms exist to solve this problem at scales of interest for the implementation of useful quantum simulations in polynomial time.

Consider a set of elementary filter functions f_1, \dots, f_m , which we think of as vectors of dimension $N - 1$. These could come from e.g. the basic Λ and Γ filters defined above, together

with k levels of concatenation for some fixed k . We can also add any other filters so long as we restrict to a polynomial upper bound on m , i.e. $m = O(N^c)$ for some constant c independent of N , in order to ensure that the algorithm below runs in polynomial time. In particular, we can add the $O(N)$ specific ‘basis’ we constructed in the appendix at $k = O(\log N)$ levels of concatenation which we use there to prove the universality of our scheme.

Given this set of dynamic filters, we wish to compile a positive linear combination of pulse sequences that will generate a given Hamiltonian. As mentioned above, this problem can be cast as a *linear program*, which is an efficient method for finding optima of linear objective functions subject to linear equality and inequality constraints [40]. Using standard linear algebra routines, this procedure may be implemented numerically and provides provably optimal solutions in terms of the total evolution time; the runtime of a linear program with n input variables and $\text{poly}(n)$ constraints is polynomial in n .

To see that our problem is a linear program, we make the following observations. Given a desired vector of couplings Ω_{eff} , we wish to find time steps $t_j \geq 0$ such that

$$\sum_{j=1}^m t_j f_j = \frac{\Omega_{\text{eff}}}{\Omega} . \quad (11)$$

Here we actually have a *vector* equation, and the division on the right-hand side is done elementwise. Suppose that each filter f_j has a cost c_j associated to it (for simplicity we can imagine that all of these costs are equal.) Then a linear program which will compile a given pulse sequence to generate the effective coupling Ω_{eff} is

$$\min_t c^T t \quad \text{subject to} \quad \sum_{j=1}^m t_j f_j = \frac{\Omega_{\text{eff}}}{\Omega} , \quad t_j \geq 0 . \quad (12)$$

The form of the program presented above may be converted to the ‘standard’ form for linear programming by adding slack variables with additional equality constraints for each of the components of the vector equation. Finding a solution requires that the selected filter set is chosen to be universal. Regardless, the program will terminate in time polynomial in m and N and will minimize the total amount of evolution time required to implement the desired coupling. Further improvements might be desired, such as having only a few nonzero values for t_j (that is, requiring implementation of only a few different filters). By Carathéodory’s theorem [41], only N of the filters need to be nonzero for any fixed value of $\Omega_{\text{eff}}/\Omega$ in order to guarantee a solution. However, finding such sparse solutions is in general NP-hard, so one would likely have to resort to heuristic methods to make improvements along these lines.

5. Example programs

Some examples of Hamiltonian engineering will help to reveal the power, generality, and utility of our approach. A simple example is the application of the filter $[f_k^A(d) + I]$, where I indicates free evolution of equal duration. This particular filter ensures that all qubits separated by distance k will be fully decoupled, combining a period of pulsed modulation with free-evolution, and exploiting the fact that $f_k^A(k) = -1$. Using concatenation, we can eliminate all interactions in the spin chain except for, e.g., nearest-neighbor interactions. This Hamiltonian is vital not only for simulation of quantum magnetism, but also many other quantum information protocols [42, 43].

A particularly useful example of a relevant Hamiltonian mapping relates to problems in quantum magnetism [44–46] where long-range qubit interactions can be engineered to scale as $\Omega_d \propto d^{-\alpha}$, $\alpha \in [0, 3]$: a form of interaction that arises in phonon-mediated spin simulators using trapped ions [35, 47, 48]. In practice, many simulators cannot reach the achievable limits of this range, or there may be a desire to induce a scaling outside of the range of this native interaction.

Here we provide details of how, using the pulse sequences we've introduced, one may adjust power-law interactions beyond the accessible bounds native to the underlying hardware. As an example, consider a system whose qubits interact through a Coulomb interaction such that $\Omega_d \propto \Omega_1/d$. In order to use this system to simulate a many-body system whose particles interact through a $1/d^2$ potential, we must construct a filter with functional dependence $f(d) \propto 1/d$ so that $\Omega_d \rightarrow \Omega_d/f(d)$. Such a filter may obviously be applied repeatedly to modify arbitrary power-law interactions.

For a system of N qubits, this transformation may be achieved through the sequential application of increasingly complex concatenations of Λ_{N+1} . Any Λ_k , $k > N$ can be used, but $k = N + 1$ minimizes the number of pulses in the primitive filter. In a system of N qubits, the map generated by Λ_k is linear over the entire chain when $k > N$ so that it can be described as $f_k^\Lambda(d) = 1 - 2\frac{d}{k}$. Rewriting this expression as $-\frac{2}{k}(d - \frac{k}{2})$ highlights that this is equal to the linear term in the Taylor expansion of $\frac{k/2}{d}$ around $d = k/2$, with a radius of convergence of $k/2$:

$$\frac{k/2}{d} = \sum_{j=0}^{\infty} \left(\frac{-(d - k/2)}{k/2} \right)^j. \quad (13)$$

We define the evolution operator U_{1/d^n} which is generated by an interaction that obeys a power law of $\Omega_d = \Omega_1/d^n$. Consider the following filtered evolution operator V ,

$$V = U_{1/d^n}(t)\Lambda_{N+1}(U_{1/d^n}(t))\Lambda_{N+1}^2(U_{1/d^n}(t))\Lambda_{N+1}^3(U_{1/d^n}(t))\dots \quad (14)$$

$$= \exp \left[-i \sum_{d=1}^N \frac{\Omega_1}{d^n} \left(1 + f_{N+1}^\Lambda(d) + (f_{N+1}^\Lambda(d))^2 + \dots \right) t \sum_{j=1}^{N-d} X_j X_{j+d} \right] \quad (15)$$

$$= \exp \left[-i \sum_{d=1}^N \frac{\Omega_1(N+1)}{2d^{n+1}} t \sum_{j=1}^{N-d} X_j X_{j+d} \right] = U_{1/d^{n+1}}. \quad (16)$$

Here the exponent on the symbol Λ_k^x denotes concatenation to x levels, and the last line follows by summing the geometric series. The effective interaction strength now obeys a new inverse power law where the exponent has been incremented by a single power. More generally, one may choose different timings for the various filtering operations in order to construct a more general dynamical mapping function,

$$V = U_{1/d^n}(\alpha_0 t)\Lambda_{N+1}(U_{1/d^n}(\alpha_1 t))\Lambda_{N+1}^2(U_{1/d^n}(\alpha_2 t))\Lambda_{N+1}^3(U_{1/d^n}(\alpha_3 t))\dots \quad (17)$$

$$= \exp \left[-i \sum_{d=1}^N \frac{\Omega_1}{d^n} g(d, \{\alpha_i\}) t \sum_{j=1}^{N-d} X_j X_{j+d} \right]. \quad (18)$$

The function $g(d, \{\alpha_i\})$ is restricted by the fact that the coefficients $\{\alpha_i\}$ are necessarily positive. This implies that the function f must have Taylor expansion coefficients that alternate

in sign, meaning that the technique is capable of producing any inverse power law filter but is incapable of producing polynomials with positive exponents. Another possibility to realize these power laws is to use the linear program of the previous section together with a family of pulse sequences to try to find the most time-efficient power law across all N of the spins.

6. Application to adiabatic protocols

Next, we describe how this approach may be applied to adiabatic simulators and preparation of entangled ground-states of designer Hamiltonians [49]. Say we wish to adiabatically evolve to the ground state of a target Hamiltonian, H_t , but can only turn on the available Hamiltonian H_a in a particular experimental apparatus. Returning again to our 1D qubit chain, we initialize in $|g_s\rangle$, which is the ground state of a simple Hamiltonian such as $H_s = -\frac{\omega}{2}\sum_j Z_j$, and allow the system to evolve under the time-dependent Hamiltonian, $H(t) = \left(1 - \frac{t}{\tau}\right)H_s + \frac{t}{\tau}H_a$. If the evolution is adiabatic, then $|\langle g_a | \hat{U}_a(\tau) | g_s \rangle|^2 \approx 1$ at the end of the interaction.

Now we apply stroboscopic filtering to drive the system to the new ground state of H_t , $|g_t\rangle$. The filtering must be done repeatedly throughout the adiabatic evolution since the Hamiltonian generally does not commute with itself at different times. The rate of application is set such that, to first order, the Hamiltonian is constant over the pulse period, Δt , giving a total evolution defined by $\prod_{l=1}^R \mathcal{F}[\hat{U}_a(l\Delta t, (l-1)\Delta t)]$ with $R = \tau/\Delta t$ the number of filtering operations performed during the adiabatic ramp, \mathcal{F} the dynamic filter, and $\hat{U}_a(t_1, t_2)$ the time-ordered evolution from t_1 to t_2 .

Using the filtering approach in adiabatic simulators to reach a target ground state $|g_t\rangle$, requires repeatedly applying the sequence at a rate fast enough that the evolving Hamiltonian is approximately constant over the time it takes to apply the pulse sequence. Defining a filter \mathcal{F} that maps the evolution operator generated by H_a to one generated by H_t gives a piecewise-constant filtered adiabatic evolution operator

$$U_a = \prod_{j=1}^R \mathcal{F}\left(\exp\left[-iH(t_j)\Delta t\right]\right), \quad (19)$$

where $R = \tau/\Delta t$ is the number of filtering operations performed during the adiabatic ramp. The state then evolves during these discrete timesteps as

$$|\psi\rangle = e^{-iH_f^{(R)}\Delta t} e^{-iH_f^{(R-1)}\Delta t} \dots e^{-iH_f^{(1)}\Delta t} |g_s\rangle. \quad (20)$$

Here $H_f^{(j)}$ is the filtered Hamiltonian at $t = j\Delta t$.

The error accrued due to the Hamiltonian changing during the filtering operations can be calculated as the overlap of $|\psi\rangle$ with a state $|\phi\rangle$ whose evolution includes the first-order time-dependence of the Hamiltonian during filtering operations. In the adiabatic approximation, the time evolution operator is given by $\exp[-i\int_t^{t+\Delta t} H(t')dt']$ which can be approximated by $\exp[-i(H(t)\Delta t + \frac{1}{2}\dot{H}(t)\Delta t^2)]$ when Δt is small. In cases like ours where, $\hat{P}_j \frac{d}{dt} H(t) \hat{P}_j = \frac{d}{dt} \hat{P}_j H(t) \hat{P}_j$ we can write

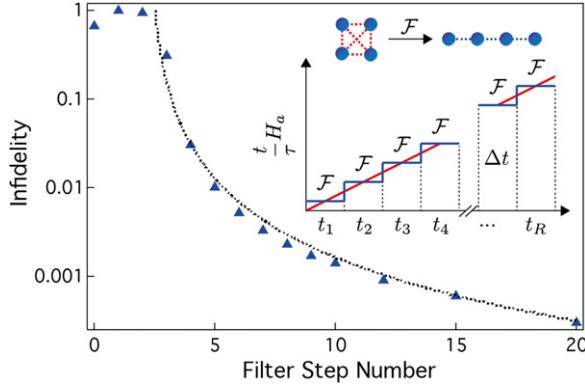


Figure 3. Application of dynamic filtering to the adiabatic evolution of a quantum simulator. Inset: schematic of the approximation of breaking a linear ramp of Hamiltonian H_a into piecewise-constant segments during which a dynamic filter, \mathcal{F} is applied. In the example treated here, we filter an all-to-all interaction between four spins to give only nearest-neighbor interactions using $\mathcal{F}(U_0) = (\Lambda_2 \circ \Lambda_3)(U_0)$. In the main panel we calculate the state fidelity of the adiabatic evolution as a function of the number of filtering operations. The error, $\approx \text{Var}(H_f^{(N)})\Delta t^2/4\hbar^2 + O(\Delta t^3)$, where the variance is calculated with respect to the initialized ground state of H_s , decreases quadratically with filtering step. The dashed line is a guide to the eye showing the quadratic improvement.

$$|\phi\rangle = e^{-i(H_f^{(N)}\Delta t + \frac{1}{2}\dot{H}_f^{(N)}\Delta t^2)} e^{-i(H_f^{(N-1)}\Delta t + \frac{1}{2}\dot{H}_f^{(N-1)}\Delta t^2)} \dots e^{-i(H_f^{(1)}\Delta t + \frac{1}{2}\dot{H}_f^{(1)}\Delta t^2)} |g_s\rangle, \quad (21)$$

which is meant to capture the detrimental effect on the filtering operation due to the changing Hamiltonian to lowest order in Δt . The fidelity of the filtered adiabatic protocol can be defined as the overlap of $|\psi\rangle$ and $|\phi\rangle$ as defined in equations (20) and (21). Thus, we have

$$|\langle\phi|\psi\rangle|^2 = \left| \langle g_s | e^{i(H^{(1)}\Delta t + \frac{1}{2}\dot{H}^{(1)}\Delta t^2)} \dots e^{i(H^{(R)}\Delta t + \frac{1}{2}\dot{H}^{(R)}\Delta t^2)} e^{-iH^{(R)}\Delta t} \dots e^{-iH^{(1)}\Delta t} | g_s \rangle \right|^2 \quad (22)$$

$$= \left| \langle g_s | e^{i\frac{1}{2}\sum_{j=1}^R \dot{H}^{(j)}\Delta t^2 + O(\Delta t^3)} | g_s \rangle \right|^2, \quad (23)$$

by the Baker–Campbell–Hausdorff formula. The sum $\sum_{j=1}^R \dot{H}^{(j)}\Delta t$ can be approximated by an integral when Δt is small enough so that the overlap is,

$$|\langle\phi|\psi\rangle|^2 \approx \left| \langle g_s | e^{i\frac{1}{2}\int_{t_1}^{t_f} \dot{H}(t) dt} | g_s \rangle \right|^2 \quad (24)$$

$$= \left| \langle g_s | e^{i\frac{1}{2}(H(t_f) - H(t_1))\Delta t} | g_s \rangle \right|^2. \quad (25)$$

From this last expression, we see that the error can be thought of as the probability of the Hamiltonian $\frac{1}{2}(H(t_f) - H(t_1))$ evolving the system out of the state $|g_s\rangle$ in a time Δt . To lowest order in Δt , this error is $\text{Var}(H(t_f) - H(t_1))\Delta t^2/4\hbar^2$ where the variance is calculated with respect to $|g_s\rangle$. In the adiabatic evolution described in this article, $|g_s\rangle$ is the ground state of

$H(t_1)$, meaning that the expression for the error can be further simplified to $\text{Var}(H(t_f))\Delta t^2/4\hbar^2$.

To test this, we numerically integrate the Schrödinger equation for a system of four qubits undergoing adiabatic evolution (figure 3), starting from a distance-independent spin coupling, $\Omega_d \equiv \Omega$, and filtered to produce only nearest-neighbor coupling. We calculate the error accrued due to deviation from the assumption of a piecewise-constant Hamiltonian during the filtering operations as the overlap of the ideal state with a state whose evolution includes a first-order time-dependence of the Hamiltonian during filtering and find that the infidelity decreases approximately as $(\Delta t)^2$. In this case, the overlap between the ground state of the target Hamiltonian and that of the unfiltered Hamiltonian is $|\langle g_t | g_a \rangle|^2 = 0.33$, but with the application of dynamic filters, surpasses 10^{-2} after just six filtering steps.

7. Moving beyond Ising interactions

We now relax the original constraints we placed on our interaction Hamiltonian in order to simplify the analytic treatments presented above in demonstrating the basic functionality of our approach. All of these results continue to hold when our Ising-type model is generalized to include both the homogeneous on-site terms and the presence of long-range Heisenberg-type interactions.

7.1. Heisenberg interactions

Similar to the work of Hedges *et al* [50], an Ising interaction (e.g. XX) can be modified into a Heisenberg-type ($S \cdot S$) by including global basis-changing $\pi/2$ pulses in the sequence of applied filters. Similarly, any native Heisenberg interaction may be modified into any other using pulse sequences applied in the appropriate basis and concatenated. We may rigorously bound the error and runtime as a polynomial function of T and N , finding only a constant multiplicative penalty in run-time and Trotter error (see section 7.2). Suppose that our target Hamiltonian is of the form (cf equation (1)).

$$H = \sum_d \left(\Omega_d^x \sum_j X_j X_{j+d} + \Omega_d^y \sum_j Y_j Y_{j+d} + \Omega_d^z \sum_j Z_j Z_{j+d} \right). \quad (26)$$

For example, if $\Omega_j^v = 0$ for $v = x, y, z$ and $j \geq 2$, then this is the standard spin-1/2 XYZ model. This Hamiltonian can be engineered through a simple extension of the method we describe above. Assuming the native interaction Hamiltonian consist solely of X_j operators, we build up the time evolution operator using another layer of Trotterization. To be more precise, we generate the Y_j interactions by applying global $\pi/2$ pulses about the Z axis, and the Z_j interactions are generated by global $\pi/2$ pulses about the Y axis. If the filtering sequence is represented by the superoperator \mathcal{F} , and the global $\pi/2$ pulses are represented by \mathcal{G} , the Heisenberg evolution U_H is given by,

$$U_H = \mathcal{F}(U)\mathcal{G}_z\mathcal{F}(U)\mathcal{G}_z\mathcal{G}_y\mathcal{F}(U)\mathcal{G}_y. \quad (27)$$

While this method necessarily incurs a Trotter-type error in the implementation, we demonstrate below that this error can be bounded, and provide such a bound for the simulation error (that is, for the case of perfect gates).

7.2. Inclusion of a transverse field

Previously, in the case of the Ising coupling, the transverse field was set to zero so that all of the terms in the Hamiltonian would commute with one another. This resulted in an exact mapping from an initial Hamiltonian of the form

$$H_I = \sum_d \Omega_d \sum_j X_j X_{j+d} \quad (28)$$

to another Hamiltonian of the same form, but with the distance function being modified as $\Omega_d \rightarrow \Omega_d f(d)$. The functional dependence $f(d)$ can take an arbitrary form, up to an overall rescaling factor.

To incorporate a homogeneous transverse-field term of the form

$$H_T = B \sum_j Z_j, \quad (29)$$

we use a Suzuki–Trotter-type decomposition of the evolution operator [51–53]. We assume that the overall strength B of the transverse field is a tunable parameter so that we can match its rescaled strength relative to the rescaled coupling strength $\tilde{\Omega}_d$ by the same overall scale factor. Given this assumption, there are two natural ways to analyze this situation: an always-on field and a field which can be switched on and off rapidly.

In some implementations, it is further reasonable to expect that the *interactions* can be switched on and off rapidly (e.g., some trapped ion experiments [54]). We will also analyze this case, since in fact it is the easiest to analyze and gives insight into the other results.

7.3. Bounding the Trotter error in a simulation

We wish to bound the deviation between the exact and the approximate (Suzuki–Trotter) evolution, assuming that all of our pulse sequences are accurate and perfectly timed. We denote the ideal evolution operator by

$$V(t) = \exp \left[-i \sum_j \hat{P}_j (H_T + H_I) \hat{P}_j \Delta t \right] = \exp \left[-i (H_T + \tilde{H}_I) t \right], \quad (30)$$

where $(\sum_j \hat{P}_j H_I \hat{P}_j) \Delta t = \tilde{H}_I t$. The approximate evolution is defined in terms of the second-order integrator (the ‘split-step’ method), given by the following formula:

$$U_r(t) = \left(\prod_{j=1}^m \exp \left[-i H_j t / 2r \right] \prod_{j'=m}^1 \exp \left[-i H_{j'} t / 2r \right] \right)^r. \quad (31)$$

Here the H_j are the allowed evolutions over a given time interval, and their exact definition depends on which resources we allow ourselves in the protocol, but they are related to the ideal effective Hamiltonian (here called simply H) by

$$H = H_T + \tilde{H}_I = \sum_{j=1}^m H_j. \quad (32)$$

We will discuss the specific choices below. We have also introduced a parameter r , the Trotter number, which counts the number of steps in our Suzuki–Trotter expansion and hence controls our error.

Note that we *cannot* use the higher-order integrators which provide faster convergence because they all require evolution for negative values of t [53]. If rapidly flipping the sign of the interactions and the transverse field is feasible in a given system, then these higher-order integrators could be used to give sharper error bounds than the ones given below.

Given this very general framework, we can upper bound the error as follows [55]

$$\| V(t) - U_r(t) \| \leq 16 m^3 \| H \|^3 t^3 / r^2. \quad (33)$$

Here the error is in terms of the operator norm (largest singular value). Note that $\| H \| \leq N^2/2 + BN$ by the triangle inequality, so this quantity is always polynomial in the number of spins. This bound on the norm of the difference has an operational meaning: it gives an upper bound on the trace distance between the evolved states, namely

$$\| V - U \| \geq \frac{1}{2} \text{Tr} |V\rho V^\dagger - U\rho U^\dagger| \quad (34)$$

for any initial state ρ and unitaries U and V .

To make the error bound in equation (33) more concrete, we need to determine m for a given protocol. The form of the bound above indicates that a smaller number of different Hamiltonians in use results in improved error scaling.

The simplest case to consider is that where both the interactions and the transverse field can be switched on and off rapidly. In this case we can use the above bound with $m = 2$, in which we simply switch between (1) the exact evolution of the effective Hamiltonian with zero transverse field, and (2) the transverse field alone with no interactions. That is, we have

$$H_1 = \tilde{H}_I \quad \text{and} \quad H_2 = H_T. \quad (35)$$

Note that \tilde{H}_I can be implemented exactly by subdividing each Suzuki-Trotter step into smaller segments, as discussed in the main text.

Next, consider the case where the interactions are always on, but the transverse field can be switched on and off rapidly. Here a good strategy is to turn on the transverse field in short, strong bursts, and turn it off while applying the filter functions. The control Hamiltonians during each of the $m = 2$ Suzuki–Trotter steps are given by

$$H_1 = \tilde{H}_I \quad \text{and} \quad H_2 = \delta \left(H_I + \frac{1}{\delta} H_T \right). \quad (36)$$

Here we introduce an additional parameter $\delta \ll 1$, and we incur a small error in the final Hamiltonian,

$$H_\delta = \tilde{H}_I + H_T + \delta H_I. \quad (37)$$

We can choose δ as small as possible to be compatible with the assumption of accurate and rapid switching of the transverse field. Thus, the error bound is nearly the same as the previous $m = 2$ bound, except that we add a small error from the inexact decomposition.

$$\begin{aligned} \| V(t) - U_r(t) \| &\leq \| V(t) - V_\delta(t) \| + \| V_\delta(t) - U_r(t) \| \\ &\leq 128 \| H_\delta \|^3 t^3 / r^2 + O(\delta \| H_I \| t). \end{aligned} \quad (38)$$

Next, consider the most pessimistic case where both the interactions and the transverse field are not rapidly switchable. In this case the best strategy seems to be to use a different Suzuki–Trotter step for each increment in a pulse sequence. In this case, we have no choice but to use the bound in equation (33) by setting m equal to the total number of pulses in the pulse sequence. For a chain of length N , we can always bound the maximum number of pulses required by $m \leq N$, which follows from Carathéodory’s theorem [41]. This gives a pessimistic but nonetheless polynomial bound on the required Trotter number to achieve a fixed error.

Finally, we remark that the same methods for bounding the Trotter error in this section also apply in section 7. Since this more general case includes, at most, three times the number of pairwise coupling terms in the Hamiltonian, this implies that the error remains polynomial in the number of qubits.

8. Conclusion

In conclusion, we have provided a general framework for dynamic filtering of spin Hamiltonians that enables programmable quantum simulators using single-qubit Pauli operations and a native long-range spin coupling. The scheme we introduce provides a system designer with the ability to perform numerical decomposition of a large class of realizable couplings into the basis of available filters, thus providing a prescription of how to ‘program’ the desired interactions. We have further showed how this technique can be applied to augment the adiabatic evolution of a spin Hamiltonian in a form of hybrid adiabatic quantum simulator.

We are excited by the fact that our approach to combine such filters in realizing a desired program for quantum simulation represents a contribution to a growing body of literature focused on the benefits of functional analysis in the context of protocol, algorithm, and circuit development for quantum information. In particular, following completion of this work we discovered significant similarity between our approach and the synthesis of diagonal operators for circuit-model quantum computation [24]. Given these connections and the breadth of applicability of functional analysis in quantum information, we hope that our results enable new capabilities well beyond quantum simulation.

The results presented herein demonstrate the utility of the pulse sequences we define for efficiently modifying native Hamiltonians in quantum simulators. From the perspective of control engineering these results demonstrate that via the quantum control protocols we have defined, the 1D translationally invariant quantum simulator in which we are interested is *efficiently controllable*, requiring resources scaling only polynomially in time and energy (using pulse number as a proxy). Assuming only Ising-type long range interactions, we may realize a much broader class of Hamiltonians including spin-1/2 XYZ Heisenberg-type models with homogeneous on-site terms and much more exotic multi-axis spin-coupling Hamiltonians with bounded error and polynomial resources. All of our pulse sequences and methods of filter combination may be applied in order to achieve these unusual models with the small associated penalties in time and resources.

While we have provided explicit examples of how one might tune the power-law of a long-range spin coupling or completely cancel undesired spin couplings on a 1D lattice, it is an interesting open problem to find additional examples where our method can be applied efficiently. This includes, for instance, the efficient generation of an effective Heisenberg Hamiltonian using only Ising couplings and single-qubit control such as that in [50], or

generalizations of our method to systems with a natural 2D geometry of spins. A fundamental open problem is also finding optimal primary filter constructions that minimize the overhead in evolution time for a desired effective Hamiltonian. In addition we wish to explore the possibility of finding a rigorous proof of polynomial runtime scaling, rather than relying only on numeric evidence supporting our claim. Finally, it is an exciting open question to explore how these techniques may be coupled with notions of fault-tolerance in quantum simulation. We note that after completion of this work we became aware of the use of similar language of Hamiltonian ‘filtering’ for quantum simulation, treating quantum simulation and spin transport in a related framework [56].

Acknowledgments

The authors thank L Viola for useful discussions. This work partially supported by the US Army Research Office under contract number W911NF-11-1-0068, the Australian Research Council Centre of Excellence for Engineered Quantum Systems CE110001013, the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the Army Research Office, and the Lockheed Martin Corporation. STF acknowledges support from an ARC Future Fellowship. All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of IARPA, the ODNI, or the US Government.

Appendix. Construction of C_N and proof of universality

Here we provide a proof of the existence of a universal filter space for an $(N - 1)$ -dimensional interaction space appropriate for N qubits. In order to construct a suitable set of extreme points to ensure that their convex hull contains the origin and a hypercube around it, we show that it is possible to generate a complete set of effective interactions such that only qubits that are separated by a specific distance possess nonzero interaction, i.e. $\Omega_{d'} \neq 0$ iff $d' = d$, for a single prescribed d . These basis interactions, to which we refer as the Kronecker delta interaction vectors, would be of the form $\alpha(0, 0, \dots, 0, \pm 1, 0, \dots, 0)$ across the $N - 1$ dimensions, where α is a positive number, and the \pm sign permits the realization of either ferromagnetic or antiferromagnetic interactions. While large α is desirable from the practical perspective of reducing necessary operation times, the actual value does not impact a proof of the existence of a universal filter space. We will use this basis exclusively in order to present our proof of universal filtering for N qubits, but more efficient approaches to the construction of C_N likely exist.

We proceed by showing how to construct both a positive and negative Kronecker delta interaction vector for the special cases $d = 1, 2, 3, 4$ and finally for the generic case $4 < d < N$. The $d = 1$ negative Kronecker delta interaction uses Λ_1 as its base function, which takes value $f_1^\Lambda(d) = -1$ at $d = 1$. We eliminate all other couplings through a judicious choice of concatenations which are enumerated through the expression, valid for all $1 < d < 2^m - 1$,

$$(f_2^\Lambda + f_0^\Lambda) \prod_{n=2}^m (f_{2^n}^\Lambda + f_0^\Lambda) \left(f_{2^n}^\Lambda + \left(1 - \frac{1}{2^{n-1}}\right) f_0^\Lambda \right) = 0. \quad (\text{A.1})$$

This statement can be proved by induction with the base case of $m = 2$ being easily checked by hand.

For the case of an arbitrarily long chain of qubits, we now explicitly describe the inductive step. Assuming that we have set to zero all couplings up to $2^{n+1} - 1$ using the concatenation scheme described above, we must show that subsequent concatenation with $(\Lambda_{2^{n+1}} + \Lambda_0)$ and $(\Lambda_{2^{n+1}} + (1 - \frac{1}{2^n})\Lambda_0)$ will extend the decoupling range out to $d < 2^{n+2} - 1$. Using the explicit form for $f_k^\Lambda(d)$, it is trivial to show that these additional concatenations zero out the $2^{n+1} - 1$, 2^{n+1} and $2^{n+1} + 1$ couplings. Additionally, every filter used in the preceding concatenations takes value 1 at $d = 2^{n+1}$ (since they all have a periodicity equal to a power of 2), implying that their behavior from $1 < d < 2^{n+1}$ will be repeated for $2^{n+1} < d < 2^{n+2}$. Given the assumption of the inductive step, this implies that the concatenation has extended the full decoupling to $d < 2^{n+2} - 1$.

Constructing the positive Kronecker delta interaction for $d = 1$ is accomplished in the same manner, in this case beginning with Λ_0 as the base function. In this rubric, a chain of N qubits requires $2m - 1 = 2\lceil \log_2(N + 1) \rceil - 3$ total concatenations for the Kronecker delta interaction at $d = 1$.

The negative $d = 2$ Kronecker delta interaction begins with Λ_2 as the base function followed by concatenation with $(\Lambda_1 + \Lambda_0)$ to eliminate all interactions for d odd. The task of eliminating residual interactions maps directly onto the construction of the $d = 1$ Kronecker delta interaction, rescaling all distances by a factor of 2, using the following:

$$(f_1^\Lambda + f_0^\Lambda)(f_4^\Lambda + f_0^\Lambda) \prod_{n=3}^m (f_{2^n}^\Lambda + f_0^\Lambda) \left(f_{2^n}^\Lambda + \left(1 - \frac{1}{2^{n-2}}\right) f_0^\Lambda \right) = 0 \quad \forall d \neq 2. \quad (\text{A.2})$$

This scheme requires a total of $2\lceil \log_2(N) \rceil - 4$ concatenations for the $d = 2$ Kronecker delta interaction so that $2m - 2 = \lceil \log_2(N - 1) \rceil - 2$.

The $d = 3$ Kronecker delta interaction is constructed starting with Λ_3 and concatenating with $(\Gamma_3 + \Lambda_0/3)$ to decouple all qubits except those that are separated by multiples of three lattice spacings. The remaining task can be accomplished by a similar sequence of filters indicated by the expression,

$$\left(f_3^\Gamma + \frac{1}{3}f_0^\Lambda\right) \prod_{n=0}^m \left(f_{9 \cdot 2^n}^\Lambda + f_0^\Lambda\right) \left(f_{9 \cdot 2^n}^\Lambda + \left(1 - \frac{1}{3 \cdot 2^{n-1}}\right) f_0^\Lambda\right) = 0 \quad \forall d \neq 3, \quad (\text{A.3})$$

requiring $2m + 3 = 2\lceil \log_2(\frac{N+2}{9}) \rceil + 1$ concatenations.

The $d = 4$ Kronecker delta interaction construction starts with Λ_4 and proceeds via concatenation with Γ_4 to decouple qubits that are *not* separated by a multiple of four lattice spacings. The decoupling is accomplished via the concatenation of the filters indicated by,

$$f_4^\Gamma \prod_{n=0}^m \left(f_{12 \cdot 2^n}^\Lambda + f_0^\Lambda\right) \left(f_{12 \cdot 2^n}^\Lambda + \left(1 - \frac{1}{3 \cdot 2^{n-1}}\right) f_0^\Lambda\right) = 0 \quad \forall d \neq 4, \quad (\text{A.4})$$

requiring $2m + 3 = 2\lceil \log_2(\frac{N+3}{12}) \rceil + 1$ concatenations.

The final construction is for the Kronecker delta interaction at $d = Q$ where $4 < Q \leq N - 1$, employing Γ_Q which is positive for all d . For $d < Q$, $f_Q^\Gamma(d) = 1 - \frac{4}{Q}$.

Applying $((1 - \frac{4}{Q})\Lambda_{2^m} + \Gamma_Q)$ for $m = 0, 1, 2, 3, \dots, \lceil \log_2 N \rceil$ decouples all qubits except those

that are separated by multiples of Q lattice spacings. The next task of decoupling all qubits separated by multiples of Q can again be mapped onto our original construction of the $d = 1$ Kronecker delta interaction by rescaling the distance by Q . The entire decoupling sequence is indicated by the following expression which is valid for all $d \neq Q$,

$$\left(\prod_{n=0}^m \left(f_{2^n}^\Lambda + \frac{1}{\left(1 - \frac{4}{Q}\right)} f_Q^\Gamma \right) \right) \left(f_{2Q}^\Lambda + f_0^\Lambda \right) \prod_{q=2}^p \left(f_{2^q Q}^\Lambda + f_0^\Lambda \right) \left(f_{2^q Q}^\Lambda + \left(1 - \frac{1}{2^{q-1}}\right) f_0^\Lambda \right) = 0 \quad (\text{A.5})$$

resulting in a total of $m + 2p = \lceil \log_2(N - 1) \rceil + 2 \lceil \log_2\left(\frac{N-1}{Q}\right) \rceil$ filtering operations.

We thus see that it is possible to construct the Kronecker delta interaction with $O(\log_2(N))$ concatenations. Accordingly we know that it is possible to define a convex hull that includes the origin and the extremal points defined by the Kronecker deltas above, proving universality for an N -qubit chain as defined herein.

The logarithmic scaling we present here is much better than brute force, however the effective coupling strengths that result from $\log N$ levels of concatenation shrink polynomially with N , leading to decreased simulation efficiency for large values of N in this particular construction. An obvious open question is if this universal filtering space can be constructed with only a *constant* number of concatenations, which would be a very strong result.

References

- [1] MacLennan B J 2007 *Encyclopedia of Complexity and System Science* ed R A Meyers (Berlin: Springer) pp 271–94
- [2] Somma R, Ortiz G, Gubernatis J E, Knill E and Laflamme R 2002 *Phys. Rev. A* **65** 042323
- [3] Somaroo S, Tseng C H, Havel T F, Laflamme R and Cory D G 1999 *Phys. Rev. Lett.* **82** 5381–4
- [4] Blatt R and Roos C F 2012 *Nat. Phys.* **8** 277–84
- [5] Tseng C H, Somaroo S, Sharf Y, Knill E, Laflamme R, Havel T F and Cory D G 1999 *Phys. Rev. A* **61** 012302
- [6] Kim K, Chang M S, Korenblit S, Islam R, Edwards E E, Freericks J K, Lin G D, Duan L M and Monroe C 2010 *Nature* **465** 590
- [7] Edwards E E, Korenblit S, Kim K, Islam R, Chang M S, Freericks J K, Lin G D, Duan L M and Monroe C 2010 *Phys. Rev. B* **82** 060412
- [8] Gerritsma R, Lanyon B P, Kirchmair G, Zähringer F, Hempel C, Casanova J, García-Ripoll J J, Solano E, Blatt R and Roos C F 2011 *Phys. Rev. Lett.* **106** 060503
- [9] Barreiro J T, Muller M, Schindler P, Nigg D, Monz T, Chwalla M, Hennrich M, Roos C F, Zoller P and Blatt R 2011 *Nature* **470** 486
- [10] Lanyon B P *et al* 2011 *Science* **334** 57
- [11] Bloch I, Delibard J and Nascimbene S 2012 *Nat. Phys.* **8** 267–76
- [12] Schindler P, Muller M, Nigg D, Barreiro J T, Martinez E A, Hennrich M, Monz T, Diehl S, Zoller P and Blatt R 2013 *Nat. Phys.* **9** 361
- [13] Wocjan P, Rötteler M, Janzing D and Beth T 2002 *Phys. Rev. A* **65** 042309
- [14] Dodd J L, Nielsen M A, Bremner M J and Thew R T 2002 *Phys. Rev. A* **65** 040301
- [15] Wocjan P, Rötteler M, Janzing D and Beth T 2002 *Quantum Inf. Comput.* **2** 133–50
- [16] Aspuru-Guzik A and Walther P 2012 *Nat. Phys.* **8** 285–91
- [17] Korenblit S *et al* 2012 *New J. Phys.* **14** 095024
- [18] Leung D W, Chuang I L, Yamaguchi F and Yamamoto Y 1999 *Phys. Rev. A* **61** 042310

- [19] Leung D 2002 *J. Mod. Opt.* **49** 1199–217
- [20] Hayes D, Khodjasteh K, Viola L and Biercuk M J 2011 *Phys. Rev. A* **84** 062323
- [21] Khodjasteh K, Sastrawan J, Hayes D, Green T J, Biercuk M J and Viola L 2013 *Nat. Commun.* **4** 2045
- [22] Owrusky P and Khaneja N 2012 arXiv:1204.0061
- [23] Soare A, Ball H, Hayes D L, Sastrawan J, McLoughlin J J, Zhen X, Green T J and Biercuk M J 2014 arXiv:1404.0820
- [24] Welch J, Greenbaum D, Mostame S and Aspuru-Guzik A 2014 *New J. Phys.* **16** 033040
- [25] Cooper A, Magesan E, Yum H and Cappellaro P 2014 *Nat. Commun.* **5** 3141
- [26] Levitt M H 1986 *Prog. Nucl. Magn. Reson. Spectros.* **18** 61
- [27] Viola L and Lloyd S 1998 *Phys. Rev. A* **58** 2733
- [28] Biercuk M J, Uys H, VanDevender A P, Shiga N, Itano W M and Bollinger J J 2009 *Nature* **458** 996
- [29] Ng H K, Lidar D A and Preskill J 2011 *Phys. Rev. A* **84** 012305
- [30] Peng X, Suter D and Lidar D A 2011 *J. Phys. B* **44** 154003
- [31] Stollsteimer M and Mahler G 2001 *Phys. Rev. A* **64** 052301
- [32] Lloyd S 1993 *Science* **261** 1569
- [33] Brennen G K, Deutsch I H and Jessen P S 2000 *Phys. Rev. A* **61** 062309
- [34] Saffman M, Walker T G and Molmer K 2010 *Rev. Mod. Phys.* **82** 2313
- [35] Britton J W, Sawyer B C, Keith A C, Wang C C J, Freericks J K, Uys H, Biercuk M J and Bollinger J 2012 *Nature* **484** 489–93
- [36] Green D N and Bass S C 1979 *IEE J. Electron. Circuits Syst.* **3** 58–68
- [37] Khodjasteh K and Viola L 2009 *Phys. Rev. Lett.* **102** 080501
- [38] Green T, Uys H and Biercuk M J 2012 *Phys. Rev. Lett.* **109** 020501
- [39] Bookatz A D, Wocjan P and Viola L 2013 arXiv:1310.4153
- [40] Boyd S and Vandenberghe L 2004 *Convex Optimization* (Cambridge: Cambridge University Press)
- [41] Rockafellar R T 1972 *Convex Analysis* (Princeton, NJ: Princeton University Press)
- [42] Ronke R, Spiller T and D’Amico I 2011 *J. Phys.: Conf. Ser.* **286** 012020
- [43] Keating T, Goyal K, Jua Y Y, Biedermann G, Landahl A and Deutsch I H 2013 *Phys. Rev. A* **87** 052314
- [44] Fisher M E, Ma S K and Nickel B G 1972 *Phys. Rev. Lett.* **29** 917–20
- [45] Sak J 1973 *Phys. Rev. B* **8** 281–3
- [46] Luijten E and Blote H W J 2002 *Phys. Rev. Lett.* **89** 025703
- [47] Porras D and Cirac J I 2004 *Phys. Rev. Lett.* **92** 207901
- [48] Kim K, Chang M S, Islam R, Korenblit S, Duan L M and Monroe C 2009 *Phys. Rev. Lett.* **103** 120502
- [49] Babbush R, Love P J and Aspuru-Guzik A 2013 arXiv:1311.3967
- [50] Hodges J S, Cappellaro P, Havel T F, Martinez R and Cory D G 2007 *Phys. Rev. A* **75** 042320
- [51] Trotter H F 1959 *Proc. Am. Math. Soc.* **10** 545–51
- [52] Suzuki M 1990 *Phys. Lett. A* **146** 6
- [53] Suzuki M 1991 *J. Math. Phys.* **32** 400–7
- [54] Islam R, Edwards E E, Kim K, Korenblit S, Noh C, Carmichael H, Lin G D, Duan L M, Wang C C J, Freericks J K and Monroe C 2011 *Nat. Commun.* **2** 377
- [55] Berry D W, Ahokas G, Cleve R and Sanders B C 2007 *Commun. Math. Phys.* **270** 359–71
- [56] Ajoy A and Cappellaro P 2013 *Phys. Rev. Lett.* **110** 220503