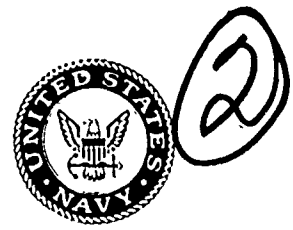
Naval Research Laboratory

Washington, DC 20375-5000



NRL Memorandum Report 6643

Variable Projection Methods With Application to Sums of Exponentials in White Noise

P. L. Ainsleigh

Underwater Sound Reference Detachment
Naval Research Laboratory
P.O. Box 568337
Orlando, FL 32856-8337

and

J. D. George

Signalmetrics, Inc.
Winter Park, FL 32789

27 December 1990

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	27 December 1990	Final FY89 til March 1990

4. TITLE AND SUBTITLE
Variable Projection Methods With Application to Sums of Exponentials in White Noise

5. FUNDING NUMBERS
PE# 62314N
WU# 59-0593-0-0
Assession# DN880-326

6. AUTHOR(S)
P. L. Ainsleigh, NRL-USRD
and
J. D. George, Signalmetrics, Inc.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Joint project between Signalmetrics, Inc., Winter Park, FL 32789 and Underwater Sound Ref. Detach., Naval Res. Lab., P.O. Box 568337, Orlando, FL 32856-8337

8. PERFORMING ORGANIZATION REPORT NUMBER
NRL Memorandum Report 6643

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
Dr. T. Goldsberry (ONT 231)
800 N. Quincy St.
Arlington, VA 22217

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
This report is a companion to NRL Memorandum Report 6644, "Stepped Sine Response Modeling Using Noisy Sampled Data," by J. D. George, P. L. Ainsleigh, and R. R. Muise. This work was supported by the Transducer Technology Project 59-0593-0-0.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Available for public release; distribution unlimitted

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

We examine Nonlinear Least-Squares (NLS) methods for estimating the pole and amplitude parameters for a sum of damped and undamped sinusoids observed in white noise. A special case of NLS, which takes advantage of the semilinear structure of this signal model, involves minimization of the Variable Projection Functional (VPF). The VPF offers computational efficiency by providing a means of estimating the signal poles independently of the amplitudes. We extend the existing set of algorithms for optimizing the VPF by deriving the exact Hessian matrix and introducing a Newton algorithm based on the exact Hessian. We also develop algorithms for optimizing the prediction filter VPF, which is a special case of the VPF especially suited for estimating the parameters of sums of exponentials, and introduce methods for constraining each error functional to contain known signal poles. In addition, we provide detailed background material covering computational linear algebra, optimization, and variable projection theory.

14. SUBJECT TERMS		15. NUMBER OF PAGES
Signal Modeling Nonlinear Least-Squares Optimization	Variable Projection	115 16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

(Blank Page)

CONTENTS

SYMBOLS

Signal Model

$\underline{a}$	basis function weighting coefficients (amplitudes)
$\underline{a}_1$	amplitudes for constant basis functions
$\underline{a}_2$	amplitudes for unknown basis functions
f	signal frequency
$F(\underline{\theta})$	signal basis function matrix
F_1	matrix of known basis functions
$F_2(\underline{\theta})$	matrix of unknown basis functions
$G(\underline{\theta})$	constrained basis funtion matrix
M	number of signal poles
M_1	number of constant signal poles
M_2	number of unknown signal poles
N	number of observations
T	sampling interval
$\underline{v}$	constrained observation vector
$\underline{x}$	ideal signal
$\underline{Y}$	observation vector
α	signal damping factor
$\underline{\epsilon}$	white noise
$\underline{\theta}$	signal pole parameters

Prediction Filter

$\underline{b}$	prediction filter coefficients
B	prediction filter convolution matrix
$\underline{c}$	constraint filter coefficients
C	constraint filter convolution matrix
$\underline{u}$	impulse response of system
U	system convolution matrix

SYMBOLS (CONT'D)

Linear Algebra

A^T	the transpose of a matrix A
A^+	the pseudoinverse of a matrix A
P_A	the projector onto the column space of matrix A
${}_AP$	the projector onto the row space of matrix A
$P^\perp$	the projector onto the orthogonal complement of a subspace
Q	orthogonal matrix (from QR factorization of A)
R	trapezoidal matrix from matrix factorization
R^N	real N-dimensional vector space
R_1	triangualr matrix (portion of trapezoidal matrix R)
S	subspace of R^N
	permutation matrix
$\bar{S}$	the orthogonal complement of the subspace S in R^N
V	orthogonal matrix (from RV factorization of A)

Optimization

$\underline{e}$	estimation error
$\underline{g}$	gradient vector
H	Hessian matrix
J	Jacobian matrix
$\chi(\underline{\theta},\underline{a})$	least-squares functional
$\chi_2(\underline{\theta},\underline{a})$	constrained least-squares functional
$\psi(\underline{\theta})$	signal basis variable projection functional
$\psi_2(\underline{\theta})$	constrained signal basis variable projection functional
$\psi_3(\underline{\theta})$	Kaufman's modified variable projection functional
$\phi(\underline{b})$	prediction filter variable projection functional
$\phi_2(\underline{u})$	constrained prediction filter variable projection functional

(Blank Page)

1.0 INTRODUCTION

In this report, we present methods for estimating the parameters in models of the form

$$x(\underline{\theta},\underline{a}) = F(\underline{\theta})\ \underline{a}\ , \tag{1-1}$$

which has been called a separable, reducible, or semilinear model [1],[2]. In this type of model, the columns of the matrix $F(\underline{\theta})$, called the basis function matrix, are the basis vectors that span the space of the model. The vector $\underline{\theta}$ contains the parameters that enter into the model nonlinearly (the nonlinear parameters), and the vector $\underline{a}$ contains the parameters that enter into the model linearly (the linear parameters). We assume that we have a vector, $\underline{y}$, containing observations of the model corrupted by white Gaussian noise.

In estimating $\underline{\theta}$ and $\underline{a}$, we restrict our attention to maximum likelihood estimation that (since we assume white Gaussian noise) reduces to least-squares estimation. That is, we minimize the cost functional (or error norm)

$$\chi(\underline{\theta},\underline{a}) = \left|\left|\ \underline{y} - \underline{x}(\underline{\theta},\underline{a})\ \right|\right|^2 . \tag{1-2}$$

A well known simplification of the least-squares problem takes advantage of the structure of the semilinear model and replaces Eq. (1-2) with a new cost functional called the Variable Projection Functional (VPF)

$$\psi(\underline{\theta}) = \left|\left|\ P_F^{\perp}(\underline{\theta})\ \underline{y}\ \right|\right|^2 . \tag{1-3}$$

Here $P_F^{\perp}(\underline{\theta})$ is the projector onto the orthogonal complement of the column space of the matrix $F(\underline{\theta})$. The term *variable projection functional* arises because the projector is a function that varies with the nonlinear parameters, unlike the fixed projector typically encountered in the linear least-squares problem (Figure 1-1 demonstrates the operation of a fixed projector). For those readers unfamiliar with the VPF, we provide tutorial material relating to the derivation of the VPF and to existing methods for minimizing the VPF.

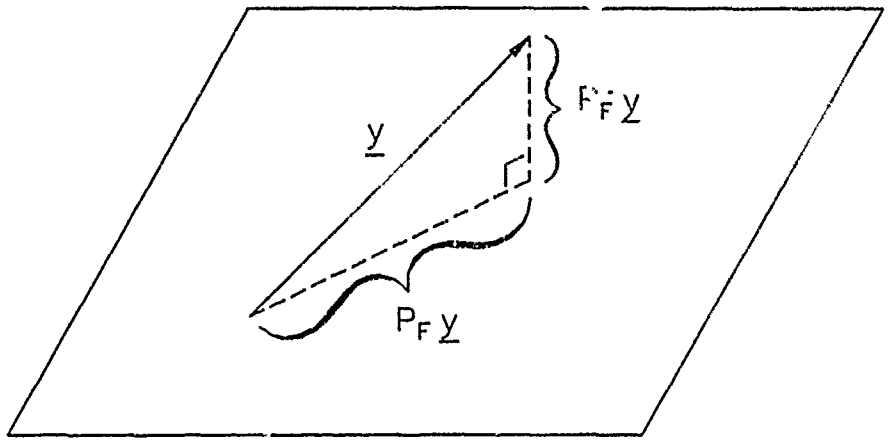


Fig. 1-1 The effect of projecting an arbitrary vector $\underline{y}$ onto the column space of the matrix F, symbolized by the plane, and onto the orthogonal complement of the column space.

As noted by Osborne [3], yet another equivalent cost functional can be obtained, if one has available some matrix $B(\underline{\theta})$ such that

$$B(\underline{\theta})\ F(\underline{\theta}) = 0. \tag{1-4}$$

This second type of VPF contains a projector onto the row space of the matrix $B(\underline{\theta})$, and is written

$$\phi(\underline{\theta}) = \left|\left|\ {}_{B}P(\underline{\theta})\ \underline{y}\ \right|\right|^2. \tag{1-5}$$

In addition to the tutorial material presented, we provide a detailed discusion of Newton algorithms that use an exact Hessian matrix for each of the VPF's. All algorithms presented in this report are given in terms of QR-based orthogonal factorizations of matrices and are implementable directly from the material presented here.

The performance of these new ML algorithms is examined in a companion report [4], and is there compared to previously published estimators. The focus here, however, is on providing the theoretical mathematical concepts on which the algorithms are based, and on providing a detailed discussion of the algorithm implementations.

As a vehicle of discussion for introducing the concepts and algorithms, we will refer to a specific model, which is a signal consisting of a sum of complex sinusoids, also expressable as a sum of undamped or damped sinusoids. In this model, the nonlinear parameters of the semilinear model correspond to the signal poles, and the linear parameters are the amplitudes of each sinusoid.

In estimating the parameters of the exponential signal, the matrix $\mathbf{B}(\underline{\theta})$ is a convolution matrix, which corresponds to the prediction filter that annihilates the ideal signal. Since this matrix is a linear function of the prediction coefficients, it allows for significant computational savings in the algorithms. Because of the functional relationship of $\mathbf{B}$ to the prediction coefficients, we will write $\mathbf{B}(\underline{b})$, where $\underline{b}$ is a vector containing the prediction coefficients. The nonlinear parameters in $\underline{\theta}$ can then be obtained as the roots of the polynomial whose coefficients are the prediction coefficients.

To distinguish between the two VPF's throughout this report, we will refer to the VPF containing the column space projector as the signal basis VPF, while we will refer to the VPF containing the row space projector as the prediction filter VPF. In the main body of this report, we develop two parallel paths. One describes theory and algorithms for obtaining estimates of the signal poles directly via minimization of a signal basis VPF. The other describes how to obtain pole estimates indirectly by first providing estimates of the prediction filter coefficients via minimization of a prediction filter VPF and then transforming the prediction coefficients into pole estimates. In both cases, the amplitudes are obtained by solving a linear least-squares problem in which the basis function matrix is constructed from the ML pole estimates.

1.1 New Material

While the application of variable projection techniques to estimating the parameters for exponential signals in noise is not new, several aspects of the

problem as considered here have not been encountered in the literature. Foremost, we here extend existing variable projection methods by deriving the Hessian matrix for the general VPF and by providing implementations of Newton's method for each of the two parallel paths.

Furthermore, we constrain the exponential signal model to include known poles. In each of the two parallel paths, these constraints are introduced differently. In the direct pole estimation path, we use a technique which is similar to deflation in the eigenvector problem; i.e., we project the signal model and the observation vector onto the orthogonal complement of the subspace spanned by the basis functions that correspond to the known poles. The method of introducing the constraint for this VPF is general and not limited to the case of an exponential signal.

In the prediction coefficient estimation path, we factor the prediction filter convolution matrix into two convolution matrices, one corresponding to the known poles and one corresponding to the unknown poles. Since this factorization is facilitated by the nature of the convolution matrix (and thus the exponential signal), this constraint is not applicable to the general model.

1.2 Report Organization

In Chapters 2 and 3, we introduce the notation that will be followed throughout the report. Chapter 2 contains the signal model and a discussion of the ML estimator for this signal in white noise. Chapter 3 provides a numerical link between the ML error norm and the two different variable projection error norms, which are the Signal Basis Variable Projection Functional and the Prediction Filter Variable Projection Functional.

In Chapter 4, we introduce the pole constraints discussed above. This is followed, in Chapter 5, by derivations of the Jacobian matrix, gradient vector, and Hessian matrix for each of the two variable projection error norms. In Chapter 6, we discuss orthogonal factorization techniques that are

key components in the nonlinear optimization schemes. Finally, in Chapter 7, we examine the implementations of the various optimization techniques, based on the results of Chapters 5 and 6.

In addition to material listed above, we provide fairly extensive tutorial material in a number of appendices. In the first of these, we examine key aspects of Linear Least-Squares (LLS) theory (such as projection operators and pseudoinverses) and describe how some of the popular orthogonal factorizations are used in solving LLS problems. This material is included not only as background for this report but also as a mathematical base for some of the nonmaximum likelihood algorithms discussed in the companion report [4]. In Appendix B, we go a layer deeper in the LLS discussion by examining the use of Householder reflectors as a method of implementing the orthogonal factorizations. In Appendix C, we discuss optimization techniques which are useful in minimizing least-squares and variable projection functionals.

In Appendices D, E, and F, we focus on the variable projection material, parallelling and, hopefully, enhancing the discussions of Golub and Pereyra [1],][5]. Appendix D contains derivations for the derivatives of the projection operators and pseudoinverse necessary to calculate the Jacobian, gradiant, and Hessian. Appendix E discusses the simplified tensor notation introduced by Golub and Pereyra, which centers around the Frechet derviative of a mapping. The drawbacks of this notation in computing second derivatives are discussed. We then expound upon a key proof from [1] that ties the variable projection method to the straightforward nonlinear least-squares method, and hence to the ML method.

(Blank Page)

2.0 BACKGROUND

In this chapter we introduce the signal model to be used as a vehicle for discussing the variable projection concepts and algorithms. In particular, the signal model will be the response of a system to a stepped sinusoid input, where the system can be characterized by a linear constant coefficient ordinary differential equation. This will include steady-state terms (whose frequency corresponds to the excitation) and several decaying terms (whose damping and frequency correspond to the system poles). Following the introduction of the signal model, we will then briefly discuss the maximum likelihood estimator. Specifically, we will show that for signals in white Gaussian noise, maximum likelihood estimation simplifies to least-squares estimation.

2.1 Signal Model

We consider the stepped sine response of a linear system modeled as a sum of real exponentials in additive white Gaussian noise,

$$y_n = x_n(\underline{\theta},\underline{a}) + \epsilon_n, \tag{2-1}$$

where

$$\begin{aligned} x_n(\underline{\theta},\underline{a}) = {} & C_0 + C_1 \cos(2\pi f_1 nT) + S_1 \sin(2\pi f_1 nT) \\ & + C_2 \exp(\alpha_2 nT) \cos(2\pi f_2 nT) \\ & + S_2 \exp(\alpha_2 nT) \sin(2\pi f_2 nT) \\ & \bullet \\ & \bullet \\ & \bullet \\ & + C_p \exp(\alpha_p nT) \cos(2\pi f_p nT) \\ & + S_p \exp(\alpha_p nT) \sin(2\pi f_p nT) \ , \end{aligned}$$

$$\underline{\theta} = \left[\alpha_2 \ , \ f_2 \ , \ \cdots \ , \ \alpha_p \ , \ f_p \right]^T ,$$

$$\underline{a} = \left[C_0 \ , \ C_1 \ , \ S_1 \ , \ \cdots \ , \ C_p \ , \ S_p \right]^T .$$

Here T is sampling interval and ϵ_n is a sequence of white Gaussian noise. Here, f_1 is the known excitation frequency, and the term C_0 is included to account for the DC biases introduced during the measurement process.

We can write the signal in vector form so that

$$\underline{Y} = \underline{x}(\underline{\theta},\underline{a}) + \underline{\epsilon}, \tag{2-2}$$

where $$\underline{x} = \left[x_0 , x_1 , \cdots , x_{N-1} \right]^T ,$$

$$\underline{Y} = \left[y_0 , y_1 , \cdots , y_{N-1} \right]^T ,$$

and $$\mathrm{cov}(\underline{\epsilon}) = E\left\{ \underline{\epsilon}\,\underline{\epsilon}^T \right\} = \sigma_\epsilon^2 \, I_N ,$$

where σ_ϵ^2 is the noise variance, and I_N is an NxN identity matrix. The signal basis functions for our signal model are

$$f_{n,1}(\underline{\theta}) = 1$$

$$f_{n,2}(\underline{\theta}) = \cos(2\pi f_1 nT)$$

$$f_{n,3}(\underline{\theta}) = \sin(2\pi f_1 nT)$$

$$f_{n,4}(\underline{\theta}) = \exp(a_2 nT)\, \cos(2\pi f_2 nT)$$

$$f_{n,5}(\underline{\theta}) = \exp(a_2 nT)\, \sin(2\pi f_2 nT)$$

$$\vdots$$

$$f_{n,M-1}(\underline{\theta}) = \exp(a_p nT)\, \cos(2\pi f_p nT)$$

$$f_{n,M}(\underline{\theta}) = \exp(a_p nT)\, \sin(2\pi f_p nT) ,$$

where $M = 2p+1$ is the number of independent basis functions required to span the ideal signal space. We can now define a set of spanning signal basis vectors, given as

$$\underline{f}_j(\underline{\theta}) = \left[f_{0,j}(\underline{\theta}) , f_{1,j}(\underline{\theta}) , \cdots , f_{N-1,j}(\underline{\theta}) \right]^T , \quad j = 1,\ldots,M .$$

Finally, we define the signal basis function matrix

$$\mathbf{F}(\underline{\theta}) = \left[\underline{f}_1(\underline{\theta}) , \underline{f}_2(\underline{\theta}) , \cdots , \underline{f}_M(\underline{\theta}) \right] ,$$

whose columns each correspond to one of the basis functions and whose rows each correspond to an instant of time in the observation window. We can now write $\underline{x}(\underline{\theta},\underline{a})$ in the separable form (as the matrix-vector product)

$$\underline{x}(\underline{\theta},\underline{a}) = \mathbf{F}(\underline{\theta})\ \underline{a} . \tag{2-3}$$

Our goal is to estimate the parameter vectors $\underline{\theta}$ and $\underline{a}$.

2.2 Maximum Likelihood Estimation for Signals in White Noise

In maximum likelihood estimation, we wish to find the parameters that maximize the probability density of the observed data given the unknown parameters. Letting $\underline{\theta}$ and $\underline{a}$ be the collection of unknown parameters, we wish to find $\underline{\theta}^*$ and $\underline{a}^*$ which maximize the conditional density function (the likelihood function)

$$\underline{\theta}^*,\underline{a}^* = \arg \min_{\underline{\theta},\underline{a}} p(y_0,y_1,\cdots,y_{N-1}|\underline{\theta},\underline{a}) .$$

For independently and identically distributed Gaussian noise, the likelihood function is

$$p(y_0,y_1,\cdots,y_{N-1}|\underline{\theta},\underline{a}) = \frac{1}{\left(2\pi\sigma_\epsilon^2\right)^{N/2}} \exp\left\{ -\frac{1}{2\sigma_\epsilon^2} \sum_{n=0}^{N-1} \left[y_n - x_n(\underline{\theta},\underline{a}) \right]^2 \right\}. \tag{2-4}$$

Since the logarithm is a monotonic function, maximizing the logarithm of the likelihood function yields the same result as maximizing the likelihood function itself.

We may therefore optimize the log likelihood function as

$$L(\underline{\theta},\underline{a}) = \ln \left\{ p(y_0, y_1, \cdots, y_{N-1}|\underline{\theta},\underline{a}) \right\}$$

$$= -\frac{N}{2} \ln(2\pi) - N \ln(\sigma_\epsilon) - \frac{1}{2\sigma_\epsilon^2} \sum_{n=0}^{N-1} \left[y_n - x_n(\underline{\theta},\underline{a}) \right]^2 . \qquad (2\text{-}5)$$

Only the last of the terms in Eq. (2-5) contains $\underline{\theta}$ and $\underline{a}$, and it appears in the expression with a negative sign. Therefore, since the noise variance is assumed to be known, the parameters that maximize the likelihood function are those which minimize

$$\sum_{n=0}^{N-1} \left[y_n - x_n(\underline{\theta},\underline{a}) \right]^2 ,$$

which is just the least-squares functional

$$\chi(\underline{\theta},\underline{a}) = \left|\left| \underline{y} - \underline{x}(\underline{\theta},\underline{a}) \right|\right|^2$$

$$= \left|\left| \underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a} \right|\right|^2 . \qquad (2\text{-}6)$$

3.0 VARIABLE PROJECTION ERROR NORMS

As we have just seen, the Maximum Likelihood (ML) estimator reduces to the Least-Squares (LS) estimator when the ideal signal is corrupted with additive white Gaussian noise. Here, we obtain further simplifications by noting the structure of the ideal signal and introduce two modified error norms, or variable projection functionals.

We begin by stating an observation by Golub and Pereyra [1] concerning separable least-squares problems, for which one can first optimize a reduced set of parameters and then find the remaining parameters as a function of the first set. This leads to the basis function Variable Projection Functional (VPF) which, in our case, allows us to find ML estimates of the signal poles independently of the weighting coefficients (amplitudes).

From the signal basis VPF, we follow Kumaresan, Scharf, and Shaw [6], and Bressler and Macovski [7], and introduce the deterministic functional relationship between the signal poles and the prediction filter coefficients for the filter that annihilates the ideal signal. We can then define a VPF based on the prediction filter coefficients and show that this prediction filter VPF is equivalent to the basis function VPF, allowing us to optimize with respect to the prediction coefficients.

3.1 Signal Basis Variable Projection Functional

As shown in the previous chapter, ML estimation of parameters $\underline{\theta}$ and $\underline{a}$ can be achieved by minimizing the nonlinear least-squares error norm; i.e., by finding

$$\begin{aligned} \underline{\theta}^*, \underline{a}^* &= \arg \min_{\underline{\theta}, \underline{a}} \chi(\underline{\theta}, \underline{a}) \\ &= \arg \min_{\underline{\theta}, \underline{a}} \left|\left| \underline{e}(\underline{\theta}, \underline{a}) \right|\right|^2 , \end{aligned} \tag{3-1}$$

where the error vector $\underline{e}(\underline{\theta}, \underline{a})$ is defined as $\underline{e}(\underline{\theta}, \underline{a}) = \underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a}$.

One approach to finding the optimal $\underline{\theta}$ and $\underline{a}$ consists of a three-step process:

1. Minimize with respect to $\underline{a}$ in order to obtain the optimal function

$$\underline{a}(\underline{\theta}) = \arg \min_{\underline{a}} \chi(\underline{\theta},\underline{a}). \tag{3-2}$$

2. Minimize

$$\psi(\underline{\theta}) = \chi\{\underline{\theta},\underline{a}(\underline{\theta})\} \tag{3-3}$$

with respect to $\underline{\theta}$ to yield

$$\underline{\theta}^* = \arg \min_{\underline{\theta}} \psi(\underline{\theta}). \tag{3-4}$$

3. Calculate the optimal numerical values for $\underline{a}$ as

$$\underline{a}^* = \underline{a}(\underline{\theta}^*). \tag{3-5}$$

It has been shown by Golub and Pereyra [1] that, for separable signal models, this multi-step optimization process yields the same values of $\underline{\theta}^*$ and $\underline{a}^*$ as does simultaneous optimization of $\chi(\underline{\theta},\underline{a})$ in both $\underline{\theta}$ and $\underline{a}$. In Appendix F, we summarize their proof.

We now obtain $\underline{a}(\underline{\theta})$ by differentiating the error functional $\chi(\underline{\theta},\underline{a})$ with respect to $\underline{a}$ and equating the resulting expression to zero.

$$\frac{\partial}{\partial \underline{a}} \chi(\underline{\theta},\underline{a}) = \frac{\partial}{\partial \underline{a}} \left|\left| \underline{e}(\underline{\theta},\underline{a}) \right|\right|^2$$

$$= \frac{\partial}{\partial \underline{a}} \left\{ \left[\underline{e}(\underline{\theta},\underline{a}) \right]^T \underline{e}(\underline{\theta},\underline{a}) \right\}$$

$$= 2 \frac{\partial}{\partial \underline{a}} \left\{ \left[\underline{e}(\underline{\theta},\underline{a}) \right]^T \right\} \underline{e}(\underline{\theta},\underline{a})$$

$$= 2 \frac{\partial}{\partial \underline{a}} \left\{ \left[\underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a} \right]^T \right\} \left[\underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a} \right]$$

$$= - 2\ \mathbf{F}^T(\underline{\theta}) \left[\underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a} \right] \qquad (3\text{-}6)$$

Equating this to zero yields

$$\mathbf{F}^T(\underline{\theta})\ \mathbf{F}(\underline{\theta})\ \underline{a} = \mathbf{F}^T(\underline{\theta})\ \underline{y}\ , \qquad (3\text{-}7)$$

which has the solution

$$\underline{a}(\underline{\theta}) = \left[\mathbf{F}^T(\underline{\theta})\ \mathbf{F}(\underline{\theta}) \right]^{-1} \mathbf{F}^T(\underline{\theta})\ \underline{y} \qquad (3\text{-}8)$$

and

$$\underline{a}(\underline{\theta}) = \mathbf{F}^+(\underline{\theta})\ \underline{y}\ , \qquad (3\text{-}9)$$

where $\mathbf{F}^+(\underline{\theta})$ is the pseudoinverse of $\mathbf{F}(\underline{\theta})$. Therefore, $\psi(\underline{\theta})$ can be written as

$$\psi(\underline{\theta}) = \left|\left| \underline{y} - \mathbf{F}(\underline{\theta})\ \mathbf{F}^+(\underline{\theta})\ \underline{y} \right|\right|^2. \qquad (3\text{-}10)$$

Noting that this is an expression for the projection of $\underline{y}$ onto the orthogonal complement of the range (or column space) of $\mathbf{F}$, it is an expression of the variable projection functional

$$\psi(\underline{\theta}) = \left|\left| \mathbf{P}_{\mathbf{F}}^{\perp}(\underline{\theta})\ \underline{y} \right|\right|^2. \qquad (3\text{-}11)$$

3.2 Prediction Filter Variable Projection Functional

Given the s-plane poles for an exponential signal of order M; i.e.,

$$s_i = \alpha_i + j2\pi f_i, \; i=1,2,\ldots,M;$$

then the z-plane poles are calculated as

$$z_i = \exp\{s_i T\}, \; i=1,2,\ldots,M.$$

The z-transform of the ideal signal will thus contain the following polynomial in the denominator

$$B(z) = (z-z_1)(z-z_2)\cdots(z-z_M)$$

$$= z^M + b_1 z^{M-1} + b_2 z^{M-2} + \cdots + b_{M-1} z + b_M \; . \qquad (3\text{-}12)$$

The ideal signal therefore satisfies the homogeneous difference equation

$$x_{i+M} + b_1 \, x_{i+M-1} + b_2 \, x_{i+M-2} + \cdots + b_M \, x_i = 0 \; ; \qquad (3\text{-}13)$$

alternatively written

$$b_M \, x_{i-M} + b_{M-1} \, x_{i-M+1} + \cdots + b_1 \, x_{i-1} + x_i = 0 \; . \qquad (3\text{-}14)$$

We can view the first M terms in this difference equation as forming an estimate (prediction) of the present signal sample based on the last M samples. The coefficents b_i are, for this reason, often called prediction coefficients.

Since the above difference equations represent a convolution between the prediction coefficient vector and the signal vector, we can now define the (N-M)×N convolution matrix

$$\mathbf{B} = \begin{bmatrix} b_M & \cdots & b_2 & b_1 & 1 & & & \\ & b_M & \cdots & b_2 & b_1 & 1 & & 0 \\ & & \ddots & & \ddots & \ddots & \ddots & \\ 0 & & & \ddots & & \ddots & \ddots & \ddots \\ & & & & b_M & \cdots & b_2 & b_1 & 1 \end{bmatrix},$$

such that the rows of **B** annihilate the ideal signal $\underline{x}$, and such that

$$\mathbf{B}\,\underline{x} = 0. \tag{3-15}$$

Also, since each column of the basis function matrix **F** also satisfies the same homogeneous difference equation, **F** is also annihilated by the rows of **B**, and we have

$$\mathbf{B}\,\mathbf{F} = 0. \tag{3-16}$$

Since matrix **B** has full row rank, its rows span an (N-M)-dimensional subspace in $\mathbf{R}^N$. Similarly, since matrix **F** has full column rank, its columns span an M-dimensional subspace in $\mathbf{R}^N$. Because the rows of **B** must be orthogonal to the columns of **F** (as dictated by the equation **BF** = 0), and because the dimensions of the respective subspaces sum to the dimension of the vector space $\mathbf{R}^N$, the row space of **B** and the column space of **F** must be orthogonal complements of each other.

Since the row space of **B** is the same as the orthogonal complement of the column space of **F**, then the projector onto the row space of **B** is the same as the projector onto the orthogonal complement of the column space of **F**; i.e.,

$$_B\mathbf{P} = \mathbf{P}_F^{\perp}, \tag{3-17}$$

and we can define an error norm whose only unknown parameters are the prediction coefficients $\underline{b}$. This error norm is defined as

$$\phi(\underline{b}) = \left|\left| {}_{B}P\, \underline{x} \right|\right|^2. \tag{3-18}$$

Since this error norm follows deterministically from the ML error norm for the signal poles, we may note the invariance principle for the ML estimator and optimize this functional to obtain ML estimates of the prediction coefficients. From here, we can then find the roots of the prediction polynomial and transform these roots from the z-plane to the s-plane to get our ML pole estimates.

4.0 SIGNAL POLE CONSTRAINTS

Since our signal model is to describe the stepped sine response of a linear system, we constrain the model to contain an undamped pole at the known excitation frequency. Also, because we expect there to be some bias introduced into the observed signal during the observation (measurement) process, the model should include a DC pole to offset the bias.

In the case of the basis function VPF, since each pole corresponds to a distinct set of basis functions (a single decaying exponential or a pair of damped or undamped sinusoids), the pole constraints translate into constraints on the individual basis functions. We can introduce these basis function constraints into the optimization process by *deflating* the error space; i.e., by projecting the error vector onto the orthogonal complement of the subspace spanned by the known basis functions. In the first section of this chapter, we develop the theory behind this deflation process and present a modified basis function VPF.

In the case of the prediction filter VPF, we deal only indirectly with the pole parameters, optimizing instead over the prediction polynomial coefficients. To introduce the pole constraints into the coefficient optimization process, we factor the convolution matrix B into two separate convolution matrices: one resulting from the known poles, and one resulting from the unknown poles.

4.1 Deflation of the Least-Squares Error Space

Recall the least-squares error norm for our signal model and noise; i.e.,

$$\chi(\underline{\theta},\underline{a}) = \left|\left| \underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a} \right|\right|^2 .$$

Let us partition the basis function matrix as

$$\mathbf{F}(\underline{\theta}) = \left[\ \mathbf{F}_1 \vdots \mathbf{F}_2(\underline{\theta})\ \right] , \qquad (4\text{-}1)$$

where $\mathbf{F}_1$ contains only the known constant basis functions (and is therefore not a function of $\underline{\theta}$) and $\mathbf{F}_2(\underline{\theta})$ contains the basis functions corresponding to the unknown poles.

If we let M_1 be the number of constant basis functions and M_2 be the number of unknown basis functions, then we can partition the amplitude vector $\underline{a}$ as

$$\underline{a} = \begin{bmatrix} \underline{a}_1 \\ \underline{a}_2 \end{bmatrix} \begin{matrix}) M_1 \\ ) M_2 \end{matrix} . \tag{4-2}$$

The M_1-vector $\underline{a}_1$ contains the weighting coefficients for the known basis functions, and the M_2-vector $\underline{a}_2$ contains the weighting coefficints for the unknown basis functions. Our signal model can then be written as the sum of two matrix vector products; i.e.,

$$\begin{aligned} x(\underline{\theta},\underline{a}) &= \mathbf{F}(\underline{\theta})\ \underline{a} \\ &= \mathbf{F}_1\ \underline{a}_1 + \mathbf{F}_2(\underline{\theta})\ \underline{a}_2 . \end{aligned} \tag{4-3}$$

The estimation error vector is then

$$\underline{e}(\underline{\theta},\underline{a}_1,\underline{a}_2) = \underline{y} - \mathbf{F}_1\ \underline{a}_1 - \mathbf{F}_2(\underline{\theta})\ \underline{a}_2 , \tag{4-4}$$

and the least-squares error norm is

$$\chi(\underline{\theta},\underline{a}_1,\underline{a}_2) = \left|\left|\ \underline{e}(\underline{\theta},\underline{a}_1,\underline{a}_2)\ \right|\right|^2 . \tag{4-5}$$

To take advantage of this signal structure, we first note that--given two orthogonal vectors $\underline{e}_1$ and $\underline{e}_2$--we can decompose a least-squares error norm consisting of the sum of $\underline{e}_1$ and $\underline{e}_2$ as

$$\left|\left|\ \underline{e}_1 + \underline{e}_2\ \right|\right|^2 = \left|\left|\ \underline{e}_1\ \right|\right|^2 + \left|\left|\ \underline{e}_2\ \right|\right|^2 . \tag{4-6}$$

Therefore, given an arbitrary N-vector $\underline{e}$, we can decompose the squared norm of $\underline{e}$ into the sum of two squared norms by projecting $\underline{e}$ onto orthogonal subspaces. Let P_1 be the projector onto the subspace spanned by the columns of the matrix F_1. Then the projector onto the orthogonal complement of this subspace is given by

$$P_1^{\perp} = I - P_1 \ . \tag{4-7}$$

Noting that $I = P_1 + P_1^{\perp}$, we can then write

$$\begin{aligned} \left|\left| \underline{e} \right|\right|^2 &= \left|\left| P_1 \underline{e} + P_1^{\perp} \underline{e} \right|\right|^2 \\ &= \left|\left| P_1 \underline{e} \right|\right|^2 + \left|\left| P_1^{\perp} \underline{e} \right|\right|^2 . \end{aligned} \tag{4-8}$$

Now, recalling the definition of the error vector $\underline{e}$, we get

$$\begin{aligned} \chi(\underline{\theta},\underline{a}_1,\underline{a}_2) &= \left|\left| P_1 \left(\underline{y} - F_1 \underline{a}_1 - F_2(\underline{\theta}) \underline{a}_2 \right) \right|\right|^2 \\ &\quad + \left|\left| P_1^{\perp} \left(\underline{y} - F_1 \underline{a}_1 - F_2(\underline{\theta}) \underline{a}_2 \right) \right|\right|^2 \\ &= \left|\left| P_1 \underline{y} - F_1 \underline{a}_1 - P_1 F_2(\underline{\theta}) \underline{a}_2 \right|\right|^2 \\ &\quad + \left|\left| P_1^{\perp} \underline{y} - P_1^{\perp} F_2(\underline{\theta}) \underline{a}_2 \right|\right|^2 , \end{aligned} \tag{4-9}$$

where we have noted that $P_1 F_1 = f_1$ and $P_1^{\perp} F_1 = 0$. Noting that the second term in the error norm is no longer a function of $\underline{a}_1$, we decompose the norm as

$$\chi(\underline{\theta},\underline{a}_1,\underline{a}_2) = \chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2) + \chi_2(\underline{\theta},\underline{a}_2), \tag{4-10}$$

where $$\chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2) = \left|\left| P_1 \underline{y} - F_1 \underline{a}_1 - P_1 F_2(\underline{\theta}) \underline{a}_2 \right|\right|^2 ,$$

and $$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| P_1^{\perp} \underline{y} - P_1^{\perp} F_2(\underline{\theta}) \underline{a}_2 \right|\right|^2 .$$

Substituting $P_1 = F_1 F_1^{+}$ into the expression for $\chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2)$, we get

$$\chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2) = \left|\left| P_1 \underline{y} - F_1 \underline{a}_1 - F_1 F_1^{+} F_2(\underline{\theta}) \underline{a}_2 \right|\right|^2$$

$$= \left|\left| P_1 \underline{y} - F_1 \left(\underline{a}_1 + F_1^{+} F_2(\underline{\theta}) \underline{a}_2 \right) \right|\right|^2$$

$$= \left|\left| P_1 \underline{y} - F_1 \underline{b} \right|\right|^2 , \qquad (4\text{-}11)$$

where $\underline{b} = \underline{a}_1 + F_1^{+} F_2(\underline{\theta}) \underline{a}_2$. We can minimize $\chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2)$ to zero by solving the <u>consistent</u> system of equations

$$F_1 \underline{b} = P_1 \underline{y} \qquad (4\text{-}12)$$

to yield

$$\underline{b} = F_1^{+} P_1 \underline{y}$$

$$= F_1^{+} F_1 F_1^{+} \underline{y}$$

$$= F_1^{+} \underline{y}. \qquad (4\text{-}13)$$

Here we have noted that $\mathbf{F}_1^+ \mathbf{F}_1 \mathbf{F}_1^+ = \mathbf{F}_1^+$. Clearly, then (given estimates for $\underline{\theta}$ and $\underline{a}_2$), $\chi_1(\underline{\theta},\underline{a}_1,\underline{a}_2)$ can be minimized to zero by letting

$$\underline{a}_1 = \mathbf{F}_1^+ \left[\underline{y} - \mathbf{F}_2(\underline{\theta})\ \underline{a}_2 \right] . \tag{4-14}$$

Given that we can minimize χ_1 to zero for any estimates of $\underline{\theta}$ and $\underline{a}_2$, then minimization of χ reduces to minimization of

$$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| \mathbf{P}_1^{\perp}\ \underline{y} - \mathbf{P}_1^{\perp}\ \mathbf{F}_2(\underline{\theta})\ \underline{a}_2 \right|\right|^2 . \tag{4-15}$$

If we now define

$$\underline{v} = \mathbf{P}_1^{\perp}\ \underline{y} \tag{4-16}$$

and

$$\mathbf{G}(\underline{\theta}) = \mathbf{P}_1^{\perp}\ \mathbf{F}_2(\underline{\theta}) , \tag{4-17}$$

then the least-squares error norm becomes

$$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| \underline{v} - \mathbf{G}(\underline{\theta})\ \underline{a}_2 \right|\right|^2 . \tag{4-18}$$

Following the same arguments as in the developments of the previous chapter, we can now optimize with respect to $\underline{\theta}$ independently of $\underline{a}_2$ by defining a (constrained) VPF as

$$\begin{aligned} \psi_2(\underline{\theta}) &= \left|\left| \underline{v} - \mathbf{G}(\underline{\theta})\ \mathbf{G}^+(\underline{\theta})\ \underline{v} \right|\right|^2 \\ &= \left|\left| \mathbf{P}_{\mathbf{G}}^{\perp}(\underline{\theta})\ \underline{v} \right|\right|^2 . \end{aligned} \tag{4-19}$$

4.2 Factorisation of the Prediction Filter Convolution Matrix

We introduce the contraints into the prediction filter VPF by factoring the convolution matrix. Recall that the denominator of the z-transform of the ideal signal contains the polynomial

$$B(z) = (z-z_1)(z-z_2)\cdots(z-z_M)$$

$$= z^M + b_1 z^{M-1} + b_2 z^{M-2} + \cdots + b_{M-1} z + b_M \,. \qquad (4\text{-}20)$$

Now assume that M_1 of the signal poles is known *a priori* and the remaining $M_2 = M - M_1$ poles are unknown. Then the polynomial $B(z)$ can be factored as

$$B(z) = C(z)\ U(z),$$

where $$C(z) = (z-z_1)(z-z_2)\cdots(z-z_{M_1})$$

$$= z^{M_1} + c_1 z^{M_1-1} + c_2 z^{M_1-2} + \cdots + c_{M_1-1} z + c_{M_1} \qquad (4\text{-}21)$$

and $$U(z) = (z-z_1)(z-z_2)\cdots(z-z_{M_2})$$

$$= z^{M_2} + u_1 z^{M_2-1} + u_2 z^{M_2-2} + \cdots + u_{M_2-1} z + u_{M_2} \qquad (4\text{-}22)$$

If we now define the three coefficent vectors

$$\underline{b} = \left[\, 1 \; b_1 \; b_2 \; \cdots \; b_M \, \right]^T ,$$

$$\underline{c} = \left[\, 1 \; c_1 \; c_2 \; \cdots \; c_{M_1} \, \right]^T ,$$

and

$$\underline{u} = \left[\, 1 \; u_1 \; u_2 \; \cdots \; u_{M_2} \, \right]^T ,$$

then we note that $\underline{b}$ can be obtained as the vector convolution of $\underline{c}$ and $\underline{u}$; i.e.,

$$\underline{b} = \underline{u} * \underline{c} \; . \tag{4-23}$$

The $(N-M)\times N$ convolution matrix **B** can therefore be factored as the product of two convolution matrices: the $(N-M)\times(N-M_1)$ matrix **U** (whose rows contain the elements of vector $\underline{u}$) and the $(N-M_1)\times N$ matrix **C** (whose rows contain the elements of vector $\underline{c}$); i.e.,

$$\mathbf{B} = \mathbf{U} \, \mathbf{C} \; . \tag{4-24}$$

Since the matrix **C** contains coefficients corresponding to constant poles, **C** itself will be constant, so that

$$\frac{\partial \mathbf{C}}{\partial b_i} = \mathbf{0} \quad , \quad i = 1,\ldots,M \; . \tag{4-25}$$

Therefore, we have

$$\frac{\partial \mathbf{B}}{\partial b_i} = \frac{\partial \mathbf{U}}{\partial u_i} \, \mathbf{C} \; , \tag{4-26}$$

where we now differentiate only with respect to the unknown parameters in $\underline{u}$. If we adopt the operator notation

$$D_i(\bullet) = \frac{\partial \bullet}{\partial u_i} , \tag{4-27}$$

Eq. (4-12) then becomes

$$D_i(B) = D_i(U)\ C . \tag{4-28}$$

It should be noted that premultiplication of an arbitrary vector $\underline{w}$ by either matrix **B**, **U**, or **C** represents the full-overlap elements of the convolution of $\underline{w}$ with the vector $\underline{b}$, $\underline{u}$, or $\underline{c}$, respectively. For instance, if $\underline{w}$ is an N-vector, then the convolution $\underline{b}*\underline{w}$ is an (N+M)-vector with 2M edge effect or transient elements and with N-M full-overlap elements corresponding to the matrix-vector product $B\underline{w}$. The products $B^T\underline{w}$, $U^T\underline{w}$, and $C^T\underline{w}$ also represent convolutions, but with both the full-overlap and edge effect elements present and with the elements of $\underline{b}$, $\underline{u}$, and $\underline{c}$ reversed.

5.0 DERIVATIVES OF THE VARIABLE PROJECTION FUNCTIONALS

In this chapter, we derive the partial derivatives; i.e., elements of the Jacobian matrix, gradient vector, and Hessian matrix for the two variable projection functionals. We first parallel Golub and Pereyra [1] in deriving the typical column of the Jacobian matrix and element of the gradient vector of the signal basis VPF. We then extend previous work by introducing the Hessian for this VPF. Following this, we repeat the Jacobian, gradient, and Hessian derivations for the prediction filter VPF.

Throughout this work, our notation is different from that of previous investigators in that we retain the index of differentiation and derive all of the derivatives as partial derivatives. Golub and Pereyra [1], on the other hand, introduced the Frechet derivative of the basis function matrix, which is an array of partial derivative matrices (a simplified view of a valence three tensor), allowing them to derive the gradient without having to resort to full tensor notation. In Appendix E, we examine this simplified tensor notation. When attempting to form the Hessian matrix in this simplified notation, however, one is lead to attempt the multiplication of two three-dimensional arrays, which is undefined in the simplified notation. The problem could be overcome by introducing the full-blown tensor notation, but the alternative developed here seemed simpler and clearer. By retaining the indexing and partial derivatives, only the regular rules of linear algebra need be observed. Furthermore, with the indexing retained, the equations more nearly reflect the computer code necessary for software implementation of the algorithms.

5.1 Derivatives of the Signal Basis Variable Projection Functional

We now introduce the derivatives for the signal basis VPF. The first two subsections are tutorial in that they expound upon the work performed by Golub and Pereyra [1]. These draw upon the derivative of the column space projector described in Appendix D. The third subsection, which introduces the Hessian matrix, presents material that has not been encountered in the literature.

JACOBIAN MATRIX

Recall the definition of the signal basis VPF

$$\Psi(\underline{\theta}) = \left|\left| P_F^{\perp} \underline{Y} \right|\right|^2.$$

The error vector in this case is

$$\underline{e}(\underline{\theta}) = P_F^{\perp} \underline{Y} . \tag{5-1}$$

The typical column of the Jacobian matrix is therefore

$$\begin{aligned} \underline{J}_i &= D_i(\underline{e}) \\ &= D_i\left(P_F^{\perp} \underline{Y} \right) \\ &= - D_i(P_F) \underline{Y} . \end{aligned}$$

Substituting Eq. (D-4) for the partial derivative of the projector, we get the desired expression

$$\begin{aligned} \underline{J}_i &= - \left\{ P_F^{\perp} D_i(F) F^{+} + (F^{+})^T D_i(F^T) P_F^{\perp} \right\} \underline{Y} \\ &= - P_F^{\perp} D_i(F) F^{+} \underline{Y} - (F^{+})^T D_i(F^T) P_F^{\perp} \underline{Y} . \end{aligned} \tag{5-2}$$

GRADIENT VECTOR

We derive an expression for the gradient by noting Eq. (C-9); i.e, $g_i = 2\, \underline{e}^T D_i(\underline{e}) = 2\, \underline{e}^T \underline{J}_i$. Substituting Eqs. (5-1) and (5-2) from above, this becomes

$$g_i = -2 \left[P_F^{\perp} Y \right]^T \left\{ P_F^{\perp} D_i(F) F^+ Y + (F^+)^T D_i(F^T) P_F^{\perp} Y \right\}$$

$$= -2 Y^T P_F^{\perp} D_i(F) F^+ Y - Y^T P_F^{\perp} (F^+)^T D_i(F^T) P_F^{\perp} Y , \qquad (5\text{-}3)$$

where, in the first term, we have noted that the projector is symmetric and idempotent. Now, since

$$P_F^{\perp} (F^+)^T = \left[(F^+) P_F^{\perp} \right]^T$$

$$= \left\{ F^+ \left[I - F F^+ \right] \right\}^T$$

$$= \left[F^+ - F^+ F F^+ \right]^T = 0 ,$$

the second term in Eq. (5-3) vanishes, leaving the desired expression for the i'th element of the gradient of the signal basis VPF, which is

$$g_i = -2 Y^T P_F^{\perp} D_i(F) F^+ Y . \qquad (5\text{-}4)$$

HESSIAN MATRIX

The typical element of the Hessian matrix is obtained by differentiating the i'th element of the gradient with respect to the j'th parameter; i.e., $H_{ij} = D_j(g_i)$. Substituting Eq. (5-4) for the gradient element, we obtain

$$H_{ij} = D_j \left[-2 Y^T P_F^{\perp} D_i(F) F^+ Y \right] ,$$

$$= -2 Y^T D_j \left[P_F^{\perp} D_i(F) F^+ \right] Y .$$

Applying the product rule of differentiation then yields

$$H_{ij} = - 2\, \underline{y}^T D_j\left[P_F^{\perp} \right] D_i(F)\, F^+ \underline{y} \; - 2\, \underline{y}^T P_F^{\perp} D_{ij}^2(F)\, F^+ \underline{y}$$

$$- 2\, \underline{y}^T P_F^{\perp} D_i(F)\, D_j(F^+)\, \underline{y} \; . \qquad (5\text{-}5)$$

Here, $D_{ij}^2(F)$ is the second partial derivative of the basis function matrix with respect to parameters θ_i and θ_j. Substituting Eqs. (D-4) and (D-11) for the derivatives of the projector and the pseudoinverse and noting that

$$D_j\left[P_F^{\perp} \right] = - D_j(P_F) \; ,$$

we obtain

$$H_{ij} = 2\, \underline{y}^T \left[P_F^{\perp} D_j(F)\, F^+ + (F^+)^T D_j(F^T)\, P_F^{\perp} \right] D_i(F)\, F^+ \underline{y}$$

$$- 2\, \underline{y}^T P_F^{\perp} D_{ij}^2(F)\, F^+ \underline{y} - 2\, \underline{y}^T P_F^{\perp} D_i(F) \left[{}_F P^{\perp} D_j(F^T)\, (F^+)^T F^+ \right.$$

$$\left. - F^+ D_j(F)\, F^+ + F^+ (F^+)^T D_j(F^T)\, P_F^{\perp} \right] \underline{y}$$

$$= 2\, \underline{y}^T P_F^{\perp} D_j(F)\, F^+ D_i(F)\, F^+ \underline{y} + 2\, \underline{y}^T (F^+)^T D_j(F^T)\, P_F^{\perp} D_i(F)\, F^+ \underline{y}$$

$$- 2\, \underline{y}^T P_F^{\perp} D_{ij}^2(F)\, F^+ \underline{y} - 2\, \underline{y}^T P_F^{\perp} D_i(F)\, {}_F P^{\perp} D_j(F^T)\, (F^+)^T F^+ \underline{y} \qquad (5\text{-}6)$$

$$+ 2\, \underline{y}^T P_F^{\perp} D_i(F)\, F^+ D_j(F)\, F^+ \underline{y} - 2\, \underline{y}^T P_F^{\perp} D_i(F)\, F^+ (F^+)^T D_j(F^T)\, P_F^{\perp} \underline{y} \; .$$

A very interesting case is when the basis function matrix has full column rank. In this case, since there are M independent columns, there must also be M independent rows, so that the rows of F span the entire M-space in which they lie. The null space of F therefore contains only the zero vector. In

this case, the projector onto the row space is the identity matrix; i.e., ${}_F\mathbf{P} = \mathbf{I}_M$. The projector onto the orthogonal complement of the row space (the projector onto the null space) is therefore the zero matrix; i.e.,

$$ {}_F\mathbf{P}^{\perp} = \mathbf{0} \ . $$

The fourth term in Eq. (5-6), which contains the projector onto the null space of $\mathbf{F}$, therefore vanishes in the full rank case, leaving the following expression:

$$ \begin{aligned} H_{ij} = 2 \left[\mathbf{P}_F^{\perp} \mathbf{Y} \right]^T \left[D_j(\mathbf{F}) \, \mathbf{F}^+ \, D_i(\mathbf{F}) - D_{ij}^2(\mathbf{F}) + D_i(\mathbf{F}) \, \mathbf{F}^+ \, D_j(\mathbf{F}) \right] \left[\mathbf{F}^+ \mathbf{Y} \right] \\ - 2 \left[(\mathbf{F}^+)^T D_i(\mathbf{F}^T) \, \mathbf{P}_F^{\perp} \mathbf{Y} \right]^T \left[(\mathbf{F}^+)^T D_j(\mathbf{F}^T) \, \mathbf{P}_F^{\perp} \mathbf{Y} \right] \\ + 2 \left[\mathbf{P}_F^{\perp} D_j(\mathbf{F}) \, \mathbf{F}^+ \mathbf{Y} \right]^T \left[\mathbf{P}_F^{\perp} D_i(\mathbf{F}) \, \mathbf{F}^+ \mathbf{Y} \right] . \end{aligned} \qquad (5\text{-}7) $$

In the stepped sine response signal modeling problem, the rank of the basis function matrix is determined prior to the pole optimization. The basis function matrix can therefore be assumed full rank during the optimization process, and Eq. (5-7) is the applicable equation for the Hessian. Chapter 7 contains an implementation of Eq. (5-7) based on the QR factorization of the basis function matrix.

5.2 Derivatives of the Prediction Filter Variable Projection Functional

We now present the derivatives of the prediction filter VPF. While these developments parallel those given in the previous section, they have not been encountered in the literature in their present form. Here, again, we first derive the Jacobian matrix by noting the partial derivative of the row space projector given in Appendix D. We then derive the gradient vector, and by differentiating the gradient, we derive the Hessian matrix.

JACOBIAN MATRIX

The definition of the prediction filter VPF is

$$\psi(\underline{b}) = \left|\left| \; {}_{B}P \; \underline{y} \; \right|\right|^2 ,$$

so that the error vector is

$$\underline{e}(\underline{b}) = {}_{B}P \; \underline{y} \; . \tag{5-8}$$

The typical column of the Jacobian matrix is therefore $\underline{J}_i = D_i)(\underline{e}) = D_i({}_{B}P)\underline{y}$. Substituting Eq. (D-5) for the partial derivative of the projector, we get the desired expression

$$\underline{J}_i = \left\{ B^+ \, D_i(B) \; {}_{B}P^{\perp} + \left[B^+ \, D_i(B) \; {}_{B}P^{\perp} \right]^T \right\} \underline{y}$$

$$= B^+ \, D_i(B) \; {}_{B}P^{\perp} \; \underline{y} \; + {}_{B}P^{\perp} \, D_i(B^T) \; (B^+)^T \, \underline{y} \; . \tag{5-9}$$

GRADIENT VECTOR

To derive an expression for the gradient, we again note Eq. (C-9); i.e, $g_i = 2 \, \underline{e}^T \, D_i(\underline{e}) = 2 \, \underline{e}^T \, \underline{J}_i$. Substituting Eqs. (5-8) and (5-9) from above, this becomes

$$g_i = 2 \left[\; {}_{B}P \; \underline{y} \right]^T \left\{ B^+ \, D_i(B) \; {}_{B}P^{\perp} + \left[B^+ \, D_i(B) \; {}_{B}P^{\perp} \right]^T \right\} \underline{y}$$

$$= 2 \, \underline{y}^T \, {}_{B}P \;\; B^+ \, D_i(B) \; {}_{B}P^{\perp} \, \underline{y} \; + 2 \, \underline{y}^T \, {}_{B}P \; {}_{B}P^{\perp} \, D_i(B^T) \; (B^+)^T \, \underline{y} \; .$$

The two projectors in the second term annihilate each other; i.e.,

$${}_{B}P \; {}_{B}P^{\perp} = 0 \; .$$

Also, the columns of the pseudoinverse of **B** span the same space as that spanned by the rows of **B**, so that the pseudoinverse is unaffected by the projector onto the row space. This becomes evident by writing

$$
\begin{aligned}
{}_{B}\mathbf{P}\,\mathbf{B}^{+} &= \mathbf{B}^{+}\,\mathbf{B}\,\mathbf{B}^{+} \\
&= \mathbf{B}^{+}.
\end{aligned}
$$

Noting the above remarks, we are left with the desired expression,

$$
g_i = 2\,\underline{Y}^{T}\,\mathbf{B}^{+}\,D_i(\mathbf{B})\,{}_{B}\mathbf{P}^{\perp}\,\underline{Y}\,. \tag{5-10}
$$

HESSIAN MATRIX

Proceeding as before, we obtain an expression for the typical element of the Hessian matrix by differentiating the i'th element of the gradient vector $H_{ij} = D_j(g_i)$. Substituting Eq. (5-10) for the gradient, we obtain

$$
\begin{aligned}
H_{ij} &= D_j\left[\, 2\,\underline{Y}^{T}\,\mathbf{B}^{+}\,D_i(\mathbf{B})\,{}_{B}\mathbf{P}^{\perp}\,\underline{Y}\,\right] \\
&= 2\,\underline{Y}^{T}\,D_j\left[\,\mathbf{B}^{+}\,D_i(\mathbf{B})\,{}_{B}\mathbf{P}\,\right]^{\perp}\underline{Y}\,.
\end{aligned}
$$

Applying the product rule of differentiation then yields

$$
\begin{aligned}
H_{ij} = {} & 2\,\underline{Y}^{T}\,D_j(\mathbf{B}^{+})\,D_i(\mathbf{B})\,{}_{B}\mathbf{P}^{\perp}\,\underline{Y} + 2\,\underline{Y}^{T}\,\mathbf{B}^{+}\,D_{ij}^{2}(\mathbf{B})\,{}_{B}\mathbf{P}^{\perp}\,\underline{Y} \\
& + 2\,\underline{Y}^{T}\,\mathbf{B}^{+}\,D_i(\mathbf{B})\,D_j\left[\,{}_{B}\mathbf{P}^{\perp}\,\right]\underline{Y}\,.
\end{aligned} \tag{5-11}
$$

Here, $D_{ij}^{2}(\mathbf{B})$ is the second partial derivative of the prediction filter convolution matrix with respect to parameters b_i and b_j. Since the elements of the convolution matrix are the prediction coefficients themselves, the first derivatives will have ones as the only nonzero elements. The second

derivative of the convolution matrix, and thus the second term in Eq. (5-11), therefore vanishes. Substituting Eqs. (D-5) and (D-11) for the derivatives of the projector and the pseudoinverse, and noting that

$$D_j\left[{}_B P^{\perp} \right] = - D_j({}_B P) \ ,$$

we obtain

$$\begin{aligned} H_{ij} = 2\, \underline{Y}^T \Big[& {}_B P^{\perp} D_j(B^T) \, (B^+)^T B^+ - B^+ D_j(B) \, B^+ \\ & + B^+ \, (B^+)^T D_j(B^T) \, P_B^{\perp} \Big] \, D_i(B) \, {}_B P^{\perp} \, \underline{Y} \\ & - 2\, \underline{Y}^T B^+ D_i(B) \left[B^+ D_j(B) \, {}_B P^{\perp} + {}_B P^{\perp} D_j(B^T) \, (B^+)^T \right] \underline{Y} \\ = \ . \ & \underline{Y}^T \, {}_B P^{\perp} D_j(B^T) \, (B^+)^T B^+ D_i(B) \, {}_B P^{\perp} \, \underline{Y} - 2\, \underline{Y}^T B^+ D_j(B) \, B^+ D_i(B) \, {}_B P^{\perp} \, \underline{Y} \\ & + 2\, \underline{Y}^T B^+ \, (B^+)^T D_j(B^T) \, P_B^{\perp} \, D_i(B) \, {}_B P^{\perp} \, \underline{Y} \\ & - 2\, \underline{Y}^T B^+ D_i(B) \, B^+ D_j(B) \, {}_B P^{\perp} \, \underline{Y} - 2\, \underline{Y}^T B^+ D_i(B) \, {}_B P^{\perp} D_j(B^T) \, (B^+)^T \, \underline{Y} \ . \end{aligned}$$

(5-12)

Due to the structure of the convolution matrix, its rows are necessarily independent; i.e., **B** has full row rank. Since there are (N-M) independent rows, there must be (N-M) independent columns, so that the columns of **B** span the entire (N-M)-space in which they lie. The projector onto the column space is therefore the identity matrix; i.e., $P_B = I$. The projector onto the orthogonal complement of the column space is therefore the zero matrix; i.e., $P_B^{\perp} = 0$. The third term in Eq. (5-12), which contains the projector onto the orthogonal complement of the column space of **B**, therefore vanishes in this case, leaving the following expression.

$$H_{ij} = -2\left[(B^{+})^{T}\,Y\right]^{T}\left[D_j(B)\,B^{+}\,D_i(B) + D_i(B)\,B^{+}\,D_j(B)\right]\left[{}_{B}P^{\perp}\,Y\right]$$

$$+ 2\left[B^{+}\,D_j(B)\,{}_{B}P^{\perp}\,Y\right]^{T}\left[B^{+}\,D_i(B)\,{}_{B}P^{\perp}\,Y\right]$$

$$- 2\left[{}_{B}P^{\perp}\,D_i(B^{T})\,(B^{+})^{T}\,Y\right]^{T}\left[{}_{B}P^{\perp}\,D_j(B^{T})\,(B^{+})^{T}\,Y\right]. \qquad (5\text{-}13)$$

Chapter 7 contains an implementation of Eq. (5-13) based on what we call an RV factorization of the convolution matrix. This factorization, as well as the QR factorization mentioned in Section 5.3, will be the subject of the next chapter.

(Blank Page)

6.0 ORTHOGONAL FACTORIZATIONS FOR NONLINEAR OPTIMIZATION

In this chapter, we consider two versions of the QR factorization, which will be utilized in effecting the pseudoinverse and projection operators necessary for the optimization strategies. The first version is one in which we transform the basis function matrix from the left such that $\mathbf{Q}\,\mathbf{F} = \mathbf{R}$, where $\mathbf{Q}$ is orthogonal and $\mathbf{R}$ is upper triangular. We will refer to this factorization as the QR factorization, even though, strictly speaking, it is the $\mathbf{Q}^T\mathbf{R}$ factorization of matrix $\mathbf{F}$.

In the second case, we transform the convolution matrix from the right such that $\mathbf{B}\,\mathbf{V} = \mathbf{R}$, where $\mathbf{V}$ is orthogonal and $\mathbf{R}$ is now lower triangular. We will refer to this as the RV factorization. In achieving this factorization, we will take advantage of the special banded structure of the convolution matrix.

6.1 QR Factorization of the Basis Function Matrix

We facilitate the formation of the VPF and its derivatives by factoring the full rank NXM basis function matrix $\mathbf{F}$ as

$$\mathbf{Q}\,\mathbf{F} = \mathbf{R} \equiv \begin{bmatrix} \mathbf{R}_1 \\ \hline \mathbf{0} \end{bmatrix}, \tag{6-1}$$

where $\mathbf{Q}$ is an orthogonal NXN matrix and

$$\mathbf{R}_1 = \begin{bmatrix} \diagdown \\ \mathbf{0} \quad \diagdown \end{bmatrix}_{M\times M}$$

is square, upper triangular, and nonsingular. We may then write $\mathbf{F}$ as

$$\mathbf{F} = \mathbf{Q}^T\,\mathbf{R}. \tag{6-2}$$

The pseudoinverse of F can now be defined as

$$F^{+} = R^{+} Q, \tag{6-3}$$

where

$$R^{+} = \left[\begin{array}{c|c} R_1^{-1} & 0 \end{array} \right] . \tag{6-4}$$

As is shown in Appendix A, the projector onto the column space of the basis function matrix can be formed from this factorization as

$$P_F = Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array} \right] Q , \tag{6-5}$$

where I_M is the MXM identity matrix. The projector onto the orthogonal complement of the column space of F is then

$$P_F^{\perp} = I_N - P_F$$

$$= Q^T \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & I_{N-M} \end{array} \right] Q . \tag{6-6}$$

As described in Appendix F, this factorization is achieved by applying successive Householder reflectors from the left of F so that $Q = H_M \cdots H_2 H_1$, where each H_i represents an elementary matrix containing a Householder reflector. Here we do not explicitly form the matrix Q or any of the H_i. Instead, we retain the information necessary to reconstruct the Householder reflectors when we need to apply the transformations.

6.2 RV Factorisation of the Convolution Matrix

Here we describe an efficient method for factoring the (N-M)XN prediction filter convolution matrix B. Since B is already upper trapezoidal, we can apply Householder reflectors from the right to transform B into a lower triangular matrix. This is essentially the same factorization used in the

complete orthogonal factorization for rank deficient matrices. The **RV** factorization results in the following:

$$\mathbf{B}\,\mathbf{V} = \mathbf{R} \equiv \left[\, \mathbf{R}_1 \;\middle|\; \mathbf{0} \,\right], \tag{6-7}$$

where $\mathbf{R}_1 = \left[\begin{array}{c} \diagdown \\ \diagdown \end{array}\!\!\begin{array}{c} \mathbf{0} \\ \diagdown \end{array}\right]_{(N-M)\times(N-M)}$

is square, lower triangular, and nonsingular. With this, we may write

$$\mathbf{B} = \mathbf{R}\,\mathbf{V}^T, \tag{6-8}$$

and define the pseudoinverse of **B** as

$$\mathbf{B}^+ = \mathbf{V}\,\mathbf{R}^+, \tag{6-9}$$

where

$$\mathbf{R}^+ = \left[\begin{array}{c} \mathbf{R}_1^{-1} \\ \hline \mathbf{0} \end{array}\right]. \tag{6-10}$$

Given the **RV** transformation described above, the projection operators onto the row space of **B** and onto the orthogonal complement of the row space are defined as

$$\begin{aligned} {}_B\mathbf{P} &= \mathbf{B}^+\,\mathbf{B} \\ &= \mathbf{V}\,\mathbf{R}^+\,\mathbf{R}\,\mathbf{V}^T \\ &= \mathbf{V}\left[\begin{array}{c|c} \mathbf{I}_{N-M} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right]\mathbf{V}^T, \end{aligned} \tag{6-11}$$

and

$$\begin{aligned} {}_B\mathbf{P}^{\perp} &= \mathbf{I}_N - {}_B\mathbf{P} \\ &= \mathbf{V}\left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I}_M \end{array}\right]\mathbf{V}^T. \end{aligned} \tag{6-12}$$

Representing each Householder reflector by the elementary transformation matrix H_i, the orthogonal matrix V can be written as the product of the individual elementary transformations; i.e., as $V = H_1 H_2 \cdots H_{N-M}$. This is an important representation of V since, when applying the orthogonal transformation represented by V, we in fact apply the sequence of elementary transformations represented by the H_i. Furthermore, we do not form any of the H_i explicitly but retain the minimal amount of information necessary to reconstruct these transformations as needed.

The order of the H_i in the above expression is important because while we form V as a sequence of transformations operating from the right of a matrix (as a postmultiplication, operating on a row vector), we generally apply the transformations to other vectors from the left (as a premultiplication, operating on a column vector). As is shown explicitly by writing $V \underline{y} = H_1 H_2 \cdots H_{N-M} \underline{y}$, changing from postmultiplication to premultiplication requires that the Householder transformations be applied in the reverse order.

The banded nature of the convolution matrix allows for computational savings during the orthogonal factorization process. Since each row starts out with exactly M nonzero elements to the right of the diagonal, and since these rows retain the zeros throughout the transformation process, each Householder reflector is constructed from an (M+1)-vector as opposed to an N-vector. Since N is usually much greater than M, this can amount to considerable savings.

Furthermore, each Householder reflector effects only the M rows immediately following the row from which it was constructed. For example, consider the i'th reflector. If we were to continue applying the reflector beyond the (i+M-1)'th row, its effects would be squandered on the zero elements contained below the diagonal; i.e., upon reaching these later rows, it would be transforming an (M+1)-dimensional vector of zeros. The net effect is to reduce the number of rows that must be transformed by each reflector from (N-M-i) to M.

7.0 CONSTRAINED MAXIMUM LIKELIHOOD ALGORITHMS

In this chapter, we describe how to implement the kernel elements of the constrained maximum likelihood algorithms, which utilize the methods of Gauss, Newton, and steepest descent. In particular, we show how to construct the Jacobian matrix, gradient vector, and Hessian matrix for each of the constrained VPF's.

First we present these derivatives for the constrained signal basis VPF, given in Eq. (4-18) as

$$\phi_2(\underline{\theta}) = \left|\left| \; P_G^{\perp}(\underline{\theta}) \; \underline{y} \; \right|\right|^2 \quad .$$

Prior to presenting these derivatives, however, we will discuss an economical way of introducing the pole constraints; i.e., of deflating the least-squares error space. This method will take advantage of the structure of the projection operator when written in terms of the orthogonal matrix from the QR factorization.

We then will implement the derivatives of the constrained prediction filter VPF,

$$\phi_2(\underline{u}) = \left|\left| \; {}_B P(\underline{u}) \; \underline{y} \; \right|\right|^2 .$$

Here, the vector $\underline{u}$ contains the prediction coefficients corresponding to the unknown poles. These coefficients are the elements of the matrix U, which results from the factorization of the convolution matrix given in Eq. (4-23) as $B = U\,C$.

7.1 Optimization of Exponential Poles

We now develop algorithms for minimizing the constrained signal basis VPF. After describing the implementation of the pole constraints, we describe the formation of the Jacobian matrix and note a simplified Gauss-Newton algorithm

introduced by Kaufman [8]. We then discuss the the gradient vector and the Hessian matrix, which are necessary for implementing the Newton algorithm.

IMPLEMENTATION OF THE POLE CONSTRAINTS

In this subsection, we describe a modified version of the constrained basis function matrix $\mathbb{G} = \mathbf{P}_1^{\perp} \mathbf{F}_2(\underline{\theta})$, and the constrained observation vector $\underline{y} = \mathbf{P}_1^{\perp} \underline{y}$. Here, $\mathbf{P}_1^{\perp}$ is the projector onto the orthogonal complement of the column space of the matrix $\mathbf{F}_1$ (which contains the basis functions corresponding to the known poles) and $\mathbf{F}_2$ is the matrix of estimated basis functions. Now recall the constrained least-squares functional, given in Eq. (4-6) as

$$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| \mathbf{P}_1^{\perp} \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \right|\right|^2 .$$

If we perform the factorization

$$\mathbf{W}_1 \, \mathbf{F}_1 = \tilde{\mathbf{R}} \, , \tag{7-1}$$

where $\mathbf{W}_1$ is an orthogonal matrix and $\tilde{\mathbf{R}}$ is an $N \times M_1$ upper triangular matrix, then we can form the projector

$$\mathbf{P}_1^{\perp} = \mathbf{W}_1^{\mathrm{T}} \, \tilde{\mathbf{I}} \, \mathbf{W}_1 , \tag{7-2}$$

where

$$\tilde{\mathbf{I}} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array} \right] \begin{array}{l} \} \, M_1 \\ \} \, N-M_1 \end{array} .$$

The constrained least-squares functional then becomes

$$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| \mathbf{W}_1^{\mathrm{T}} \, \tilde{\mathbf{I}} \, \mathbf{W}_1 \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \right|\right| .$$

If we now note the isometric properties of $\mathbf{W}_1$, this becomes

$$\chi_2(\underline{\theta},\underline{a}_2) = \left|\left| \tilde{\mathbf{I}} \, \mathbf{W}_1 \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \right|\right|^2 .$$

And, finally, if we partition $\mathbf{W}_1$ as

$$\mathbf{W}_1 = \left[\begin{array}{c} \mathbf{W}_{11} \\ \hline \mathbf{W}_{12} \end{array} \right] \begin{array}{l} \} \, M_1 \\ \} \, N\text{-}M_1 \end{array},$$

then our least-squares functional can be written as

$$\chi_2(\underline{\theta}, \underline{a}_2) = \left|\left| \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & \mathbf{I} \end{array} \right] \left[\begin{array}{c} \mathbf{W}_{11} \\ \hline \mathbf{W}_{12} \end{array} \right] \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \right|\right|^2$$

$$= \left|\left| \left[\begin{array}{c} 0 \\ \hline \mathbf{W}_{12} \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \end{array} \right] \right|\right|^2$$

$$= \left|\left| \mathbf{W}_{12} \left(\underline{y} - \mathbf{F}_2(\underline{\theta}) \, \underline{a}_2 \right) \right|\right|^2$$

$$= \left|\left| \underline{v} - \mathbf{G}(\underline{\theta}) \, \underline{a}_2 \right|\right|^2 .$$

Substituting the linear least-squares estimator for $\underline{a}_2$, we get

$$\phi_2(\underline{\theta}) = \left|\left| \mathbf{P}_G^{\perp} \, \underline{v} \right|\right|^2 ,$$

where

$$\mathbf{G} = \mathbf{W}_{12} \, \mathbf{F}_2 \tag{7-3}$$

and

$$\underline{v} = \mathbf{W}_{12} \, \underline{y} \, . \tag{7-4}$$

To effect these, we apply the orthogonal matrix $\mathbf{W}_1$ to the columns of $\mathbf{F}_2$ and to $\underline{y}$, then discard the first M_1 rows of $\mathbf{W}_1\mathbf{F}_2$ and the first M_1 elements of $\mathbf{W}_1\underline{y}$, leaving the $(N\text{-}M_1)\times M_2$ matrix $\mathbf{G}$ and the $(N\text{-}M_1)$-vector $\underline{v}$.

PSEUDOINVERSE AND PROJECTOR DEFINITIONS

Throughout the remainder of this section, we use results from the QR factorization of the constrained basis funtion matrix, given by

$$\mathbf{Q}\,\mathbf{G} = \mathbf{R}\,, \qquad \mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \hline \mathbf{0} \end{bmatrix}. \tag{7-5}$$

In particular, we will use the following definitions for the pseudoinverse and projection operators:

$$\mathbf{G}^{+} = \mathbf{R}^{+}\,\mathbf{Q}\,, \qquad \mathbf{R}^{+} = \left[\, \mathbf{R}_1^{-1} \;\middle|\; \mathbf{0} \,\right], \tag{7-6}$$

$$\mathbf{P}_G = \mathbf{Q}^T\,\mathbf{I}_1\,\mathbf{Q}\,, \qquad \mathbf{I}_1 = \left[\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \begin{matrix} \} M_2 \\ \} N\text{-}M \end{matrix}, \tag{7-7}$$

$$\mathbf{P}_G^{\perp} = \mathbf{Q}^T\,\mathbf{I}_2\,\mathbf{Q}\,, \qquad \mathbf{I}_2 = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right] \begin{matrix} \} M_2 \\ \} N\text{-}M \end{matrix}. \tag{7-8}$$

JACOBIAN MATRIX CALCULATION

The Jacobian matrix is defined as the derivative of the error vector with respect to the parameter vector. If we take the partial derivative with respect to a single parameter, then we get a column of the Jacobian matrix. An expression for this, given in Eq. (5-2), is

$$\underline{J}_i = -\,\mathbf{P}_G^{\perp}\,\mathbf{D}_i(\mathbf{G})\,\mathbf{G}^{+}\,\underline{y} \;-\; (\mathbf{G}^{+})^T\,\mathbf{D}_i(\mathbf{G}^T)\,\mathbf{P}_G^{\perp}\,\underline{y}\,.$$

Substituting the definitions for the pseudoinverse and projector, we get

$$\underline{J}_i = - \mathbf{Q}^T \mathbf{I}_2 \mathbf{Q} \mathbf{D}_i(\mathbf{G}) \mathbf{R}^+ \mathbf{Q} \underline{v} - \mathbf{Q}^T (\mathbf{R}^+)^T \mathbf{D}_i(\mathbf{G}^T) \mathbf{Q}^T \mathbf{I}_2 \mathbf{Q} \underline{v} , \tag{7-9}$$

and/or $$\underline{J}_i = - \mathbf{Q}^T \left[\mathbf{I}_2 \mathbf{Q} \mathbf{D}_i(\mathbf{G}) \mathbf{R}^+ \mathbf{Q} \underline{v} + (\mathbf{R}^+)^T \mathbf{D}_i(\mathbf{G}^T) \mathbf{Q}^T \mathbf{I}_2 \mathbf{Q} \underline{v} \right] . \tag{7-10}$$

If we now define the N-vectors

$$\underline{z}_{i1} = \mathbf{Q} \mathbf{D}_i(\mathbf{G}) \mathbf{R}^+ \mathbf{Q} \underline{v} \tag{7-11}$$

and $$\underline{z}_{i2} = (\mathbf{R}^+)^T \mathbf{D}_i(\mathbf{G}^T) \mathbf{Q}^T \mathbf{I}_2 \mathbf{Q} \underline{v} , \tag{7-12}$$

then the typical column of the Jacobian matrix becomes

$$\underline{J}_i = - \mathbf{Q}^T \left[\mathbf{I}_2 \underline{z}_{i1} + \underline{z}_{i2} \right] . \tag{7-13}$$

At this point, it is convenient to define the $(N-M_1)$-vector

$$\underline{w} = \mathbf{Q} \underline{v} \tag{7-14}$$

so that Eq. (7-4) then becomes

$$\underline{z}_{i1} = \mathbf{Q} \left\{ \mathbf{D}_i(\mathbf{G}) \left\{ \mathbf{R}^+ \underline{w} \right\} \right\} . \tag{7-15}$$

Here we have included braces to denote the order in which we perform the operations. First, we perform back substitution of the $M_2 \times M_2$ matrix $\mathbf{R}_1$ with the first M_2 elements of $\underline{w}$ to yield the M_2-vector $\underline{x} = \mathbf{R}^+ \underline{w}$. Since $\mathbf{D}_i(\mathbf{G})$ is an $(N-M_1) \times M_2$ partial derivative matrix with only one or two nonzero columns, the next step will involve some indexing to selectively multiply the nonzero elements of each row by the coresponding elements of $\underline{x}$ to yield the $(N-M_1)$-vector $\mathbf{D}_i(\mathbf{G}) \mathbf{R}^+ \underline{w}$. Next, we apply the orthogonal matrix $\mathbf{Q}$ to yield $\underline{z}_{i1}$.

In evaluating $\underline{z}_{i2}$, we substitute Eq. (7-15) into Eq. (7-13) to yield

$$\underline{z}_{i2} = (R^{+})^{T}\left\{ D_i(G^T) \left\{ Q^T I_2 \underline{w} \right\} \right\} .$$

Here, we've again used braces to indicate the order of operations. We first replace the first M_2 elements of vector $\underline{w}$ with zeros and apply to $\underline{w}$ the transpose of the orthogonal matrix Q. We then form the inner product between the nonzero columns of $D_i(G)$; i.e., those which correspond to the i'th parameter and the N-vector $Q^T I_2 \underline{w}$. These inner products are the nonzero elements of the M-vector $D_i(G^T)\ Q^T I_2 \underline{w}$. We then perform forward substitution of the lower triangular matrix R_1^T with the first M elements of the vector $D_i(G^T)\ Q^T I_2 \underline{w}$ to yield the M_2 nonzero elements of $\underline{z}_{i2}$.

To complete the i'th column of the Jacobian matrix, we would define an $(N-M_1)$-vector $\underline{z}_i$, whose first M_2 elements are the elements of $\underline{z}_{i2}$ and whose last N-M elements are the last N-M elements of $\underline{z}_{i1}$, and then apply to this vector the transpose of the orthogonal matrix Q. Since, however, we normally use the Jacobian matrix for approximating the Hessian matrix as

$$H \simeq 2\ J^T J\ , \qquad (7\text{-}16)$$

this leading matrix Q^T will cancel itself in the product, so that the final step of applying matrix Q^T can be skipped. Using this approximation to the Hessian, the approximate Newton direction (the Gauss-Newton direction) can be obtained (see Appendix C) by solving the linear least-squares problem $J\ \underline{d} = -\underline{e}$. Here, we note that the leading matrix Q^T appears in the definitions of both the Jacobian matrix and the error vector $\underline{e} = Q^T I_2 Q \underline{v}$, so that it should be neglected when the Gauss-Newton direction is the desired end result.

A much simpler Gauss-Newton algorithm, however, has been devised by Kaufman [8] that allows us to avoid computing $\underline{z}_{i2}$, which is the computationally more costly of the two $\underline{z}_i$ terms. In a manner similar to the introduction of the pole constraints, Kaufman's algorithm takes advantage of the structure of the projection operator when constructed from the QR factorization of the basis function matrix. This is the topic of the next subsection.

MODIFIED GAUSS-NEWTON ALGORITHM

Noting that the matrix $\mathbf{Q}$ is isometric (i.e., length-preserving), we know that

$$\left|\left| P_G^{\perp} \underline{y} \right|\right|^2 = \left|\left| \mathbf{Q} P_G^{\perp} \underline{y} \right|\right|^2$$

$$= \left|\left| \mathbf{Q} \mathbf{Q}^T I_2 \mathbf{Q} \underline{y} \right|\right|^2$$

$$= \left|\left| I_2 \mathbf{Q} \underline{y} \right|\right|^2. \tag{7-17}$$

We can therefore define the modified variable projection functional

$$\psi_3(\underline{\theta}) = \left|\left| \mathbf{Q}_2 \underline{y} \right|\right|^2, \tag{7-18}$$

where we have partitioned $\mathbf{Q}$ as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 \\ \hline \mathbf{Q}_2 \end{bmatrix} \begin{matrix} \} M_2 \\ \} N-M_2 \end{matrix}. \tag{7-19}$$

While the partial derivative of $\mathbf{Q}_2$ is dependent upon the orthogonalization process in which the matrix $\mathbf{Q}$ is determined (and therefore is not unique), Kaufman [8] derives the following general formula whose results (though nonunique) are similar *within an orthogonal transformation*:

$$D_i(\mathbf{Q}_2) = - \mathbf{Q}_2 D_i (G) R^+ \mathbf{Q} + \Gamma \mathbf{Q}, \tag{7-20}$$

where

$$\Gamma^T + \Gamma = 0. \tag{7-21}$$

Since the matrix Γ is not unique, neither is $D_i(\mathbf{Q}_2)$. We can, however, choose $\Gamma = 0$, which certainly satisfies Eq. (7-21), leaving

$$\tilde{\underline{J}}_i = D_i(Q_2)\ \underline{v} = -\ Q_2\ D_i(G)\ R^+\ Q\ \underline{v}\ , \tag{7-22}$$

which is just the last N-M elements of the N-vector $\underline{z}_{i1}$ from the previous subsection.

For this modified signal basis VPF, the error vector is

$$\underline{e} = Q_2\ \underline{v}\ .$$

Defining the matrix $\tilde{J}$, whose columns are the $\tilde{\underline{J}}_i$, the equation for the modified Gauss-Newton direction becomes

$$\underline{d} = \tilde{J}^+\ Q_2\ \underline{v}\ .$$

GRADIENT VECTOR CALCULATION

The typical element of the gradient vector is given in Eq. (5-4) as

$$g_i = -\ 2\ \underline{v}^T\ P_G^{\perp}\ D_i(G)\ G^+\ \underline{v}\ .$$

Substituting for the pseudoinverse and projector, we get

$$g_i = -\ 2\ \underline{v}^T\ Q^T\ I_2\ Q\ D_i(G)\ R^+\ \underline{v}$$

$$= -\ 2\ \underline{w}^T\ I_2\ \underline{z}_{i1}\ , \tag{7-23}$$

where $\underline{w}$ and $\underline{z}_{i1}$ are formed as described in the previous subsection.

HESSIAN MATRIX CALCULATION

We begin by recalling the expression from Chapter 5 for the typical element of

the Hessian matrix, which is given by

$$H_{ij} = 2 \left[P_G^{\perp} \underline{v} \right]^T \left[D_j(G)\, G^+ D_i(G) - D_{ij}^2(G) + D_i(G)\, G^+ D_j(G) \right] \left[G^+ \underline{v} \right]$$

$$- 2 \left[(G^+)^T D_j(G^T)\, P_G^{\perp} \underline{v} \right]^T \left[(G^+)^T D_j(G^T)\, P_G^{\perp} \underline{v} \right]$$

$$+ 2 \left[P_G^{\perp} D_i(G)\, G^+ \underline{v} \right]^T \left[P_G^{\perp} D_i(G)\, G^+ \underline{v} \right] .$$

If we substitute the definitions for the pseudoinverse and projector in terms of the orthogonal factorization, we get

$$H_{ij} = 2 \left[Q^T I_2 Q \underline{v} \right]^T \left[D_j(G)\, R^+ Q\, D_i(G) - D_{ij}^2(G) + D_i(G)\, R^+ Q\, D_j(G) \right] R^+ Q \underline{v}$$

$$- 2 \left\{ \left[R^+ Q \right]^T D_i(G^T)\, Q^T I_2 Q \underline{v} \right\}^T \left[R^+ Q \right]^T D_j(G^T)\, Q^T I_2 Q \underline{v}$$

$$+ 2 \left[Q^T I_2 Q D_j(G)\, R^+ Q \underline{v} \right]^T Q^T I_2 Q D_i(G)\, R^+ Q \underline{v}$$

$$= 2 \underline{w}^T I_2 \left[Q D_j(G)\, R^+ Q D_i(G) - Q D_{ij}^2(G) + Q D_i(G)\, R^+ Q D_j(G) \right] R^+ \underline{w}$$

$$- 2 \left[(R^+)^T D_i(G^T)\, Q^T I_2 \underline{w} \right]^T Q Q^T \left[(R^+)^T D_j(G^T)\, Q^T I_2 \underline{w} \right]$$

$$+ 2 \left[I_2 Q D_j(G)\, R^+ \underline{w} \right]^T Q Q^T \left[I_2 Q D_i(G)\, R^+ \underline{w} \right] , \qquad (7\text{-}24)$$

where $\underline{w}$ is defined the same way as during the Jacobian calculation. Note that the product QQ^T will become identity in the last two terms. Let us also recall our previous definitions of $\underline{x}$, $\underline{z}_{i1}$, and $\underline{z}_{i2}$; i.e.,

$$\underline{x} = \mathbf{R}^{+}\ \underline{w},$$

$$\underline{z}_{i1} = \mathbf{Q}\ D_i(G)\ \mathbf{R}^{+}\ \underline{w}\ ,$$

and

$$\underline{z}_{i2} = (\mathbf{R}^{+})^{T}\ D_i(G^{T})\ \mathbf{Q}^{T}\ I_2\ \underline{w}\ .$$

We now can write the typical element of the Hessian as

$$H_{ij} = 2 \left\{ \underline{z}_{j2}^{T}\ \underline{z}_{i1} + \underline{z}_{i2}^{T}\ \underline{z}_{j1} - \underline{z}_{i2}^{T}\ \underline{z}_{j2} + \underline{z}_{j1}^{T}\ I_2\ \underline{z}_{i1} \right\}$$

$$- 2\ \underline{w}^{T}\ I_2\ \mathbf{Q}\ D_{ij}^{2}(G)\ \underline{x}\ . \qquad (7\text{-}25)$$

Here we will need to perform some special indexing to achieve the multiplication by the second partial derivative matrices. Otherwise, this last term is calculated straightforwardly. Having obtained the Hessian matrix, the Newton direction is the solution to the equation $H\ \underline{d} = -\ g$, which is a square symmetric system that can be solved efficiently using the LU decomposition.

In the neighborhood of the minimum (where we are most likely to use Newton's method), the functional is approximately quadratic so that a line search to determine step size is not necessary; i.e., we set the step size equal to one.

7.2 Optimization of Prediction Filter Coefficients

We will now discuss the algorithms for minimizing the prediction filter VPF. This path yields much more efficient algorithms because the convolution matrix is a linear function of the coefficients. The formation of the convolution matrix and its derivatives is therefore purely a matter of indexing. Furthermore, the second partial derivatives of the convolution matrix vanish, thus simplifying the Hessian calculation. There does not, however, appear to be a Gauss-Newton algorithm which is equivalent to the Kaufman algorithm.

Because forming the Hessian matrix is only slightly more costly than forming the Jacobian matrix for this error norm, and because Newton's method generally does not need a line search to determine the step size, Newton's method is computationally more efficient than the Gauss-Newton method when it can be used; i.e., when the Hessian is positive definite.

PSEUDOINVERSE AND PROJECTOR DEFINITIONS

Throughout the remainder of this section, we will use results from the RV factorization of the convolution matrix, given by

$$\mathbf{B}\,\mathbf{V} = \mathbf{R}, \qquad \mathbf{R} = \left[\begin{array}{c|c} \mathbf{R}_1 & \mathbf{0} \end{array}\right].$$

In particular, we will use the following definitions for the pseudoinverse and projection operators:

$$\mathbf{B}^{+} = \mathbf{V}\,\mathbf{R}^{+}, \qquad \mathbf{R}^{+} = \left[\begin{array}{c} \mathbf{R}_1^{-1} \\ \hline \mathbf{0} \end{array}\right],$$

$${}_{\mathbf{B}}\mathbf{P} = \mathbf{V}\,\mathbf{I}_1\,\mathbf{V}^{T}, \qquad \mathbf{I}_1 = \left[\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \begin{array}{l} \}\ \text{N-M} \\ \}\ \text{M} \end{array},$$

$${}_{\mathbf{B}}\mathbf{P}^{\perp} = \mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^{T}, \qquad \mathbf{I}_2 = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right] \begin{array}{l} \}\ \text{N-M} \\ \}\ \text{M} \end{array}.$$

JACOBIAN MATRIX CALCULATION

We begin by recalling the expression from Chapter 5 for typical column of the Jacobian matrix, given by

$$\underline{J}_i = \mathbf{B}^{+}\,\mathbf{D}_i(\mathbf{B})\;{}_{\mathbf{B}}\mathbf{P}^{\perp}\,\underline{y} + {}_{\mathbf{B}}\mathbf{P}^{\perp}\,\mathbf{D}_i(\mathbf{B}^{T})\;(\mathbf{B}^{+})^{T}\underline{y}\,.$$

If we recall that the partial derivative of the convolution matrix can be factored as $D_i(B) = D_i(U)\ C$, then we get

$$\underline{J}_i = V\ R^+\ D_i(U)\ C\ V\ I_2\ V^T\ \underline{Y} \ +\ V\ I_2\ V^T\ C^T\ D_i(U^T)\ (R^+)^T\ V^T\ \underline{Y}\ . \qquad (7\text{-}26)$$

Let us now define

$$\underline{z}_{i1} = R^+\ D_i(U)\ C\ V\ I_2\ V^T\ \underline{Y}\ , \qquad (7\text{-}27)$$

and

$$\underline{z}_{i2} = V^T\ C^T\ D_i(U^T)\ (R^+)^T\ V^T\ \underline{Y}\ . \qquad (7\text{-}28)$$

The typical column of the Jacobian can then be written as

$$\underline{J}_i = V \left\{ \underline{z}_{i1} + I_2\ \underline{z}_{i2} \right\}\ . \qquad (7\text{-}29)$$

As before, we note that the primary purpose for forming the Jacobian matrix is to calculate the Gauss-Newton direction $\underline{d}$ by solving the linear least-squares problem $J\ \underline{d} \cong -\ \underline{e}$. We also note that the orthogonal matrix V appears as the leading term on both sides of the equation (i.e., in the definition of both J and $\underline{e}$), so it can be ignored.

In discussing the formation of the $\underline{z}_i$, an example case will be helpful to illustrate the manipulations of the convolution matrices. We will consider the case with $N = 12$, $M_1 = 3$, and $M_2 = 2$ (thus $M = 5$). Thus, for our example

$$B = \begin{bmatrix} b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 1 \end{bmatrix},$$

$$U = \begin{bmatrix} u_2 & u_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & u_2 & u_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & u_2 & u_1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_2 & u_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & u_2 & u_1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & u_2 & u_1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u_2 & u_1 & 1 \end{bmatrix},$$

$$C = \begin{bmatrix} c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & 1 \end{bmatrix}.$$

Before evaluating the $\underline{z}_i$, it is convenient to define

$$\underline{w} = \mathbf{V}^T \underline{y} \ , \tag{7-30}$$

so that the expression for $\underline{z}_{i1}$ becomes

$$\underline{z}_{i1} = \mathbf{R}^+ \left\{ D_i(U) \left\{ \mathbf{C} \left\{ \mathbf{V} \, \mathbf{I}_2 \, \underline{w} \right\} \right\} \right\} .$$

As before, the braces indicate the order of operations in forming the vector.

We begin by setting to zero the first N-M elements of the vector $\underline{w}$ and by applying the orthogonal transformation represented by matrix **V**. We then effect the premultiplication by matrix **C** as a vector convolution, without edge effects, between $\mathbf{V} \, \mathbf{I}_2 \, \underline{w}$ and the vector

$$\left[\, 1 \quad c_1 \quad c_2 \, \cdots \, c_{M_1} \, \right] .$$

If we now examine the structure of the partial derivative of matrix **U** with respect to one of the coefficients, we see that these derivative matrices are merely $(N-M)\times(N-M)$ identity matrices with zero columns added to make an $(N-M)\times(N-M_1)$ matrix. Shown below is the partial derivative of matrix **U** with respect to u_2 for our example.

$$D_2(U) = \frac{\partial U}{\partial u_2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} .$$

If we premultiply an arbitrary $(N-M_1)$-vector $\underline{x}$ with this partial derivative, we see that the result is simply to select an (N-M)-element segment of the vector $\underline{x}$. For our above example and the vector

$$\underline{x} = \left[\, x_1 \, x_2 \, \cdots \, x_9 \, \right]^T ,$$

the resulting product is

$$D_2(U)\ \underline{x} = \left[\ x_1\ \ x_2\ \ldots\ x_6\ \ x_7\ \right]^T.$$

The partial derivative of our example matrix U with repect to u_1 would contain zeros in the first and last column, and an identity matrix in columns two through nine. This would appear as

$$D_1(U)\ \underline{x} = \left[\ x_2\ \ x_3\ \ldots\ x_7\ \ x_8\ \right]^T.$$

Extending this to the general case, we would obtain

$$D_{M_2}(U)\ \underline{x} = \left[\ x_1\ \ x_2\ \ldots\ x_{N-M}\ \right]^T,$$

$$D_{M_2-1}(U)\ \underline{x} = \left[\ x_2\ \ x_3\ \ldots\ x_{N-M+1}\ \right]^T,$$

$$\vdots$$

$$D_1(U)\ \underline{x} = \left[\ x_{M_2}\ \ x_{M_2+1}\ \ldots\ x_{N-M_1-1}\ \right]^T.$$

The matrix product can therefore be effected simply by selecting the appropriate elements from the $(N-M_1)$-vector $\mathbf{C}\ \mathbf{V}\ \mathbf{I}_2\ \underline{w}$ to form the $(N-M)$-vector $D_i(U)\ \mathbf{C}\ \mathbf{V}\ \mathbf{I}_2\ \underline{w}$. Having obtained this result, we now effect the premultiplication of the matrix $\mathbf{R}^+$ by performing back substitution of the $(N-M)\times(N-M)$ matrix $\mathbf{R}_1$ with the first $(N-M)$ elements of the product $D_i(U)\ \mathbf{C}\ \mathbf{V}\ \mathbf{I}_2\ \underline{w}$ to yield the nonzero elements of $\underline{z}_{i1}$.

The expression for $\underline{z}_{i2}$ is

$$\underline{z}_{i2} = \mathbf{V}^T \left\{ \mathbf{C}^T \left\{ D_i(U^T) \left\{ (\mathbf{R}^+)^T\ \underline{w} \right\} \right\} \right\}.$$

We begin forming $\underline{z}_{i2}$ by performing forward elimination of the (N-M)x(N-M) lower triangular matrix R_1^T with the first (N-M) elements of $\underline{w}$ to yield $\underline{x} = (R^+)^T \underline{w}$. We then again note the special structure of the partial derivative matrices and recognize that the transpose of each of these matrices is simply an (N-M)x(N-M) identity matrix with zero rows added to make an $(N-M_1)$x(N-M) matrix. Thus when we premultiply the above (N-M)-vector $\underline{x}$ with one of the derivative matrices, the effect is simply to pad the vector with zero elements to make an $(N-M_1)$-vector; i.e., the vector

$$\underline{x} = \left[\begin{array}{cccc} x_1 & x_2 & \cdots & x_7 \end{array} \right]^T,$$

when premultiplied by $D_2(U^T)$ from our example becomes

$$D_2(U^T)\ \underline{x} = \left[\begin{array}{cccccc} x_1 & x_2 & \cdots & x_6 & x_7 & 0 \quad 0 \end{array} \right]^T.$$

For the derivative with respect to u_1, this product becomes

$$D_1(U^T)\ \underline{x} = \left[\begin{array}{cccccc} 0 & x_1 & x_2 & \cdots & x_6 & x_7 \quad 0 \end{array} \right]^T.$$

For the general case, with an (N-M)-vector $\underline{x}$, the resulting products are the following $(N-M_1)$-vectors:

$$D_{M_2}(U^T)\ \underline{x} = \left[\begin{array}{cccccc} x_1 & x_2 & \cdots & x_{N-M} & 0 & \cdots 0 \end{array} \right]^T,$$

$$D_{M_2-1}(U^T)\ \underline{x} = \left[\begin{array}{cccccc} 0 & x_1 & x_2 & \cdots & x_{N-M} & 0 \cdots 0 \end{array} \right]^T,$$

$$\vdots$$

$$D_1(U^T)\ \underline{x} = \left[\begin{array}{cccccc} 0 \cdots 0 & x_1 & x_2 & \cdots & x_{N-M} & 0 \end{array} \right]^T,$$

where the total number of zeros in each vector is equal to M_2.

The next step, after having effected the multiplication of the partial derivatives, is to premultiply by the transpose of the convolution matrix **C**. This is equivalent to convolving, with edge effects, $D_i(U^T)\ \underline{x}$ with the vector

$$\left[\begin{array}{cccc} c_{M_1} & c_{M_1+1} & \cdots & c_1 \quad 1 \end{array} \right]^T .$$

We then apply the transpose of matrix **V** to yield the result $\underline{z}_{i2}$.

GRADIENT VECTOR CALCULATION

Recall from Chapter 5 that the typical element of the gradient vector is

$$g_i = 2\ \underline{Y}^T\ \mathbf{B}\ \ D_i(\mathbf{B})\ {}_{\mathbf{B}}\mathbf{P}^{\perp}\ \underline{Y}\ .$$

Substituting the definitions for the pseudoinverse and projection operator in terms of the **RV** factorization and recalling that $D_i(\mathbf{B}) = D_i(\mathbf{U})\ \mathbf{C}$, this becomes

$$g_i = 2\ \underline{Y}^T\ \mathbf{V}\ \mathbf{R}^+\ D_i(\mathbf{U})\ \mathbf{C}\ \mathbf{V}\ \mathbf{I}_2\ \mathbf{V}^T\ \underline{Y}\ . \tag{7-31}$$

If we now substitute $\underline{w}$ for $\mathbf{V}^T\underline{Y}$, we get

$$g_i = 2\ \underline{w}^T\ \mathbf{R}^+\ D_i(\mathbf{U})\ \mathbf{C}\ \mathbf{V}\ \underline{w}_2 . \tag{7-32}$$

But we can also recognize that this is the same as

$$g_i = 2\ \underline{w}^T\ \underline{z}_{i1}, \tag{7-33}$$

where $\underline{z}_{i1}$ is formed as described in the previous subsection.

HESSIAN MATRIX CALCULATION

Recall from Chapter 5 that the typical element of the Hessian matrix for

the prediction filter VPF is given by

$$H_{ij} = 2\,\underline{Y}^T \Big\{ \Big[{}_B\mathbf{P}^{\perp} D_j(\mathbf{B}^T)\,(\mathbf{B}^+)^T + \mathbf{B}^+ D_j(\mathbf{B})\,\mathbf{B}^+ \Big] D_i(\mathbf{B})\,{}_B\mathbf{P}^{\perp} - \mathbf{B}^+ D_i(\mathbf{B}) \Big[\mathbf{B}^+ D_j(\mathbf{B})\,{}_B\mathbf{P}^{\perp} + {}_B\mathbf{P}^{\perp} D_j(\mathbf{B}^T)\,(\mathbf{B}^+)^T \Big] \Big\} \underline{Y}$$

$$= 2\,\underline{Y}^T {}_B\mathbf{P}^{\perp} D_j(\mathbf{B}^T)\,(\mathbf{B}^+)^T \mathbf{B}^+ D_i(\mathbf{B})\,{}_B\mathbf{P}^{\perp}\underline{Y} - 2\,\underline{Y}^T \mathbf{B}^+ D_j(\mathbf{B})\,\mathbf{B}^+ D_i(\mathbf{B})\,{}_B\mathbf{P}^{\perp}\underline{Y}$$

$$- 2\,\underline{Y}^T \mathbf{B}^+ D_i(\mathbf{B})\,\mathbf{B}^+ D_j(\mathbf{B})\,{}_B\mathbf{P}^{\perp}\underline{Y} - 2\,\underline{Y}^T \mathbf{B}^+ D_i(\mathbf{B})\,{}_B\mathbf{P}^{\perp} D_j(\mathbf{B}^T)\,(\mathbf{B}^+)^T \underline{Y}\,.$$

Substituting the definitions for the pseudoinverse and projection operator in terms of the RV factorization, and recalling that $D_i(\mathbf{B}) = D_i(\mathbf{U})\,\mathbf{C}$, this becomes

$$H_{ij} = 2\,\underline{Y}^T \mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \mathbf{C}^T D_j(\mathbf{U}^T)\,(\mathbf{R}^+)^T \mathbf{V}^T \mathbf{V}\,\mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \underline{Y}$$

$$- 2\,\underline{Y}^T \mathbf{V}\,\mathbf{R}^+ D_j(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \underline{Y}$$

$$- 2\,\underline{Y}^T \mathbf{V}\,\mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{R}^+ D_j(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \underline{Y}$$

$$- 2\,\underline{Y}^T \mathbf{V}\,\mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \mathbf{C}^T D_j(\mathbf{U}^T)\,(\mathbf{R}^+)^T \mathbf{V}^T \underline{Y}$$

$$H_{ij} = 2\,\underline{w}^T \mathbf{I}_2\,\mathbf{V}^T \mathbf{C}^T D_j(\mathbf{U}^T)\,(\mathbf{R}^+)^T \mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\underline{w}$$

$$- 2\,\underline{w}^T \mathbf{R}^+ D_j(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\underline{w}$$

$$- 2\,\underline{w}^T \mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{R}^+ D_j(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\underline{w}$$

$$- 2\,\underline{w}^T \mathbf{R}^+ D_i(\mathbf{U})\,\mathbf{C}\,\mathbf{V}\,\mathbf{I}_2\,\mathbf{V}^T \mathbf{C}^T D_j(\mathbf{U}^T)\,(\mathbf{R}^+)^T \underline{w}\,. \qquad (7\text{--}34)$$

We can further simplify this as

$$H_{ij} = 2 \left[R^{+} D_j(U)\ C\ V\ I_2\ \underline{w} \right]^T \left[R^{+} D_i(U)\ C\ V\ I_2\ \underline{w} \right]$$

$$- 2 \left[V^T C^T D_j(U^T)\ (R^{+})^T \underline{w} \right]^T \left[R^{+} D_i(U)\ C\ V\ I_2\ \underline{w} \right]$$

$$- 2 \left[V^T C^T D_i(U^T)\ (R^{+})^T \underline{w} \right]^T \left[R^{+} D_j(U)\ C\ V\ I_2\ \underline{w} \right]$$

$$- 2 \left[I_2\ V^T C^T D_i(U^T)\ (R^{+})^T \underline{w} \right]^T \left[I_2\ V^T C^T D_j(U^T)\ (R^{+})^T \underline{w} \right], \qquad (7\text{-}35)$$

and $$H_{ij} = 2 \left\{ - \underline{z}_{j2}^T \underline{z}_{i1} - \underline{z}_{i2}^T \underline{z}_{j1} + \underline{z}_{j1}^T \underline{z}_{i1} - \underline{z}_{i2}^T I_2\ \underline{z}_{j2} \right\}, \qquad (7\text{-}36)$$

where the $\underline{z}_i$, $\underline{z}_j$ were calculated for the Jacobian matrix. As before, the Newton direction is calculated from the equation $H\ \underline{d} = -\ g$, and the step is taken with step size equal to one.

(Blank Page)

REFERENCES

[1] G. H. Golub and V. Pereyra, "The Differentiation of Pseudoinverses and Nonlinear Least Squares Problems whose Variables Separate", *SIAM Journal Numerical Analysis*, **10**, (2) (April 1973).

[2] T. A. Parks, "Reducible Nonlinear Programming Problems," Ph.D. thesis, Rice University, Houston, TX, May 1985.

[3] M. R. Osborne, "Some Special Nonlinear Least Squares Problems", *SIAM Journal Numerical Analysis*, **12**, (4) (September 1975).

[4] J. D. George, P. L. Ainsleigh, and R. R. Muise, "Stepped Sine Response Modeling Using Noisy Sampled Data", NRL Memorandum Report No.6644 (still being written).

[5] G. H. Golub and V. Pereyra, "The Differentiation of Pseudoinverses, Separable Nonlinear Least Squares Problems, and Other Tales", *Generalized Inverses and Applicatications*, M.Z. Nashed, Ed. (Academic Press, New York, 1976).

[6] R. Kumaresan, L. L. Scharf, and A. K. Shaw, "An Algorithm for Pole-Zero Modeling and Spectral Analysis", *IEEE Transactions on ASSP*, **ASSP-34**, (5) (October 1986).

[7] Y. Bresler and A. Macovski, "Exact Maximum Likelihood Parameter Estimation of Superimposed Exponential Signals in Noise," *IEEE Transactions on ASSP* **ASSP-34** (5) (June 1986).

[8] L. Kaufman, "A Variable Projection Method for Solving Separable Nonlinear Least Squares Problems", *BIT 15* (1975).

[9] P. R. Halmos, *Introduction to Hilbert Space* (Chelsea Publishing Co., New York, 1957).

[10] J. H. Wilkinson, *The Algebraic Eigenvalue Problem* (Oxford University Press, London, 1965).

[11] C. L. Hanson and R. J. Lawson, *Solving Least Squares Problems* (Prentice Hall, Englewood Cliffs, NJ, 1974).

[12] G. H. Golub and C. F. Van Loan, *Matrix Computations* (Johns Hopkins University Press, Baltimore, MD, 1983).

[13] R. C. Rao and S. K. Mitra, *The Generalized Inverse of a Matrix and its Applications* (John Wiley, New York, 1971).

[14] G. H. Golub, V. Klema, and G. W. Stewart, Stanford University Report Stan-CS-76-559 (AD-A032 348), "Rank Degeneracy and Least Squares Problems," Stanford U. Computer Science Department, Palo Alto CA (August 1976).

[15] G. W. Stewart, *Introduction to Matrix Computations* (Academic Press, New York, 1973).

REFERENCES (CONT'D)

[16] Y. Bard, *Nonlinear Parameter Estimation* (Academic Press, New York, 1974).

[17] D. G. Luenberger, *Linear and Nonlinear Programming* (Addison Wesley, Reading, MA, 1984).

APPENDIX A - REVIEW OF LINEAR LEAST-SQUARES THEORY

In this appendix, we review some of the elements of linear least-squares theory essential to nonlinear least-squares optimization. The purpose here is two-fold: to elucidate in some detail the nature of the projection operator, the generalized inverse, and the pseudoinverse; and to describe the application of the **QR** factorization (and its relatives) to solving linear least-squares problems.

We will begin by examining the orthogonal projection operator. Here, following the the work of Halmos [9] and Wilkinson [10], the properties of the projection operator will be discussed. The eigenstructur of this operator is also examined. We then examine the properties of the generalized inverse for finding general solutions and minimum norm solutions to consistent equations and linear least-squares problems. This will culminate in a discussion of the Moore-Penrose generalized inverse, or pseudoinverse, which leads to the minimum norm least-squares solution.

Following this, we examine several scenarios in which **QR** factorization techniques are applied to the linear least-squares problem. While only the full rank case is of interest in the nonlinear algorithms presented in this paper, we examine the rank deficient and near rank deficient cases for application to other aspects of the signal modeling problem, namely the use of the complete orthogonal factorization for effecting rank reduction of the data matrix in the algorithm for obtaining the initial estimates (see Ref. [4] for details).

In the full rank case, the **QR** factorization is seen to lead directly to the Moore-Penrose generalized inverse. In the rank deficient case, we examine the truncated **QR** factorization and the complete orthogonal factorization as outlined by Hanson and Lawson [11], Golub and Pereyra [5], and Golub and Van Loan [12]. It is seen that while the g-inverse formed using the truncated **QR** factorization does not lead to the pseudoinverse, the complete orthogonal factorization does achieve the minimum norm solution and thus provides an alternative to the singular value decomposition for forming reduced rank

approximations. Finally, the application of the complete orthogonal factorization to nearly rank deficient matrices is discussed.

A.1 The Orthogonal Projection Operator

Every projection operator is defined on a particular linear manifold (subspace) in a finite dimensional vector space. One of the key relationships between the projector and the corresponding subspace is that the subspace is invariant under the transformation represented by the projection operator; i.e., the transformation operating on any vector from the subspace will return another vector in the same subspace. Also, the projector represents the identity transformation over this invariant subspace. One special class of projectors is called the orthogonal projector, which must satisfy somewhat stricter requirements regarding the associated subspace. These properties we will now develop.

Consider the N-dimension vector space R^N and a subspace S in R^N. There exists a subspace $\bar{S}$, called the orthogonal complement of S, such that R^N is the direct sum of S and $\bar{S}$. Drawing from the work of Halmos [9], the following definitions and Eqs. (A-1) through (A-7) characterize the orthogonal projection operators associated with S.

Definition 1 There exists an operator P that maps every vector in R^N onto the subspace S by projecting along the orthogonal complement $\bar{S}$. This operator is called the orthogonal projection operator onto the subspace S in R^N.

Definition 2 There exists an operator, $P^{\perp}$, which maps every vector in R^N onto the complement subspace $\bar{S}$ by projecting along the subspace S. This operator is called the orthogonal projection operator onto the orthogonal complement of S in R^N.

Furthermore, these projectors satisfy the relationship

$$P^{\perp} = I - P \ . \tag{A-1}$$

For convenience, we will subsequently use the term *projector* to mean *orthogonal projection operator*; i.e., we will consider only those projectors for which a projector onto the orthogonal complement subspace exists and is defined by Eq. (A-1).

Now consider an arbitrary vector $\underline{z} \in R^N$. We can decompose $\underline{z}$ as $\underline{z} = \underline{z}_1 + \underline{z}_2$, where $\underline{z}_1 \in S$ and $\underline{z}_2 \in \bar{S}$. The projectors defined above satisfy the following six relationships:

$$P \underline{z}_1 = \underline{z}_1 \tag{A-2}$$

$$P^{\perp} \underline{z}_1 = 0 \tag{A-3}$$

$$P \underline{z}_2 = 0 \tag{A-4}$$

$$P^{\perp} \underline{z}_2 = \underline{z}_2 \tag{A-5}$$

$$P \underline{z} = \underline{z}_1 \tag{A-6}$$

$$P^{\perp} \underline{z} = \underline{z}_2 . \tag{A-7}$$

From Eqs. (A-6) and (A-2), we see that

$$P^2 \underline{z} = P (P \underline{z}) = P \underline{z}_1 = \underline{z}_1 = P \underline{z},$$

so that

$$P^2 = P; \tag{A-8}$$

i.e., the projector is idempotent. We see from Eqs. (A-4) and (A-7) that

$$P (P^{\perp} \underline{z}) = P \underline{z}_2 = 0$$

so that

$$P P^{\perp} = 0 \ . \tag{A-9}$$

Similarly, from Eqs. (A-3) and (A-6), we see that

$$P^{\perp} (P \underline{z}) = P^{\perp} \underline{z}_1 = 0$$

so that

$$P^{\perp} P = 0 . \tag{A-10}$$

Finally, from Eqs. (A-6) and (A-7) and from the definition of orthogonality

$$(P \underline{z})^T (P^{\perp} \underline{z}) = \underline{z}_1^T \underline{z}_2 = 0 .$$

Substituting Eq. (A-1) and transposing within the parentheses yields

$$\underline{z}^T P^T (I - P) \underline{z} = 0.$$

Since this is true for all $\underline{z}$ in R^N, we must have

$$P^T (I - P) = 0$$

so that

$$P^T = P^T P.$$

Since the right-hand side is symmetric, we must have

$$P = P^T. \tag{A-11}$$

Note that it is this symmetry separating the orthogonal projection operator from the general projector which need only be idempotent. In summary, the projectors P and $P^{\perp}$ are idempotent, symmetric, and mutually annihilating.

Further insight can be gained by examining the eigenstructure of the projector. Since the subspace S is invariant under the transformation P, and since P represents the identity transformation over S, S is spanned by a set of the eigenvectors of P, each of which corresponds to a unity eigenvalue [9]. The remaining eigenvectors span the complement of S in R^N and correspond to eigenvalues of zero. The set of all eigenvectors of P forms a basis for R^N. To show that the eigenvalues of P are constrained to the values zero and

one, we note that any symmetric matrix **A** can be transformed [10] such that

$$\mathbf{U}^T \mathbf{A} \mathbf{U} = \mathbf{D}, \tag{A-12}$$

where **U** is an orthogonal matrix and $\mathbf{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ is a diagonal matrix containing the eigenvalues of **A**. The i'th column of **U** is the eigenvector corresponding to the i'th diagonal element of **D**, the i'th eigenvalue of **A**. This is shown clearly by writing Eq. (A-12) as $\mathbf{A U} = \mathbf{U D}$. With this similarity transformation, we may rewrite **A** as

$$\mathbf{A} = \mathbf{U D U}^T. \tag{A-13}$$

Now let **A** be idempotent as well as symmetric, then $\mathbf{A A} = \mathbf{U D U}^T \mathbf{U D U}^T = \mathbf{U D}^2 \mathbf{U}^T$. But since this must also equal **A** as expressed in Eq. (A-13), we must have

$$\mathbf{D}^2 = \mathbf{D}. \tag{A-14}$$

Since the eigenvalues of a symmetric matrix must be real, the diagonal matrix **D** must contain only zeros and ones for Eq. (A-14) to hold.

To see that this eigenstructure exemplifies the operation of the projector, consider the arbitrary N-vector $\underline{z}$ in the space $\mathbf{R}^N$, which has a set of basis vectors $\underline{u}_i$, i=1,...,N. We may choose this set of basis vectors to be the eigenvectors of the NxN matrix **P**, which is the projector onto the (say, M-dimensional) subspace **S**. We may now write

$$\mathbf{P}\, \underline{u}_i = \lambda_i\, \underline{u}_i. \tag{A-15}$$

and

$$\underline{z} = a_1\, \underline{u}_1 + a_2\, \underline{u}_2 + \bullet\bullet\bullet + a_N\, \underline{u}_N. \tag{A-16}$$

Thus

$$\mathbf{P}\, \underline{z} = \lambda_1\, a_1\, \underline{u}_1 + \lambda_2\, a_2\, \underline{u}_2 + \bullet\bullet\bullet + \lambda_N\, a_N\, \underline{u}_N. \tag{A-17}$$

M of the N eigenvalues, those corresponding to the eigenvectors that span the subspace S, will have value unity while the remaining eigenvalues will have a value of zero. We therefore have the result

$$P\,\underline{z} = \sum_{i=1}^{M} a_{j_i}\,\underline{u}_{j_i}\,, \tag{A-18}$$

where the j_i denote those eigenvectors which span S. Thus, components of $\underline{z}$ in S are unchanged while components in $\bar{S}$ are annihilated.

A.2 The Generalized Inverse

In general, an NXM matrix A is a linear transformation that maps an arbitrary vector $\underline{x}$ from an M-dimensional space to an N-dimensional space, which is the range (column space) of the mapping (matrix). We desire an inverse transformation that will map an N-vector $\underline{y}$ lying in the range of A back into an M-dimensional space. If the vector $\underline{y}$ does not lie in the range of A, then the inverse mapping must first approximate $\underline{y}$ with a suitable vector in the range of the mapping. Let us first consider the case where $\underline{y}$ lies in the column space of A (consistent equations).

For the NXM matrix A, the MXN matrix A^+ is a generalized inverse (g-inverse) of A if $\underline{x} = A^+\,\underline{y}$ is a solution to the equation $A\,\underline{x} = \underline{y}$, for any $\underline{y}$ that makes the system consistent [13]. Substituting the first equation into the second, we get $A\,A^+\,\underline{y} = \underline{y}$. Now suppose that given an arbitrary M-vector $\underline{w}$, we let $\underline{y} = A\,\underline{w}$. Since this $\underline{y}$ is generated from the columns of A, it clearly lies in the column space of A. Substituting this into the previous equation yields $A\,A^+\,A\,\underline{w} = A\,\underline{w}$. In general, this requires that the generalized inverse satisfy

$$A\,A^+\,A = A. \tag{A-19}$$

In the most unrestricted sense, this is all that is required of a g-inverse. If we wish, however, to consider the case of inconsistent

equations, then we must impose further restrictions that determine how we wish to approximate $\underline{y}$ before transforming.

A.3 g-inverse for Linear Least-Squares Solution

Geometrically, the best approximant to $\underline{y}$ that makes the system consistent is the projection of $\underline{y}$ onto the column space of **A**, resulting in the least-squares solution to the system of equations. Thus, for the arbitrary N-vector $\underline{z}$, the inconsistent equation $\mathbf{A}\,\underline{x} \simeq \underline{z}$ can be made consistent by premultiplying both sides by the projection operator for the column space of **A**, yielding $\mathbf{P}_A\,\mathbf{A}\,\underline{x} = \mathbf{P}_A\,\underline{z}$. But the projection of the columns of **A** onto themselves leaves them unaffected, so this reduces to

$$\mathbf{A}\,\underline{x} = \mathbf{P}_A\,\underline{z} \ . \tag{A-20}$$

The g-inverse solution for $\underline{x}$ is then $\underline{x} = \mathbf{A}^+\,\mathbf{P}_A\,\underline{z}$. Substituting this for $\underline{x}$ and noting that the projection operator is idempotent, Eq. (A-20) becomes $\mathbf{A}\,(\mathbf{A}^+\,\mathbf{P}_A\,\underline{z}) = (\mathbf{P}_A\,\mathbf{P}_A)\,\underline{z}$, $\mathbf{A}\,\mathbf{A}^+\,\underline{z}_1 = \mathbf{P}_A\,\underline{z}_1$. From this, we see that the generalized inverse for solving linear least-squares problems must be such that $\mathbf{A}\,\mathbf{A}^+ = \mathbf{P}_A$ is the projector onto the column space of **A**. This then requires that the product $\mathbf{A}\,\mathbf{A}^+$ be idempotent and symmetric. That this product is idempotent is equivalent to the requirement of Eq. (A-19). We do have the further restriction, though, that $\mathbf{A}^+$ must satisfy

$$\left(\mathbf{A}\,\mathbf{A}^+\right)^T = \mathbf{A}\,\mathbf{A}^+ \ . \tag{A-21}$$

A.4 g-inverse for Minimum Norm Solution of Consistent Equations

We know from Section A.2 that the g-inverse that provides the solution for the consistent system of equations

$$\mathbf{A}\,\underline{x} = \underline{y} \tag{A-22}$$

must satisfy $\mathbf{A}\,\mathbf{A}^+\,\mathbf{A} = \mathbf{A}$. From this, it follows that $\mathbf{A} - \mathbf{A}\,\mathbf{A}^+\,\mathbf{A} = 0$. Thus

$$A \left(I - A^+ A \right) = 0 \ . \qquad \text{(A-23)}$$

We can therefore state that for any $\underline{z}$,

$$\left(I - A^+ A \right) \underline{z}$$

is a solution to the homogeneous system of equations $A \underline{x} = 0$. As is the case for a linear differential equation, the general solution for the set of simultaneous linear equations in Eq. (A-22) is the sum of the homogeneous solution and a particular solution, and can thus be written $\underline{x} = \underline{x}_p + \underline{x}_h = A^+ \underline{y} + \left(I - A^+ A \right) \underline{z}$.

If we denote the g-inverse leading to the minimum norm solution as A_m^+, then we desire to have

$$\left|\left| A_m^+ \underline{y} \right|\right|^2 \leq \left|\left| A^+ \underline{y} + \left(I - A^+ A \right) \underline{z} \right|\right|^2 \qquad \text{(A-24)}$$

for all $\underline{y}$ and $\underline{z}$. The right-hand side of Eq. (A-24) can be written

$$\left|\left| A^+ \underline{y} + \left(I - A^+ A \right) \underline{z} \right|\right|^2$$
$$= \left\{ A^+ \underline{y} + \left(I - A^+ A \right) \underline{z} \right\}^T \left\{ A^+ \underline{y} + \left(I - A^+ A \right) \underline{z} \right\} ,$$

so that, since all terms are real, Eq. (A-24) becomes

$$\left|\left| A_m^+ \underline{y} \right|\right|^2 \leq \left|\left| A^+ \underline{y} \right|\right|^2 + 2 \underline{y}^T (A^+)^T \left(I - A^+ A \right) \underline{z} + \left|\left| \left(I - A^+ A \right) \underline{z} \right|\right|^2 .$$

This is a minimum when the middle term is zero, which occurs when the particular solution is orthogonal to the homogeneous solution. In general, this requires that

$$(A^+)^T \left[I - A^+ A \right] = 0 ,$$

so that $(A^+)^T = (A^+)^T A^+ A$. For this to be true, it is necessary and sufficient [13] that

$$A^+ A A^+ = A^+ \tag{A-25}$$

and

$$\left[A^+ A \right]^T = A^+ A . \tag{A-26}$$

We will now show that the product $A^+ A$ is, in fact, the projector onto the row space of A.

$$\begin{aligned} A^+ A &= \left[A^+ A \right]^T \\ &= A^T (A^+)^T \\ &= A^T (A^T)^+ , \end{aligned}$$

which is the projection operator onto the columns of A^T, or the rows of A.

In summary, the g-inverse for obtaining a minimum norm solution to $A \underline{x} = \underline{y}$ must be such that

$$_A P = P_{A^T} = A^+ A . \tag{A-27}$$

Let us re-examine the general solution, now written

$$\underline{x} = A^+ \underline{y} + \left[I - {}_A P \right] \underline{z} ,$$

where we have substituted Eq. (A-27) into the homogeneous solution. Recall from Section A.1 that (I - P) is the projector onto the orthogonal complement of the subspace for which P is the projector. Thus we see that the homogeneous solution is confined to the null space; i.e., the orthogonal

complement of the row space of A. Since the minimum norm solution must be orthogonal to this homogeneous solution, what we really are striving for in the minimum norm solution is that solution lying in the row space of the matrix A.

A.5 g-inverse for Minimum Norm Least-Squares Solution

Combining the results of the last two sections, we see that the g-inverse for obtaining the minimum norm least-squares solution [i.e., the Moore-Penrose generalized inverse (pseudoinverse)] must be such that

$$P_A = A\,A^+$$

and

$${}_AP = A^+\,A$$

are, respectively, the orthogonal projectors onto the column space and the row space of A. This is equivalent to the following requirements:

$$A\,A^+\,A = A \qquad\qquad A^+\,A\,A^+ = A^+$$

$$\left(A\,A^+\right)^T = A\,A^+ \qquad\qquad \left(A^+\,A\right)^T = A^+\,A.$$

It is interesting to note that in forming the minimum norm linear least-squares solution, we are actually performing a three-stage process. Starting with the least-squares problem

$$A\,\underline{x} \simeq \underline{y} \tag{A-28}$$

(where A is not necessarily full rank), we obtain the minimum norm solution

$$\underline{x} = A^+\,\underline{y}. \tag{A-29}$$

Stage I: Projection of $\underline{y}$ onto the column space of A to obtain a consistent set of equations. We form $\hat{\underline{y}} = A\,A^+\,\underline{y} = P_A\,\underline{y}$.

Stage II: Solution to the consistent set of equations $\mathbf{A}\,\hat{\underline{x}} = \hat{\underline{y}}$ to yield the general solution $\hat{\underline{x}} = \mathbf{A}^{+}\,\hat{\underline{y}} + [\mathbf{I} - \mathbf{A}^{+}\,\mathbf{A}]\,\underline{z}$, where $\underline{z}$ is an arbitrary vector in R^{N}.

Stage III: Projection of $\hat{\underline{x}}$ onto the row space of $\mathbf{A}$ to obtain the minimum norm solution (eliminate the homogeneous part of the solution). We form $\underline{x} = \mathbf{A}^{+}\,\mathbf{A}\,\hat{\underline{x}} = {}_{A}\mathbf{P}\,\hat{\underline{x}} = {}_{A}\mathbf{P}\,\mathbf{A}^{+}\,\mathbf{P}_{A}\,\underline{y}$.

We now conclude this portion of our discussion of linear least-squares theory. The remainder of the chapter will examine the QR factorization family as it is used for forming projection operators and solving linear least-squares problems.

A.6 QR Factorization of Full Rank Matrices

Consider the NXM matrix A of rank r = M≤N. There exists an NXN orthogonal matrix Q, such that

$$\mathbf{Q}\,\mathbf{A} = \mathbf{R} \equiv \left[\begin{array}{c} \mathbf{R}_1 \\ \hline \mathbf{0} \end{array}\right], \tag{A-30}$$

where $\mathbf{R}_1 = \left[\begin{array}{c} \diagdown \\ \mathbf{0} \quad \diagdown \end{array}\right]_{M\times M}$

is square, upper triangular, and nonsingular. With this, we may write $\mathbf{A} = \mathbf{Q}^{T}\,\mathbf{R}$, and define a g-inverse of A as

$$\mathbf{A}^{+} = \left[\begin{array}{c|c} \mathbf{R}_1^{-1} & \mathbf{0} \end{array}\right] \mathbf{Q}\,. \tag{A-31}$$

Recalling that $P_A = A\,A^+$, the projection operator becomes

$$P_A = Q^T \left[\begin{array}{c} R_1 \\ \hline 0 \end{array}\right] \left[\begin{array}{c|c} R_1^{-1} & 0 \end{array}\right] Q$$

$$= Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array}\right] Q \,, \qquad \text{(A-32)}$$

where I_M is the MXM identity matrix. We see by inspection that this is symmetric, and by squaring we see that it is idempotent

$$P_A^2 = Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array}\right] Q\, Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array}\right] Q$$

$$= Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array}\right] Q = P_A,$$

where we have noted the orthogonality of matrix Q. Thus we see that the g-inverse defined in Eq. (A-31) is adequate for forming the projection operator onto the column space of A.

Now recall the projection operator onto the row space of A,

$$_AP = A^+ A \,.$$

Substituting Eq. (A-31) into this equation yields

$$_AP = \left[\begin{array}{c|c} R_1^{-1} & 0 \end{array}\right] Q\, Q^T \left[\begin{array}{c} R_1 \\ \hline 0 \end{array}\right]$$

$$= I_M \,.$$

Thus this factorization satisfies the requirements for the Moore-Penrose g-inverse. The least-squares functional for the linear least-squares problem $A\,\underline{x} \simeq \underline{y}$ then becomes

$$\phi(\underline{x}) = \left|\left| \mathbf{A}\,\underline{x} - \underline{y} \right|\right|^2$$

$$= \left|\left| \mathbf{Q}\,\mathbf{A}\,\underline{x} - \mathbf{Q}\,\underline{y} \right|\right|^2$$

$$= \left|\left| \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} \underline{x} - \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \underline{y} \right|\right|^2 , \qquad \text{(A-33)}$$

where we have partitioned $\mathbf{Q}$ as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \begin{matrix} \} \, \mathbf{M} \\ \} \, \mathbf{N-M} \end{matrix} .$$

The solution for $\underline{x}$ in this case is determined uniquely as

$$\underline{x}_{LS} = \mathbf{R}_1^{-1}\,\mathbf{Q}_1\,\underline{y} , \qquad \text{(A-34)}$$

leaving a residual sum of squared error

$$\phi(\underline{x}_{LS}) = \left|\left| \mathbf{Q}_2\,\underline{y} \right|\right|^2 . \qquad \text{(A-35)}$$

A.7 QR Factorization of Rank Deficient Matrices

In the case of rank deficient matrices, the QR factorization does not lead to a g-inverse that satisfies the Moore-Penrose conditions. In this section, it is shown that a truncated version of the QR factorization with column pivoting can, however, be used to construct a g-inverse suitable for forming the projector onto the column space of A. In the next section, the complete orthogonal factorization will be presented, which solves the problem of rank degeneracy and leads to the Moore-Penrose pseudoinverse.

Consider the N×M matrix A of rank r<M≤N. There exists an N×N orthogonal matrix Q and an M×M permutation matrix S, such that

$$Q\,A\,S = R \equiv \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right], \tag{A-36}$$

where $R_{11} = \left[\begin{array}{c} \diagdown \\ 0 \end{array}\right]_{r\times r}$.

By truncating R (i.e., replacing R_{12} by a zero matrix), a g-inverse of A is

$$A^{+} = S \left[\begin{array}{c|c} R_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q. \tag{A-37}$$

For this factorization, the projector onto the range of A becomes

$$P_A = Q^T \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right] S^T S \left[\begin{array}{c|c} R_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q$$

$$= Q^T \left[\begin{array}{c|c} I_M & 0 \\ \hline 0 & 0 \end{array}\right] Q \quad , \tag{A-38}$$

which (as in the full rank case) conforms to the requirements of a projection operator. The g-inverse formed from the truncated QR factorization is therefore suitable for forming the projector onto the column space of the basis function matrix.

Let us now form the product

$$A^{+} A = S \left[\begin{array}{c|c} R_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q\, Q^T \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right] S^T$$

$$= S \left[\begin{array}{c|c} I_r & R_{11}^{-1} R_{12} \\ \hline 0 & 0 \end{array}\right] S^T . \tag{A-39}$$

This is clearly nonsymmetric and therefore does not constitute an orthogonal projector. Thus, the g-inverse formed from the truncated QR factorization does not conform to the requirements for the pseudoinverse.

A.8 Complete Orthogonal Factorization of Rank Deficient Matrices

To obtain a minimum norm least-squares solution in the rank deficient case, we use an extension of the QR factorization, the complete orthogonal factorization. This factorization is a suitable alternative to the singular value decomposition for performing the rank reduction necessary to obtain the minimum norm linear least-squares solution.

Again consider the NXM matrix A with rank r<M≤N and the orthogonal factorization

$$Q\,A\,S = R \equiv \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & 0 \end{array}\right],$$

where $R_{11} = \left[\begin{array}{c} \diagdown \\ 0 \end{array}\right]_{r\times r}$.

There exists an MXM orthogonal matrix V such that

$$R\,V = Q\,A\,S\,V = \tilde{R} \equiv \left[\begin{array}{c|c} \tilde{R}_{11} & 0 \\ \hline 0 & 0 \end{array}\right], \qquad \text{(A-40)}$$

where $\tilde{R}_{11} = \left[\begin{array}{c} \diagdown \\ 0 \end{array}\right]_{r\times r}$.

From this, we may write $A = Q^T \tilde{R} V^T S^T$ and define the g-inverse as

$$A^+ = S\,V \left[\begin{array}{c|c} \tilde{R}_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q\,. \qquad \text{(A-41)}$$

Now forming the projection operator P_A, we obtain

$$
\begin{aligned}
P_A &= A\,A^{+} \\
&= Q^T \left[\begin{array}{c|c} \tilde{R}_{11} & 0 \\ \hline 0 & 0 \end{array}\right] V^T S^T \; S\,V \left[\begin{array}{c|c} \tilde{R}_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q \\
&= Q^T \left[\begin{array}{c|c} I_r & 0 \\ \hline 0 & 0 \end{array}\right] Q \,, \qquad \text{(A-42)}
\end{aligned}
$$

which, once again, is seen to be symmetric and idempotent. If we now attempt to form the projector onto the row space of A, we get

$$
\begin{aligned}
{}_A P &= A^{+}\,A \\
&= S\,V \left[\begin{array}{c|c} \tilde{R}_{11}^{-1} & 0 \\ \hline 0 & 0 \end{array}\right] Q\,Q^T \left[\begin{array}{c|c} \tilde{R}_{11} & 0 \\ \hline 0 & 0 \end{array}\right] V^T\,S^T \\
&= S\,V \left[\begin{array}{c|c} I_r & 0 \\ \hline 0 & 0 \end{array}\right] V^T S^T \,, \qquad \text{(A-43)}
\end{aligned}
$$

which is also symmetric and idempotent. Thus the complete orthogonal factorization leads to a g-inverse, which is suitable for forming both projection operators. This g-inverse therefore satisfies all of the requirements for the Moore-Penrose generalized inverse.

Using the g-inverse constructed from the complete orthogonal factorization, the least-squares functional becomes

$$\phi(\underline{x}) = \left|\left| \mathbf{A}\,\underline{x} - \underline{y} \right|\right|^2$$

$$= \left|\left| \mathbf{Q}\,\mathbf{A}\,\underline{x} - \mathbf{Q}\,\underline{y} \right|\right|^2$$

$$= \left|\left| \mathbf{Q}\,\mathbf{A}\,\mathbf{S}\,\mathbf{S}^T\,\underline{x} - \mathbf{Q}\,\underline{y} \right|\right|^2$$

$$= \left|\left| \mathbf{Q}\,\mathbf{A}\,\mathbf{S}\,\mathbf{V}\,\mathbf{V}^T\mathbf{S}^T\,\underline{x} - \mathbf{Q}\,\underline{y} \right|\right|^2$$

$$= \left|\left| \tilde{\mathbf{R}}\,\mathbf{V}^T\mathbf{S}^T\,\underline{x} - \mathbf{Q}\,\underline{y} \right|\right|^2 .$$

If we let $\underline{w} = \mathbf{S}^T\,\underline{x}$ partition $\mathbf{Q}$ as

$$\mathbf{Q} = \left[\begin{array}{c} \mathbf{Q}_1 \\ \hline \mathbf{Q}_2 \end{array}\right] \begin{array}{l} \} \, r \\ \} \, N-r \end{array} ,$$

and partition $\mathbf{V}$ as

$$\mathbf{V} = \left[\, \mathbf{V}_1 \;\middle|\; \mathbf{V}_2 \,\right] ,$$

where $\mathbf{V}_1$ is $M \times r$ and $\mathbf{V}_2$ is $M \times (M-r)$, then we obtain for the least-squares functional

$$\phi(\underline{w}) = \left|\left| \left[\begin{array}{c|c} \tilde{\mathbf{R}}_{11} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c} \mathbf{V}_1^T \\ \hline \mathbf{V}_2^T \end{array}\right] \underline{w} - \left[\begin{array}{c} \mathbf{Q}_1 \\ \hline \mathbf{Q}_2 \end{array}\right] \underline{y} \right|\right|^2$$

$$= \left|\left| \left[\begin{array}{c} \tilde{\mathbf{R}}_{11}\,\mathbf{V}_1^T \\ \hline \mathbf{0} \end{array}\right] \underline{w} - \left[\begin{array}{c} \mathbf{Q}_1 \\ \hline \mathbf{Q}_2 \end{array}\right] \underline{y} \right|\right|^2 .$$

From this, we obtain the solution

$$\underline{w}_1 = V_1 \tilde{R}_{11}^{-1} Q_1 \underline{Y} \cdot \tag{A-44}$$

We obtain a final minimum norm solution as

$$\underline{x}_{LS} = S V_1 \tilde{R}_{11}^{-1} Q_1 \underline{Y} , \tag{A-45}$$

which, as in the full rank case, leaves a residual sum of squares

$$\phi(\underline{x}_{LS}) = \left|\left| Q_2 \underline{Y} \right|\right|^2 . \tag{A-46}$$

A.9 Near Rank Deficient Matrices

Consider the N×M matrix A with numerical rank $\rho = M \leq N$, but whose expected (ideal rank) is $r \leq M$. There exists an N×N orthogonal matrix Q and an M×M permutation matrix S, such that

$$Q A S = R \equiv \begin{bmatrix} R_1 \\ \hline 0 \end{bmatrix} , \tag{A-47}$$

where

$$R_1 = \begin{bmatrix} \diagdown & \\ & 0 \end{bmatrix}_{M \times M} .$$

Column pivoting at each stage of the factorization will result in a matrix that can be further partitioned as

$$R_1 = \left[\begin{array}{c|c} R_{11} & R_{12} \\ \hline 0 & R_{22} \end{array}\right] ,$$

where

$$R_{11} = \begin{bmatrix} \diagdown & \\ & 0 \end{bmatrix}_{r \times r}$$

and $$R_{22} = \begin{bmatrix} 0 & \diagdown \\ & \end{bmatrix}_{(M-r)\times(M-r)} .$$

If **A** were truly rank deficient, R_{22} would consist of zeros. But because of perturbations in **A**, R_{22} will have non-zero elements. If the perturbations are small, however, then the elements of R_{22} should also be small so that the rank deficiency can be uncovered when $||R_{22}||$ becomes much smaller than $||A||$. Then rank reduction can be achieved by setting R_{22} to zero and solving the remainder of the problem as a truly rank deficient case.

Golub and Van Loan [12] point out that there are cases in which at no step during the orthogonalization process is the norm of R_{22} very small, even though the original matrix is rank deficient. But they also go on to say that this method of rank determination "works well in practice." The reader is referred to Section 6.4 of Golub and Van Loan [12], and to Golub, Klema, and Stewart [14].

(Blank Page)

APPENDIX B - ORTHOGONAL FACTORIZATION BY SUCCESSIVE HOUSEHOLDER TRANSFORMATION

In this appendix, we describe a method of triangularizing matrices which uses the highly stable Householder transformation, also called an elementary reflector (see Ref. 15). This transformation introduces zeros into a column or row of a matrix, depending on whether it is applied from the left (premultiplication) or from the right (postmultiplication). In the former case, we premultiply by a sequence of elementary reflectors to transform an N×M matrix (N>M) into an upper triangular matrix (or upper trapezoidal matrix if N<M). This leads to the **QR** factorization defined as $\mathbf{Q}\,\mathbf{A} = \mathbf{R}$. In the latter case, we transform an N×M matrix (N<M) into a lower triangular matrix (or lower trapezoidal if N>M). This can lead to the **RV** factorization defined as $\mathbf{A}\,\mathbf{V} = \mathbf{R}$, or to the complete orthogonal factorization as discussed in Appendix A.

B.1 Householder Transformation from the Left

Given a full rank N×M matrix **A**, where N>M, we premultiply **A** with asequence of M elementary matrices; i.e., we form $\mathbf{H}_M \cdots \mathbf{H}_2\,\mathbf{H}_1\,\mathbf{A} = \mathbf{R}$, to yield the N×M upper triangular matrix **R**. The $\mathbf{H}_i$ are defined as

$$\mathbf{H}_i = \left[\begin{array}{c|c} \mathbf{I}_{i-1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{U}_i \end{array}\right],$$

where $\mathbf{I}_{(i-1)}$ is an (i-1)×(i-1) identity matrix and $\mathbf{U}_i$ is an (N-i+1)×(N-i+1) Householder reflector matrix constructed to introduce zeros below the i'th diagonal element of $\mathbf{A}^{(i-1)} = \mathbf{H}_{i-1} \cdots \mathbf{H}_1\,\mathbf{A}$. For example, at the beginning of the fourth stage of the triangulari ion process, we have the matrix

$$A^{(3)} = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} & a_{15}^{(1)} & \cdots & a_{1M}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & a_{24}^{(2)} & a_{25}^{(2)} & \cdots & a_{2M}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & a_{34}^{(3)} & a_{35}^{(3)} & \cdots & a_{3M}^{(3)} \\ 0 & 0 & 0 & a_{44}^{(3)} & a_{45}^{(3)} & \cdots & a_{4M}^{(3)} \\ 0 & 0 & 0 & a_{54}^{(3)} & a_{55}^{(3)} & \cdots & a_{5M}^{(3)} \\ 0 & 0 & 0 & a_{64}^{(3)} & a_{65}^{(3)} & \cdots & a_{6M}^{(3)} \\ 0 & 0 & 0 & a_{74}^{(3)} & a_{75}^{(3)} & \cdots & a_{7M}^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & a_{N4}^{(3)} & a_{N5}^{(3)} & \cdots & a_{NM}^{(3)} \end{bmatrix} .$$

Here the superscripts on the elements designate the number of previous Householder reflectors that have transformed the particular element; e.g., the elements in the first row are affected only by the first Householder reflector so that these elements have the superscript (1), the elements in the second row are affected by the first two Householder reflectors so that these elements have the superscript (2), and the nonzero elements below the second row have been transformed by all three of the previous reflectors so that these elements have the superscript (3).

We wish to introduce zeros below the first element of the vector designated within the box. If we denote this (N-i+1)-vector as $\underline{x}$, then we we wish to find the Householder reflector, U, such that

$$U \underline{x} = - \sigma \underline{e}_1, \qquad \text{(B-1)}$$

where σ is a scalar, and

$$\underline{e}_1 = \left[\begin{array}{ccccc} 1 & 0 & 0 & 0 & \cdots & 0 \end{array} \right]^T$$

is the first vector in the standard basis for the space $\mathbb{R}^{N-i+1}$.

The Householder reflector that achieves this is defined as

$$\mathbf{U} = \mathbf{I} - \beta^{-1} \underline{u} \underline{u}^T , \tag{B-2}$$

where

$$\sigma = \mathrm{sgn}(x_1) \, ||\underline{x}|| , \tag{B-3}$$

$$u_1 = x_1 + \sigma , \tag{B-4}$$

$$u_i = x_i , \; i=2,\ldots,n , \tag{B-5}$$

and

$$\beta = \sigma \, u_1 . \tag{B-6}$$

Since only the vector $\underline{u}_i$ and the scalar β_i are necessary for forming $\mathbf{H}_i$ at each stage, all information concerning the construction of the $\mathbf{H}_i$ can be saved by storing the last N-i elements of $\underline{u}_i$ below the i'th diagonal element of A and storing the pre-transformation value of the diagonal element in an auxiliary vector (note that the post-transformation value of the diagonal element is σ_i, so that β_i is indirectly available).

When applying the Householder transformation to an arbitrary vector $\underline{y}$ we use the equation

$$\begin{aligned} \mathbf{U} \underline{y} &= \left(\mathbf{I} - \beta^{-1} \underline{u} \underline{u}^T \right) \underline{y} \\ &= \underline{y} - \beta^{-1} \left(\underline{u}^T \underline{y} \right) \underline{u} . \end{aligned} \tag{B-7}$$

Since this is constructed directly from the vector $\underline{u}$ and the scalar β, we need never explicitly form the elementary matrices $\mathbf{H}_i$.

After Householder reflectors have been constructed for all M columns, the orthogonal matrix Q is defined as

$$Q = H_M \cdots H_2 H_1 . \tag{B-8}$$

Since the H_i are elementary matrices, they are symmetric, so that the transpose of Q is just

$$Q^T = H_1 H_2 \cdots H_M . \tag{B-9}$$

We can therefore multiply by the transpose of Q simply by reversing the order in which each of the elementary transformations are applied. Thus one need never explicitly form the matrix Q.

The stability of the Householder method can be ensured by using column pivoting at each stage of the factorization to bring the column of largest norm to the pivot position. At each stage, the pivoting causes an elementary matrix to be factored to the right; i.e., each column swap is recorded by postmultiplying by an identity matrix with the same two columns interchanged. At the i'th step, we would form $A^{(i)} = H_i A^{(i-1)} S_i$, where S_i is an elementary matrix representing the i'th column pivot. At completion, the factorization appears as $Q A S = H_M \cdots H_2 H_1 A S_1 S_2 \cdots S_M = R$.

B.2 Householder Transformation from the Right

Given an NXM matrix A with full row rank, we postmultiply A with asequence of elementary matrices to yield $A H_1 H_2 \cdots H_M = R$. Here, the H_i are formed in exactly the same way as when transforming from the left, except that U_i is constructed to introduce zeros to the right of the i'th diagonal element; i.e., into the i'th row.

To parallel our previous example, at the i'th stage of the triangularization process, we have the matrix

$$A^{(3)} = \begin{bmatrix} a_{11}^{(1)} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ a_{21}^{(1)} & a_{22}^{(2)} & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ a_{31}^{(1)} & a_{32}^{(2)} & \boxed{a_{33}^{(3)} & a_{34}^{(3)} & a_{35}^{(3)} & a_{36}^{(3)} & a_{37}^{(3)} & \cdots & a_{3M}^{(3)}} \\ a_{41}^{(1)} & a_{42}^{(2)} & a_{43}^{(3)} & a_{44}^{(3)} & a_{45}^{(3)} & a_{46}^{(3)} & a_{47}^{(3)} & \cdots & a_{48}^{(3)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ a_{N1}^{(1)} & a_{N2}^{(2)} & a_{N3}^{(3)} & a_{N4}^{(3)} & a_{N5}^{(3)} & a_{N6}^{(3)} & a_{N7}^{(3)} & \cdots & a_{NM}^{(3)} \end{bmatrix} .$$

Here, we construct the Householder reflector to introduce zeros to the right of the first element in the vector designated within the box. Denoting this (M-i+1)-vector as $\underline{x}^T$, we wish to find the Householder reflector U, such that

$$\underline{x}^T U = - \sigma \, \underline{e}_1^T , \tag{B-10}$$

where σ and $\underline{e}_1$ are the same as in Section B.1. The construction of U is also the same as in Eq. (B.1).

In applying the Householder reflector to an arbitrary row vector $\underline{y}^T$, we use the equation

$$\begin{aligned} \underline{y}^T U &= \underline{y}^T \left(I - \beta^{-1} \, \underline{u} \, \underline{u}^T \right) \\ &= \underline{y}^T - \beta^{-1} \left(\underline{y}^T \underline{u} \right) \underline{u}^T . \end{aligned} \tag{B-11}$$

Row pivoting could be used to ensure stability, just as column pivoting was used in the previous section. Usually, however, transformations from the right are applied to matrices that already exhibit special structure and are expected to be reasonably conditioned, so that row pivoting would serve only to disturb the existing structure.

After Householder reflectors have been constructed for all N rows, the orthogonal matrix V is defined as

$$V = H_1 H_2 \cdots H_M . \tag{B-12}$$

Again, since the H_i are symmetric, the transpose of V is

$$V^T = H_M \cdots H_2 H_1 . \tag{B-13}$$

Essentially, applying orthogonal transformations from the right of a matrix is the same as applying transformations from the left of the transpose of the same matrix, and then transposing the entire equation. First, $Q A^T = R_u$, where R_u denotes an upper triangular matrix. Then, transposing yields $A Q^T = R_u^T$. Now, letting $V = Q^T$ and $R_1 = R_u^T$ (a lower triangular matrix), we get our desired factorization.

APPENDIX C - REVIEW OF LEAST-SQUARES OPTIMIZATION TECHNIQUES

In nonlinear optimization (also see Refs. 16 and 17) we begin with an initial set of parameters and, through a succession of iterations, update the parameters in a way such that the sequence of updated parameters hopefully converges to the ideal set. We go about this by assigning a measure of closeness between a model of the physical phenomenon, which is a function of the parameter set, and the experimentally observed values. In the case of least-squares, we use as that measure the sum of the squares of the errors between our parameterized model and the observations; i.e., we assign the cost functional (or error norm)

$$\Phi(\underline{\theta}) = \left|\left| \underline{e}(\underline{\theta}) \right|\right|^2 = \left|\left| \underline{y} - \underline{x}(\underline{\theta}) \right|\right|^2 , \qquad \text{(C-1)}$$

where $\underline{\theta}$ is the set of parameters, $\underline{y}$ is the vector of observations, and $\underline{x}(\underline{\theta})$ is the parameterized model. We then go about minimizing this cost functional by appropriately adjusting the parameter vector, $\underline{\theta}$.

In this appendix, we introduce three optimization techniques that can be used to minimize the least-squares functional. They are the method of steepest descent, Newton's method, and the Gauss-Newton method. We first will introduce a general class of strategies called gradient methods, within which the three methods mentioned fit. The steepest descent method follows directly from the acceptability criteria for the gradient methods. We then introduce Newton's method as a second-order Taylor approximation to the error functional, and finally we introduce the Gauss-Newton method as a simplification of Newton's method.

C.1 Gradient Methods of Optimization

At each iteration the *adjustment*, or update, consists of a direction and a step size; i.e., the update is

$$\Delta = \rho \, \underline{d} , \qquad \text{(C-2)}$$

where ρ is the step size and $\underline{d}$ is the step direction. To assure that this update can lead to a decrease in the cost functional, we must first require that the step direction $\underline{d}$ form greater than a 90° angle with the gradient (be *downhill* on the contour of Φ) at the current iterate of parameters $\underline{\theta}^{(\iota)}$.

To ascertain this, we first note that given a direction $\underline{d}$, the updated parameter vector is solely a function of the step size; i.e., $\underline{\theta}(\rho) = \underline{\theta}^{(\iota)} + \rho\ \underline{d}$. The cost functional along this direction is then

$$\Phi_d\{\underline{\theta}(\rho)\} = \Phi\left\{ \underline{\theta}^{(\iota)} + \rho\ \underline{d} \right\} .$$

Differentiating this with repect to ρ and evaluating at $\rho = 0$ yields the directional derivative of the cost functional at the current iterate; i.e.,

$$\Phi_d' = \left. \frac{\partial \Phi_d\{\underline{\theta}(\rho)\}}{\partial \rho} \right|_{\rho = 0} .$$

Noting the chain rule, this becomes

$$\Phi_d' = \left[\frac{\partial \Phi_d\{\underline{\theta}(\rho)\}}{\partial \underline{\theta}(\rho)} \right]^T \left[\frac{\partial \underline{\theta}(\rho)}{\partial \rho} \right] \Bigg|_{\rho = 0}$$

$$= \underline{g}^{(\iota)^T}\ \underline{d} ,$$

where $\underline{g}^{(\iota)}$ is the gradient of $\Phi(\underline{\theta})$ at $\underline{\theta}^{(\iota)}$. The step direction $\underline{d}$ is then *downhill* if the directional derivative is negative; i.e.,

$$\Phi_d' = \underline{g}^T\ \underline{d} < 0 , \qquad \text{(C-3)}$$

where we have dropped the iteration index on the gradient vector. One way to assure this is to let the step direction be

$$\underline{d} = -\ \underline{g} , \qquad \text{(C-4)}$$

which is precisely the case in the method of steepest descent--so called because initially, this is the direction in which the cost functional descends most rapidly. Steepest descent is well known, however, for its slow convergence due to a zig-zag pattern along troughs in the cost functional contour. Therefore, an alternative is desirable.

Another way to assure that the directional derivative is negative is to find a positive definite matrix, **R**, and let

$$\underline{d} = - \mathbf{R}\, \underline{g} \ . \tag{C-5}$$

An optimization technique in which the direction is so chosen, regardless of whether the matrix **R** is positive definite, is called a gradient method. If the matrix **R** is strictly positive definite, then $\underline{d}$ is called an acceptable gradient direction. For all gradient directions, the directional derivative is given by the quadratic form

$$\Phi_d' = - \underline{g}^T \mathbf{R}\, \underline{g} \ . \tag{C-6}$$

C.2 Newton's Method

In Newton's method, we choose for the matrix **R** the inverse of the Hessian matrix of the error functional at the current iterate. The typical element of the Hessian matrix is

$$H_{ij}^{(\iota)} = \left. \frac{\partial^2 \Phi(\underline{\theta})}{\partial \theta_j \partial \theta_i} \right|_{\underline{\theta} = \underline{\theta}^{(\iota)}} .$$

This choice for the matrix **R** arises by approximating the error functional with a second-order Taylor polynomial

$$P_2(\underline{\theta}) = \Phi\left(\underline{\theta}^{(\iota)} \right) + \underline{g}^{(\iota)^T} \left[\underline{\theta} - \underline{\theta}^{(\iota)} \right] + \frac{1}{2} \left[\underline{\theta} - \underline{\theta}^{(\iota)} \right] \mathbf{H}^{(\iota)} \left[\underline{\theta} - \underline{\theta}^{(\iota)} \right] ,$$

and by optimizing this approximate cost functional over the parameters; i.e.,

differentiating with respect to $\underline{\theta}$ and equating the result to zero to yield the optimum parameter set

$$\underline{\theta}_{opt} = \underline{\theta}^{(\iota)} - \left[\mathbf{H}^{(\iota)} \right]^{-1} \underline{g}^{(\iota)} \quad . \tag{C-7}$$

If the Hessian matrix is positive definite, then its inverse is also positive definite and Newton's method yields an acceptable direction. Furthermore, if the objective functional is quadratic, the Taylor approximation is exact and Newton's method will converge in a single iteration. Even if the error surface is not quadratic but is nearly so (as is often the case in the neighborhood of the ideal parameters), then Newton's method offers quadratic convergence without the need for a line search to determine step size.

C.3 The Gradient of the Least-Squares Error Norm

At this point, it will be useful to obtain an expression for the gradient of the least-squares error functional. The arbitrary least-squares functional can be written as

$$\Phi(\underline{\theta}) = \left[\underline{e}(\underline{\theta}) \right]^T \underline{e}(\underline{\theta}) \quad . \tag{C-8}$$

A typical term in the gradient is

$$g_i = D_i \left[\underline{e}^T \underline{e} \right] \quad .$$

Thus

$$g_i = D_i(\underline{e}^T)\ \underline{e} + \underline{e}^T\ D_i(\underline{e}) \quad , \tag{C-9}$$

$$g_i = 2\ D_i(\underline{e}^T)\ \underline{e} \quad , \tag{C-10}$$

and $$g_i = 2\ \underline{e}^T\ D_i(\underline{e})\ , \qquad \text{(C-11)}$$

where D_i is the operator that performs partial differentiation with respect to θ_i and where we have dropped the iteration index, as well as the explicit dependence of $\underline{e}$ on $\underline{\theta}$. The equality of the last three expressions follows from the fact that both terms in Eq. (C-9) represent the same scalar product (since the error vector is real). In further derivations, we will use either Eq. (C-10) or Eq. (C-11), depending on which is more convenient for the given purpose.

C.4 The Gauss-Newton Method

If we differentiate (C-11) with respect to the j'th parameter, we get the ij'th component of the Hessian matrix as

$$H_{ij} = 2\ D_i(\underline{e}^T)\ D_j(\underline{e}) + 2\ \underline{e}^T\ D_{ij}^2(\underline{e})\ , \qquad \text{(C-12)}$$

where $D_{ij}^2(\underline{e})$ is the second partial derivative of the error vector with respect to the parameters θ_i and θ_j, or the second-order sensitivity derivative. (Similarly, $D_i(\underline{e})$ is the first-order sensitivity derivative. The names reflect the fact that these quantities measure the sensitivity of the parameterized model to changes in the parameters.) If we assume that the error vector is small in the neighborhood in which we are optimizing, then we can neglect the term involving the second-order sensitivity derivative and approximate the Hessian matrix as

$$N_{ij} = 2\ D_i(\underline{e}^T)\ D_j(\underline{e})\ . \qquad \text{(C-13)}$$

The equation for the approximate Newton direction; i.e., $\underline{d} = -N^{-1}g$, can be rewritten as the system of equations

$$N\ \underline{d} = -\ g. \qquad \text{(C-14)}$$

If we now define the Jacobian matrix of the error vector, given by

$$J = \left[\; D_1(\underline{e}) \; \middle| \; D_2(\underline{e}) \; \middle| \; \cdots \; \middle| \; D_K(\underline{e}) \; \right] ,$$

then we can rewrite the gradient, from Eq. (C-10), as

$$\underline{g} = 2 \, J^T \, \underline{e} \; , \tag{C-15}$$

and we can write the Gauss approximation to the Hessian matrix as

$$N = 2 \, J^T \, J \; . \tag{C-16}$$

Note that the matrix N is always at least positive semidefinite, so that the concerns of the Hessian being negative definite or indefinite are alleviated with this approximation. Substituting Eqs. (C-15) and (C-16) into Eq. (C-14), the system of equations for the Gauss-Newton direction is then

$$J^T \, J \, \underline{d} = - \, J^T \, \underline{e} \; . \tag{C-17}$$

But this is just the set of normal equations for the linear least-squares problem in which the error vector is projected onto the columns of the Jacobian matrix; i.e.,

$$J \, \underline{d} \simeq \underline{e} \; . \tag{C-18}$$

The solution for this problem is

$$\underline{d} = {}^{+}- \, J \;\; \underline{e} \; , \tag{C-19}$$

where J^+ is the pseudoinverse of the Jacobian matrix. Thus the Gauss-Newton method can be viewed as a sequence of linear approximations to the cost functional. Note that, unlike Newton's method, this method does require a line search to find an optimum step size ρ.

While there are several other popular optimization techniques for least-squares, including a number of variations on Newton's method that use other methods to ensure the positive definiteness of the Hessian matrix (e.g., the Marquardt method, the Greenstadt method, etc.), we will confine our attention to the three methods mentioned.

(Blank Page)

APPENDIX D - DERIVATIVES OF PROJECTORS AND PSEUDOINVERSES

In this appendix, we parallel Golub and Pereyra [1] in deriving the derivative of the projection operators (both onto the column space of a matrix and onto the row space of a matrix) and the derivative of the pseudoinverse. As noted in Chapter 5 of the text, we retain the index of differentiation in all derivations so that the final results are expressions for partial derivatives.

D.1 The Derivative of the Projection Operator

To calculate the derivatives of the variable projection functionals, we need a method for calculating the derivative of the projection operators $D_i(P_A)$ and $D_i({}_AP)$. We proceed first in evaluating $D_i(P_A)$ by noting that P_A is idempotent, so that

$$\begin{aligned} D_i(P_A) &= D_i(P_A\, P_A) \\ &= D_i(P_A)\, P_A + P_A\, D_i(P_A) \ . \end{aligned} \tag{D-1}$$

Thus the problem becomes one of evaluating the two terms on the right-hand side of Eq. (D-1). We start with the first term by noting that $A = P_A\, A$ so that

$$\begin{aligned} D_i(A) &= D_i(P_A A) \\ &= D_i(P_A)\, A + P_A\, D_i(A) \ . \end{aligned} \tag{D-2}$$

Rearranging yields

$$\begin{aligned} D_i(P_A)\, A &= D_i(A) - P_A\, D_i(A) \\ &= P_A^{\perp}\, D_i(A) . \end{aligned} \tag{D-3}$$

Postmultiplying both sides of this equation by A^+ and noting, on the far left hand side, that $A\,A^+ = P_A$, we obtain the first of the desired expressions,

$$D_i(P_A)\,P_A = P_A^{\perp}\,D(A)\,A^+. \tag{D-4}$$

We proceed with the second term by noting that the projection operator, and hence its partial derivative, is symmetric, so that

$$\left[\,D_i(P_A)\,P_A\,\right]^T = P_A\,D_i(P_A)\,.$$

Substituting Eq. (D-4) into the above equation yields the desired result,

$$\begin{aligned} P_A\,D_i(P_A) &= \left[\,P_A^{\perp}\,D_i(A)\,A^+\,\right]^T \\ &= (A^+)^T\,D_i(A^T)\,P_A^{\perp}\,. \end{aligned} \tag{D-5}$$

Substituting Eqs. (D-4) and (D-5) back into Eq. (D-1) yields the partial derivative of the projection operator, given by

$$D_i(P_A) = P_A^{\perp}\,D_i(A)\,A^+ + (A^+)^T\,D_i(A^T)\,P_A^{\perp}\,. \tag{D-6}$$

Since the row space of a matrix is the same as the column space of the transpose of the matrix, to obtain an expression for the derivative of the row space projector, we merely replace A by its transpose everywhere in Eq. (D-6), which yields

$$\begin{aligned} D_i({}_AP) &= {}_AP^{\perp}\,D_i(A^T)\,(A^+)^T + \left[\,{}_AP^{\perp}\,D_i(A^T)\,(A^+)^T\,\right]^T \\ &= A^+\,D_i(A)\,{}_AP^{\perp} + \left[\,A^+\,D_i(A)\,{}_AP^{\perp}\,\right]^T. \end{aligned} \tag{D-7}$$

D.2 The Derivative of the Pseudoinverse

We now provide an expression for the partial derivative of the pseudoinverse of a matrix, which is necessary for deriving the Hessian matrix of the VPF. We begin by noting that

$$I = P_A + P_A^{\perp},$$

so that

$$D_i(A^+) = D_i(A^+)\, P_A + D_i(A^+)\, P_A^{\perp}. \tag{D-8}$$

We now wish to find expressions for both terms on the right-hand side of Eq. (D-8). Looking at the first of these two terms, we begin by recalling that $A^+ = A^+ A A^+$ so that $D_i(A^+) = D_i(A^+)\, P_A + A^+ D_i(A)\, A^+ + {}_AP\, D_i(A^+)$. Rearranging yields

$$D_i(A^+)\, P_A = {}_AP^{\perp}\, D_i(A^+) - A^+ D_i(A)\, A^+. \tag{D-9}$$

We now need a manageable expression for the first term on the right-hand side of this expression. To obtain this, we again start with

$$A^+ = A^+ A A^+$$

$$= {}_AP\, A^+.$$

Then

$$D_i(A^+) = D_i({}_AP)\, A^+ + {}_AP\, D_i(A^+). \tag{D-10}$$

Rearranging yields

$${}_AP^{\perp}\, D_i(A^+) = D_i({}_AP)\, A^+. \tag{D-11}$$

Substituting Eq. (D-7) into Eq. (D-10), we obtain

$$ {}_{A}P^{\perp}\, D_i(A^+) \;=\; A^+\, D_i(A)\, {}_{A}P^{\perp}\, A^+ + \left[A^+\, D_i(A)\, {}_{A}P^{\perp} \right]^T A^+ . $$

But

$$ {}_{A}P^{\perp}\, A^+ = \left[I - A^+ A \right] A^+ $$

$$ = A^+ - A^+ A A^+ \;=\; 0 , $$

so we get

$$ {}_{A}P^{\perp}\, D_i(A^+) \;=\; \left[A^+\, D_i(A)\, {}_{A}P^{\perp} \right]^T A^+ . \qquad \text{(D-12)} $$

Transposing on the right and substituting this back into Eq. (D-9) yields

$$ D_i(A^+)\, P_A = {}_{A}P^{\perp}\, D_i(A^T)\, (A^+)^T\, A^+ - A^+\, D_i(A)\, A^+ . \qquad \text{(D-13)} $$

We now turn our attention to the second term in Eq. (D-8). We proceed by writing

$$ A^+ = A^+ A A^+ $$

$$ = A^+ P_A , $$

so that

$$ D_i(A^+) = D_i(A^+)\, P_A + A^+\, D_i(P_A) . \qquad \text{(D-14)} $$

Rearranging and substituting Eq. (D-6) in the rightmost term, we obtain

$$ D_i(A^+)\, P_A^{\perp} \;=\; A^+\, P_A^{\perp}\, D_i(A)\, A^+ + A^+ \left[P_A^{\perp}\, D_i(A)\, A^+ \right]^T . \qquad \text{(D-15)} $$

But
$$A^+ P_A^{\perp} = A^+ \left(I - A^+ A \right)$$

$$= A^+ - A^+ A A^+ = 0 ,$$

so we get
$$D_i(A^+) P_A^{\perp} = A^+ (A^+)^T D_i(A^T) P_A^{\perp}. \qquad \text{(D-16)}$$

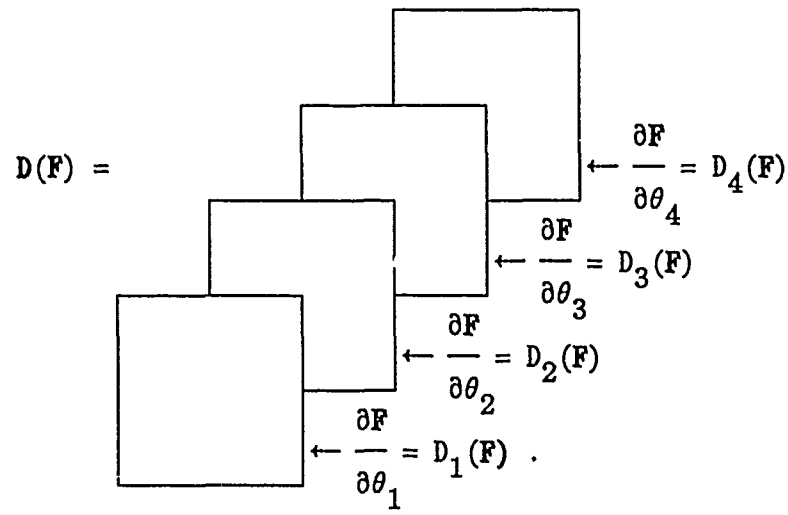
Finally, we obtain the partial derivative of the pseudoinverse by substituting Eqs. (D-13) and (D-16) back into Eq. (D-8), which yields

$$D_i(A^+) = {}_A P^{\perp} D_i(A^T) (A^+)^T A^+ - A^+ D_i(A) A^+ + A^+ (A^+)^T D_i(A^T) P_A^{\perp}. \qquad \text{(D-17)}$$

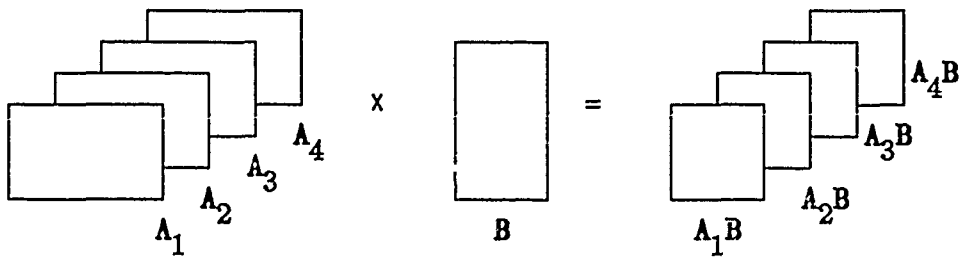
(Blank Page)

APPENDIX E - FRECHET DERIVATIVES AND A SIMPLIFIED TENSOR NOTATION

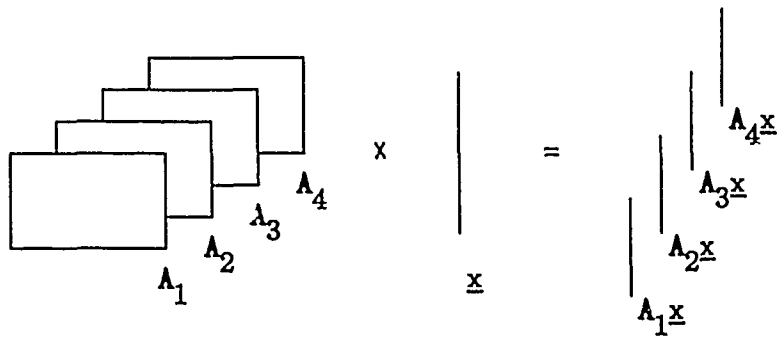
Here we examine the simplified tensor notation of Golub and Pereyra [1] and show the difficulty in carrying this notation on to higher order derivatives. First, we define the Frechet derivative of the basis function matrix, which is a three-dimensional array consisting of the partial derivatives of the basis function matrix with respect to each of the signal pole parameters. Consider the example case of a model with two conjugate pole pairs. There are then four signal pole parameters, and the Frechet derivative of the basis function matrix [denoted by **D(F)**] will appear as



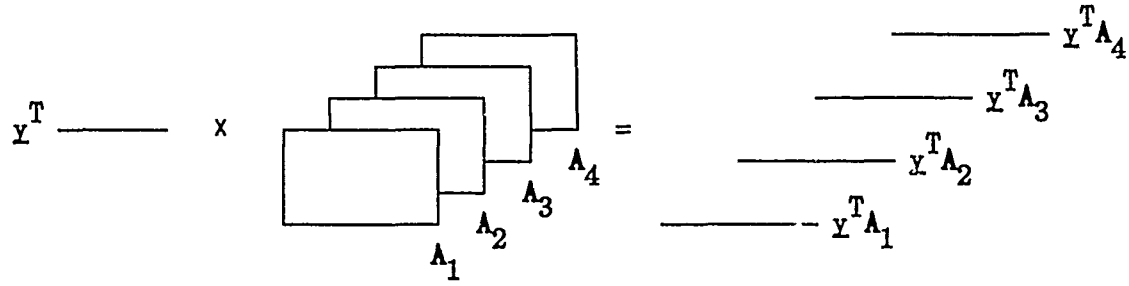
Multiplication of this *tensor* with a matrix is achieved by multiplying each partial derivative matrix with the multiplicand matrix, yielding another three-dimensional array as illustrated below.



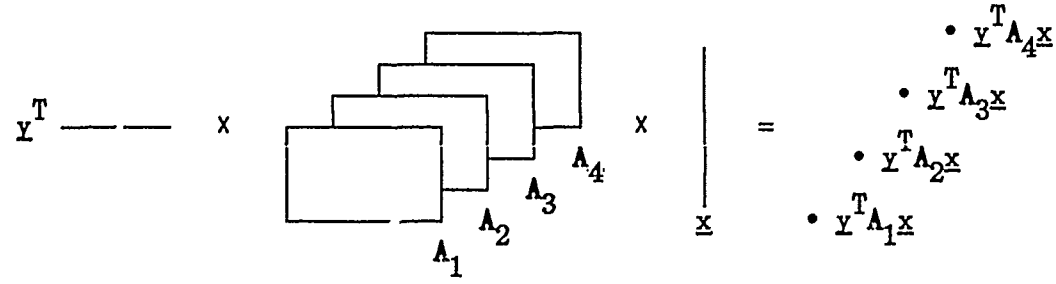
Multiplication of the tensor with a vector results in what we will call a "degenerate tensor of valence two." There are two cases in which this occurs. When the valence three tensor premultiplies a column vector, the result appears as follows.



When a row vector premultiplies the valence three tensor, the result appears as follows.



If we pre- and post-multiply the valence three tensor with a row vector and a column vector, respectively, then we obtain what we will call a "degenerate tensor of valence one." This result appears as follows.



That the results in the last three cases are not vectors and matrices as we normally think of them should be clear from the illustrations. In the last case in particular, this valence one tensor is neither a row vector nor a column vector in the usual sense; i.e., in the plane of the paper. This last case precisely describes the gradient vector.

The real problem occurs, however, when one wishes to form the Hessian matrix by again differentiating the gradient *vector* with respect to the parameter vector. When we performed the differentiation above, the partial derivatives were lined up in a *third dimension* that was not along a column or a row. Now we wish to differentiate a valence one tensor whose elements already lie along this *third dimension*, so that we must now line up the second partial derivatives along some *fourth dimension*. Additional notation will, at this point, be necessary in order to distinguish which *dimension* we are dealing with. This problem becomes critical when attempting to multiply two of these valence three tensors (one in the *third dimension* and one in the *fourth dimension*), as occurs when forming the Hessian. One could introduce notation to keep track of these different *dimensions*, or one could abandon this now not-so-simplified notation and adopt the full tensor notation. Or one could simply treat each of the derivatives as a group of partial derivatives, retaining the indexing, and ignore the specifics of the higher order vector calculus, as we have chosen to do.

(Blank Page)

APPENDIX F - PROOF THAT OPTIMIZATION OF THE VARIABLE PROJECTION FUNCTIONS LEADS TO THE SAME MINIMA AS OPTIMIZATION OF THE LEAST-SQUARES FUNCTIONAL

In this appendix, we expound upon the proof by Golub and Pereyra [1] that sequential optimization of the signal poles and amplitudes via the variable projection functional leads to the same optimum values as does the simultaneous optimization of the parameters via the least-squares functional. For simplicity in the proof, we will follow the notation of Golub and Pereyra and introduce the Frechet derivative D(F), which is an array of partial derivative matrices. This derivative is described further in Appendix E.

Recall the least-squares cost functional and error vector, given by

$$\phi(\underline{\theta},\underline{a}) = \left|\left| \underline{e}(\underline{\theta},\underline{a}) \right|\right|^2$$

and

$$\underline{e}(\underline{\theta},\underline{a}) = \underline{y} - \mathbf{F}(\underline{\theta})\ \underline{a}\ .$$

Let us now define the Jacobian matrix of the least-squares error vector,

$$\mathbf{J}_{\underline{\theta},\underline{a}} = \left[\ \mathbf{J}_{\underline{\theta}}\ ,\ \mathbf{J}_{\underline{a}}\ \right]\ , \tag{F-1}$$

where

$$\mathbf{J}_{\underline{\theta}} = \frac{\partial \underline{e}}{\partial \underline{\theta}^T} = -\ \mathrm{D}\left[\ \mathbf{F}(\underline{\theta})\ \right]\ \underline{a}\ , \tag{F-2}$$

and

$$\mathbf{J}_{\underline{a}} = \frac{\partial \underline{e}}{\partial \underline{a}^T} = -\ \mathbf{F}(\underline{\theta})\ . \tag{F-3}$$

The gradient vector of the least-squares functional can then be defined as

$$\nabla_{\lambda} v(\underline{\theta},\underline{a}) = 2\ \underline{e}^T\ \mathbf{J}_{\underline{\theta},\underline{a}}$$

$$= -\ 2\left[\ \underline{y} - \mathbf{F}\ \underline{a}\ \right]^T \left[\ \mathrm{D}(\mathbf{F})\ \underline{a}\ ,\ \mathbf{F}\ \right]\ . \tag{F-4}$$

If we now let

$$\underline{a}^* = F^+ \underline{y} \ , \tag{F-5}$$

then the gradient of the least-squares functional becomes

$$\begin{aligned} \nabla\chi(\underline{\theta} \ , \underline{a}^*) &= - 2 \left[\underline{y} - F \underline{a}^* \right]^T \left[D(F) \ \underline{a}^* \ , \ F \right] \\ &= - 2 \left[\underline{y} - F F^+ \underline{y} \right]^T \left[D(F) \ F^+ \underline{y} \ , \ F \right] \\ &= - 2 \ \underline{y}^T \ P_F^{\perp} \left[D(F) \ F^+ \underline{y} \ , \ F \right] \\ &= - 2 \left\{ \left[\underline{y}^T P_F^{\perp} \ D(F) \ F^+ \underline{y} \right] \cdot \left[\underline{y}^T P_F^{\perp} \ F \right] \right\} . \end{aligned} \tag{F-6}$$

But, since $$P_F^{\perp} \ F \ = \ 0 \ ,$$

the gradient of the least-squares functional becomes

$$\nabla\chi(\underline{a},\underline{\theta}) = - 2 \left\{ \left[\underline{y}^T P_F^{\perp} \ D(F) \ F^+ \underline{y} \right] \ , \ 0 \right\} . \tag{F-7}$$

Now recall the signal basis VPF,

$$\psi(\underline{\theta}) = \left|\left| \ P_F^{\perp} \ \underline{y} \ \right|\right|^2 .$$

The gradient of this functional is

$$\nabla\psi(\underline{\theta}) = \ 2 \ \underline{y}^T \ P_F^{\perp} \ D(F) \ F^+ \ \underline{y} \ .$$

If we assume that $\underline{\theta}^*$ is a critical point of $\psi(\underline{\theta})$, then

$$\nabla\psi(\underline{\theta}^*) = 2\ \underline{y}^T\ P_F^{\perp}\ D(F)\ F^+\ \underline{y} = 0\ .$$

Therefore, for $\underline{\theta}^*$ and $\underline{a}^*$ as defined above, the gradient of the least-squares functional becomes

$$\nabla\chi(\underline{\theta}^*,\underline{a}^*) = \left[\ 0\ ,\ 0\ \right]\ ,$$

which proves that a critical point of $\psi(\underline{\theta})$ is a critical point of $\chi(\underline{\theta},\underline{a})$ when $\underline{a}$ is defined as above. We will now prove that a global minimizer of $\psi(\underline{\theta})$ is also a global minimizer of $\chi(\underline{\theta},\underline{a})$.

For any given $\underline{\theta}$, the solution of the least-squares problem becomes a linear problem, which is straightforwardly minimized by letting $\underline{a}^* = F^+\ \underline{y}$. Therefore, for any given $\underline{\theta}$, $\psi(\underline{\theta}) \leq \chi(\underline{\theta},\underline{a})$. Now, if we assume that $\underline{\theta}^*$ is a global minimizer of $\psi(\underline{\theta})$ and $\underline{a}^*$ is defined as in Eq. (F-5), then certainly $\chi(\underline{\theta}^*,\underline{a}^*) = \psi(\underline{\theta}^*)$. Now assume that there exist $\tilde{\underline{\theta}}$ and $\tilde{\underline{a}}$ such that $\chi(\tilde{\underline{\theta}},\tilde{\underline{a}}) < \chi(\underline{\theta}^*,\underline{a}^*)$. Since $\psi(\tilde{\underline{\theta}}) \leq \chi(\tilde{\underline{\theta}},\tilde{\underline{a}})$, this requires that $\psi(\tilde{\underline{\theta}}) \leq \chi(\tilde{\underline{\theta}},\tilde{\underline{a}}) \leq \chi(\underline{\theta}^*,\underline{a}^*) = \psi(\underline{\theta}^*)$. But since $\underline{\theta}^*$ is the global minimizer of $\psi(\underline{\theta})$, we must have equality on all counts. Therefore, a global minimizer of $\psi(\underline{\theta})$ is also a global minimizer of $\chi(\underline{\theta},\underline{a})$ over $\underline{\theta}$. The converse [i.e., that a global minimizer of $\chi(\underline{\theta},\underline{a})$ over $\underline{\theta}$ is also a global minimizer of $\psi(\underline{\theta})$] is also proven by Golub and Pereyra [1], following a similar argument as above.

SUPPLEMENTARY

INFORMATION

CRDATA

DEPARTMENT OF THE NAVY
NAVAL RESEARCH LABORATORY
UNDERWATER SOUND REFERENCE DETACHMENT
ORLANDO, FL 32856-8337

IN REPLY REFER TO

Ser 5902.6-140-91
27 September 1991

From: Naval Research Laboratory, (Code 5902)
Orlando, FL 32856-8337

Subj: CHANGE TO NRL MEMORANDUM REPORT 6643

Ref: (a) NRL Memorandum Report 6643 of 27 Dec 1990

1. Delete reference 4 in reference (a).

2. Please call Irene Gonzalez (407-857-5131) or Lu'Anne Jevnager (407-857-5237), if you have any questions concerning this matter.

Terese Givens
TERESE E. GIVENS
Administrative Officer

Distribution:
USRD Standard Distribution List

AD A231 344