CONTINUATION APPROACHES FOR NONLINEAR LEAST SQUARES

by

Joshua Ivan Wold

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

Doctor of Philosophy

in

Engineering

MONTANA STATE UNIVERSITY
Bozeman, Montana

November, 2015

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Professor Steven Shaw for his excellent guidance. I benefited immeasurably from our many insightful conversations. Without his magnanimity in sharing knowledge and patience in the face of unique working conditions, this project would have been impossible.

I would also like to thank my committee members, Professors Robert Maher, Neda Nategh, and Dana Longcope.

Many thanks also to Professor Daniel Trudnowski for all the advice over the years, serving on my committee, and giving me the opportunity to get a head start on my career.

Finally, I would like to thanks my friends and family for their unwavering support. Especially my wife Ashley and our son Matthew - you are my inspiration.

TABLE OF CONTENTS

TABLE OF CONTENTS - CONTINUED

LIST OF FIGURES

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

ABSTRACT

Output error (OE) minimization is a commonly-used method for parameter estimation. It has the virtues that under reasonable conditions, the parameter estimates it produces are good (often maximum-likelihood) and only a simulation model is required to compute the predictions (making it equally useful for linear and nonlinear systems). Unfortunately, the procedure for computing the OE parameter estimates involves the minimization of a nonlinear cost function which requires the use of an iterative nonlinear least-squares (NLLS) algorithm such as Gauss-Newton. The present work begins with the observation that the one-step linear prediction error (PE) method - for which the parameter estimates can be computed via linear least-squares but suffers from noise-related bias - can be viewed as a particular transformation of the OE residuals. Further, that this transformation can be parameterized in such a way that a continuation of residual sequences is available that combines the properties of PE and OE. Minimizing a sequence of the cost functions associated with this continuation is shown to be a reliable method for finding the OE global minimum. It is later shown that the continuation need not begin with PE and that, in principle, any linear transformation of the OE residuals may be used provided that the resulting sequence of cost function global minima meet certain criteria. These criteria are provided in the form of a theorem which, in turn, makes use of the Implicit Function Theorem, an elementary result from analysis. The result is a method for guaranteeing that the continuation procedure has converged on the OE global minimum given a linear transformation has been chosen. No closed-form solution is given for the design of the transformation, but two designs are provided and shown to be successful on test problems.

1

INTRODUCTION

Mathematical modeling is an important part of engineering and physical science. In fact, it might be argued that modeling is *the* fundamental activity in science, if one is inclined to regard the laws of the physical sciences as "mere" models as opposed to objects of grander ontological status. In any case, mathematical models of physical systems are the workhorses of engineering. They are used for tasks ranging from control design and simulation to simply trying to better understand the system at hand. Properly performed, model building consists of both *a priori* physical reasoning, validation through experiment, and possibly revisions driven by data.

Mathematical modeling and data analysis, in the most general terms, involves combining measured data with a range of prior information in the form of model structures, initial guesses, statistical assumptions, and user choices. The objective of these analyses is to improve the starting knowledge by adjusting parameters, refining model choices, and so forth by comparison to actual measurements. In the specific context of dynamic system identification, prior information often consists of a system of differential equations drawn from physical laws (e.g., Newton's or Kirchhoff's laws), choice of a general model structure, initial guesses for model parameters, and information about expected noise involved with any measurements taken from the physical system. Data is often a time-series of measurements of one or more system inputs and outputs. This prior information and data can be combined using the process of system identification to produce a better model, which may consist of estimates for model parameters and/or a revised model structure.

In the present work, the focus is on the narrow problem of using system measurements to improve estimates of model parameters. It is assumed that an

adequate model structure (more on this below) has already been chosen, or that parameters estimated using the techniques developed here can be used in conjunction with model-order selection techniques to obtain a model structure. A common and conceptually simple technique for computing good parameter estimates is to generate a vector of residuals from the difference between the measurements and computer-generated model predictions and minimize the norm of this residual vector over the parameter estimates. This vector norm can be considered a function mapping the vector of parameter estimates to a scalar "fitness" value and is often called a cost function. The key choice that the user must make is how to form the residual vector, in particular, how to generate the predictions. As will be detailed in the following subsections, the choice has enormous consequences for both the statistical quality of the estimates obtained, and the difficulty of the process to obtain them.

Two common methods for forming the residual vector and cost function will be referred to as prediction error (PE) and output error (OE). In a PE method, the error is formed between a measurement and the prediction formed given prior measurements. In a OE method, the error is between the measurement and the output of a simulation model – the output errors at a particular sample are independent of other measured data points. In what follows, PE predictors for linear systems will be assumed to have the form of an inner product of a vector of model parameters and a vector containing the prior input and output measurements. Thus the PE parameters of a linear system can be estimated via linear least squares. This definition of PE is not to be confused with the much more general prediction error methods (PEM) common in the system identification literature (see, e.g., [28]) for which the predictions are the output of *any* model which may or may not be a function of prior measurements. OE for the same system is a nonlinear estimation problem. Intuitively, the fidelity of model needed to predict one step ahead (PE) is somewhat less than the fidelity

needed to achieve predictions of the data record from the initial guess forward (OE). The difficulty of the associated estimation problem reflects this.

The present research begins by exploring OE and PE and proposes a method for generating residuals that can be seen as an intermediate between OE and PE. This method is shown to combine the best features of PE (estimates are easy to obtain) and OE (estimates are good in a statistical sense). This method is then generalized considerably to the point that all residuals vectors (including PE and intermediates) are viewed as a linear transformation of the OE residuals. Once this general framework is established, the goal of the method is stated precisely and conditions for its success are provided.

Problem Introduction

As mentioned above, part of the prior information that a user brings to a modeling problem is the choice of a model structure. This choice has consequences for both the difficulty of the identification procedure and the quality of the model. For reasons discussed below, structures are very commonly chosen for which the parameter estimation problem requires minimization of a nonlinear cost function, which can be quite difficult. This situation is so common, in fact, that Ljung [29] has identified the difficulties associated with minimizing these nonlinear cost functions as one of the important open problems in system identification, describing solutions as "convexification" of the cost function.

One reason these nonlinear minimization problems arise so frequently is because in certain situations they are known to produce models with parameters close to the true underlying parameters of the physical system (assuming such parameters exist and the model accurately describes the behavior of the system). In the case of zero-

mean, white measurement noise, it has long been known that minimization of the OE cost function produces parameter estimates with good statistical properties [34, 43]. PEM are a class of techniques that are capable of producing estimates with good statistical properties under much more general noise conditions [2,26,30]. OE methods are a special case of PEM and under certain conditions minimizing the OE cost function and PEM have been shown to produce equally efficient estimates [19]. In the present work, the noise will always be assumed zero-mean white so that OE methods will produce estimates with desirable properties. Nonlinear cost functions are common to both OE and PEM. Thus, any progress made toward decreasing the difficulty of minimizing these cost functions should also apply to PEM.

OE models are also desirable for reasons other than the good statistical properties of the estimates. As mentioned above, the OE residual vector is generated as the difference between the measured system output and the output of a simulation model. Therefore, the user only needs to provide a simulation model of the system with initial guesses for the parameters, which are often readily available even for nonlinear systems. Many of the sophisticated techniques that are designed for the identification of linear systems do not work for nonlinear systems. The OE framework has no trouble dealing with nonlinear systems provided that they can be simulated.

Mathematical models of physical systems exist on a continuum between fully physical models for which the user supplies all of the equations and every parameter has a physical interpretation to black-box models where the model structure and possibly order have been chosen by the user, and for which the parameters have no physical interpretation. Black-box models are generally in some canonical form. For instance, state-space models in observable canonical form or state-space models with all entries in the system matrices variable can be viewed as black-box models. Models that fall somewhere between physical and black-box models are often called grey-

box or semi-physical models. Grey-box models consist of a combination of physical and non-physical equations and parameters. As an example, state-space models of mechanical systems may be derived from Newton's Laws and have system matrices that contain various physical parameters such as friction and inertia. Often, no attempt is made to create a physical model of the noise processes and they are simply modeled as (possibly filtered) statistical distributions whose parameters are estimated from the data but have no physical interpretation.

It is very often the case that a working scientist or engineer has a good dynamic model for their system, has performed experiments and would like to find the parameter values that best match their model to the experimental results. A conceptually appealing approach is to use an OE procedure to identify their grey-box model. As will be explored in detail later, identifying these grey-box models can be difficult. Nevertheless, OE methods are the natural choice for users who want both good statistical properties of the parameter estimates and physically parameterized models. The goal of the present work is to provide a reliable method for minimizing the OE cost function associated with grey-box models. The impetus for setting this goal is related to both the practical importance of the problem and its difficulty.

The following subsections provide an introduction to PE and OE to give a flavor of the tradeoff between model quality and difficulty associated with producing the model. For ease of presentation, the development assumes a discrete-time, linear, transfer function representation for the system. However, the observations made apply to other representations, specifically, the state-space models that will be used later in this document. This is basic material in the system identification literature and is described, for instance, in the popular textbooks [28, 44].

Output Error (OE)

For OE minimization, the user first generates predicted values for the system output using a simulation model and preliminary parameter values, then attempts to find the system parameters that produce simulated output values that "best fit" the measured output from the system. For the present purposes, "best fit" will be taken to mean minimizing the mean of squared errors between the simulated output and the measured output, and a justification for this choice will be offered later in this section. To begin, consider the linear, discrete-time model,

$$y_k + a_1 y_{k-1} + \cdots + a_n y_{k-n} = b_1 u_{k-1} + \cdots + b_n u_{k-n}, \tag{1.1}$$

where the sequences $\{u_k\}$ and $\{y_k\}$ represent measurements of the system input and output respectively, and the coefficients $\{a_k\}$ and $\{b_k\}$ are the parameters to be identified. For notational convenience, gather these parameters into the vector,

$$\theta = [a_1 \cdots a_n \;\; b_1 \cdots b_n]^T \tag{1.2}$$

and the input and output lags into the vector,

$$\psi_k^T = [-y_{k-1} \cdots - y_{k-n} \;\; u_{k-1} \cdots u_{k-n}]. \tag{1.3}$$

Define the transfer functions,

$$\begin{aligned} A(q,\theta) &= 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \\ B(q,\theta) &= b_1 q^{-1} + \cdots + b_n q^{-n} \end{aligned} \tag{1.4}$$

where q is the shift operator.

Define θ_0 as the actual value of θ for the underlying system. Then, (1.1)-(1.4) can be combined to write,

$$y_k = \frac{B(q,\theta_0)}{A(q,\theta_0)}u_k = \psi_k^T\theta_0. \tag{1.5}$$

The first equality in (1.5) emphasizes that the formulation is a general linear, discrete-time transfer function. Next, define the predictor,

$$\hat{y}_k(\theta) = \frac{B(q,\theta)}{A(q,\theta)}u_k, \tag{1.6}$$

where $\hat{y}_k(\theta)$ is the predicted value for y_k and the notation is intended to emphasize the dependence of the prediction on the variable parameter vector θ. Equation (1.6) can be rewritten,

$$\hat{y}_k(\theta) = [-\hat{y}_{k-1}(\theta)\cdots-\hat{y}_{k-n}(\theta)\;\; u_{k-1}\cdots u_{k-n}]\theta. \tag{1.7}$$

Define the output error,

$$e_k = y_k - \hat{y}_k(\theta). \tag{1.8}$$

As discussed above, the goal of OE minimization is to compute the values of the vector θ that minimize the sum of squared error between the predicted and measured outputs, which amounts to minimizing the 2-norm of the sequence e_k. Define the cost function,

$$V(\theta) = \frac{1}{2N}\sum_{k=1}^{N} e_k^2. \tag{1.9}$$

Clearly, the output error will be identically 0 (in this noise-free case) when the estimated parameter vector equals the actual parameter vector and, therefore, the cost function will be 0. That is, minimizing the cost function achieves the goal of

computing the correct parameter values. All that remains is to find a method to carry out the minimization of (1.9). Unfortunately, since the predictor for $\hat{y}_k$ depends on previous lags of $\hat{y}$ (see (1.7)) and these lags in turn depend on the parameter estimate θ, the predictor (and therefore the error sequence) is clearly nonlinear in the parameter vector θ. In general, the nonlinear relationship between the predictor and the parameter estimate requires an iterative method to solve for the value of θ that minimizes the cost function.

Despite this difficulty, OE minimization is very useful for two reasons. First, all that is required to produce the predicted values $\hat{y}_k$ is a simulation model (1.6), which is often readily available. Second, the statistical properties of the estimates produced are good (for a detailed description of parameter estimation in a statistical framework, see [28], Ch. 7). Consider the noisy measurement sequence,

$$\tilde{y}_k = y_k + \epsilon_k = \frac{B(q,\theta)}{A(q,\theta)} u_k + \epsilon_k. \tag{1.10}$$

where $\epsilon_k \sim N(0, \sigma^2)$, i.e. ϵ_k is zero-mean Gaussian noise. One common method to assess the quality of a parameter estimate is to define the likelihood function. It is defined as the joint probability density function (pdf) for all of the measurements,

$$L(\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_N|\theta) = f(\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_N|\theta) = f(\tilde{y}_1|\theta) f(\tilde{y}_2|\theta) \cdots f(\tilde{y}_N|\theta) \tag{1.11}$$

where L is the likelihood function, f is the pdf for the measurements, and where the second equality holds because of the assumed sample-to-sample independence of the noise. The difference between a likelihood function and a pdf is one of interpretation. For a pdf to be meaningful, it must be evaluated over a range (because the probability of any single point is 0). A likelihood function can be meaningfully evaluated at a

single point. Thus, it makes sense to seek the maximum of the likelihood function. Since the distribution of the noise was assumed to be known, the pdf for a given measurement is,

$$f(\tilde{y}_k|\theta) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}\left(\tilde{y}_k - \frac{B(q,\theta)}{A(q,\theta)}u_k\right)^2}, \tag{1.12}$$

which implies,

$$L(\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_N|\theta) = \frac{1}{(2\pi)^{N/2}\sigma^N} e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{N}\left(\tilde{y}_k - \frac{B(q,\theta)}{A(q,\theta)}u_k\right)^2}. \tag{1.13}$$

Clearly, (1.13) is maximized when the sum in the exponent is minimized. This sum can be rewritten using (1.6),

$$\sum_{k=1}^{N}\left(\tilde{y}_k - \frac{B(q,\theta)}{A(q,\theta)}u_k\right)^2 = \sum_{k=1}^{N}\left(\tilde{y}_k - \hat{y}_k(\theta)\right)^2 = (2N)V(\theta). \tag{1.14}$$

Thus, for the noise assumptions made here, the likelihood function is maximized when the least squares cost function is minimized, or, the maximum likelihood estimate (MLE) is identically equal to the least squares estimate for the parameter vector θ, a very desirable result.

The main difficulty of OE minimization is that the nonlinear relationship between the predictor and the parameter estimate requires an iterative technique to solve. Techniques such as the Gauss-Newton and Levenberg-Marquardt algorithms exist which are numerically simple, tractable, and converge to a local minimum of the cost function in most reasonable situations. Unfortunately, these techniques cannot be guaranteed to converge to the global minimum of the cost function. As will be seen in Ch. 2, even simple systems with linear dynamics can have cost functions with many closely spaced local minima in which the iterative algorithm can become trapped. Therefore, convergence to the global minimum depends heavily on the

quality of the initial guess that starts the algorithm. The initial guess is then left to the user and must consist of prior information produced by some means outside the system identification procedure.

Prediction Error (PE)

From the previous section, we know that OE minimization produces estimates with good statistical properties, but the algorithms required to compute the estimates are iterative and can become trapped in local minima. In this section, PE minimization is considered which, assuming the problem can be formulated properly, reduce to the solution of a *linear* least squares problem. However, the estimates produced are not MLE.

The difference between OE and PE is effectively in the way the predictions are formed. As mentioned in the last section, the sequences $\{u_k\}$ and $\{y_k\}$ are assumed to be available as measurements from the physical system; therefore, the vector ψ is known and the following predictor can be formed,

$$\hat{y}_k(\theta) = \psi_k^T \theta = [-y_{k-1} \cdots - y_{k-n} \;\; u_{k-1} \cdots u_{k-n}]\theta \tag{1.15}$$

Notice the contrast with OE. For OE, the current prediction ($\hat{y}_k$) is generated using the previous *predicted* values ($\hat{y}_{k-1}$, etc.), while for PE the current prediction is generated using the previous *measured* values (y_{k-1}, etc.). With the error sequence and cost function defined the same as in (1.8) and (1.9) respectively, it is again clear that the error sequence will be identically 0 when the estimated parameter vector equals the actual parameter vector and, therefore, the cost function will be 0. That is, minimizing the cost function achieves the goal of computing the correct parameter values. The only remaining task is to actually carry out the computation which can

be done easily by finding the least squares solution $\hat{\theta}$ to the matrix equation,

$$\begin{bmatrix} \hat{y}_k \\ \hat{y}_{k-1} \\ \vdots \\ \hat{y}_1 \end{bmatrix} + \begin{bmatrix} e_k \\ e_{k-1} \\ \vdots \\ e_1 \end{bmatrix} = \begin{bmatrix} \psi_k^T \\ \psi_{k-1}^T \\ \vdots \\ \psi_1^T \end{bmatrix} \hat{\theta} + \begin{bmatrix} e_k \\ e_{k-1} \\ \vdots \\ e_1 \end{bmatrix} = \begin{bmatrix} y_k \\ y_{k-1} \\ \vdots \\ y_1 \end{bmatrix} \tag{1.16}$$

$$\hat{Y} + E = \Psi\hat{\theta} + E = Y,$$

where e represents the noise and any modeling error and the capital letters are understood to be vectors/matrices making the notation more compact.

For the noise-free development above, the least-squares solution to (1.16) will be exact and $\hat{\theta}$ will be equal to θ, and, in this simplified situation, PE minimization produces the desired result. Unfortunately, PE methods are much less robust in the presence of noise than OE methods. To see this, first define the transfer function,

$$A_{n-1}(q,\theta) = a_1 q^{-1} + \cdots + a_n q^{-n}, \tag{1.17}$$

which is simply the denominator transfer function defined in (1.4) above less the q^0 term. If the noisy measurements are defined as,

$$\tilde{y}_k = y_k + \epsilon_k, \tag{1.18}$$

with the noise sequence again zero-mean Gaussian, the predictor (1.15) is now,

$$\hat{y}_k(\hat{\theta}) = [-\tilde{y}_{k-1} \cdots - \tilde{y}_{k-n} \;\; u_{k-1} \cdots u_{k-n}]\theta = \psi_k^T\theta + A_{n-1}(q,\theta)\epsilon_k \tag{1.19}$$

From the last equality in (1.19), it can be seen that the additive noise term is now filtered through the system dynamics. The noise at sample k is clearly not independent because the term $A_{n-1}(q,\theta)\epsilon_k$ depends on previous values of ϵ. Without independence, minimizing the sum of squared error does *not* coincide with the maximum likelihood estimate as it did for OE. That is, even though PE minimization finds the global minimum of the least squares cost function, the estimate is not guaranteed to be close to the underlying parameter value. In practice, the estimate will contain a noise-related bias. An example of this bias are shown in the next chapter. This is one main difficulty with PE. For linear systems, though, effective methods exist to work around this problem, including approximate pre-filtering of the data to "whiten" the noise in the equation error formulation.

Another difficulty with PE is that it requires an explicit vector inner product formulation for the predictor (1.15). This predictor was easy to construct for the linear system above parameterized by the coefficients of the transfer function. It is more difficult to construct for linear systems parameterized differently, and may be impossible for some nonlinear systems. In practice, most users who wish to identify model parameters have a simulation model. Therefore, it would be better if the method only required a simulation model rather than a predictor whose relationship to the model may not be obvious.

The main advantage of PE minimization is that, assuming a predictor can be formulated, the solution that minimizes the L^2 cost function can be found with relative ease by solving (1.16) in a least-squares sense.

Summary

Prediction error minimization produces parameter estimates with less desirable statistical distributions and require that an explicit predictor be formed which is not, in general, a simple task. However, provided the predictor can be formed, the actual computation of the estimate is not difficult and the estimate is guaranteed to be the global minimum of the prediction error cost function. Output error methods produce parameter estimates with excellent statistical distributions and require nothing more than a simulation model and nonlinear solver to produce the estimate. However, the solution produced is not guaranteed to be the global minimum of the cost function and, for systems with many local minima, may actually be quite far from the global minimum depending on the quality of the initial guess.

Related Methods

System identification is a remarkably broad field. Practitioners with quite diverse backgrounds have made contributions across a wide variety of journals. Any attempt an overview of the field is destined to be incomplete. This literature review will attempt to describe some of the methods most closely associated with the problem at hand; to reliably minimize the OE cost function associated with grey-box models.

Many methods start with a PE, linear predictor-type model (often called equation error in the literature), modify it in some way to solve the bias problem, and still solve a linear problem. For instance, in one application of instrumental variable methods (IVM) [42,49] the least squares normal equations are modified using a vector of "instruments" that are uncorrelated with the noise. Write the least-squares normal equations as above in (1.16), then the standard solution method is to premultiply by

Ψ^T and invert,

$$\hat{\theta} = (\Psi^T\Psi)^{-1}\Psi^T Y. \tag{1.20}$$

The matrix Ψ contains measurements Y, and if Y contains any measurement noise, then the solution $\hat{\theta}$ will depend on the noise (resulting in bias). Other solutions to (1.16) are possible however. Instead of pre-multiplying by Ψ^T, one IVM approach is to premultiply by some other matrix Φ^T composed of "instrumental variables". Then (1.20) becomes,

$$\hat{\theta} = (\Phi^T\Psi)^{-1}\Phi^T Y. \tag{1.21}$$

As long as $\Phi^T\Psi$ is invertible, (1.21) will result in a solution. If the instrumental variables are also uncorrelated with the noise, then Φ^T multiplied by the noise portion of Y will be 0 (on average) and the solution will not depend on the noise (resulting in 0 bias). This is just one simple example of IVM. The predictions need not be produced in the linear predictor manner as above. In general, IVM encompasses a large number of techniques for which neither the predictions nor the instruments are specified in advance.

Other methods can produce good parameter estimates but require that the model take on some specified structure. With respect to a physical system, these models must be considered black-box. In [39] Regalia constrains the model in such a way that the noise adds a fixed term to the cost function (and therefore does not affect the location of the minimum). Specifically, for a model in transfer function form, it is shown that the noise contribution to the cost function is a function of the norm of the denominator coefficient vector only. If the model is constrained in such a way that the denominator norm is fixed (and equal to 1), then the noise adds a constant term a constant term and does not affect the location of the minimum (leading to zero bias).

The related online methods in [1,22,25] use various techniques to trade low bias in OE minimization for global convergence in PE minimization. All of the techniques add a "meta-parameter" that effectively controls the tradeoff by weighting the innovation for the parameter update with more OE or more PE.

The Stieglitz-Mcbride algorithms [48] apply a pre-filter (that depends on the current parameter guess) to the PE residuals, solves the linear problem, updates the pre-filter with the new parameter estimate, and repeats. These algorithms have been shown in [48] to converge to the correct parameter estimates when the system order is known and the measurement noise is white. These algorithms are analyzed in detail in Chapter 8 of [38].

A family of techniques called subspace identification methods [56–59] have been developed which are able to identify black-box state space models (for which all of the entries to the matrices are assumed unknown) using a non-iterative procedure involving matrix decompositions. Conceptually, these methods can be thought of as using the extended observability matrix (estimated from the I/O measurements) [28] to estimate the sequence of states. Once the sequence of states has been estimated, a linear regression problem can be written to estimate the system matrices.

All of the above methods provide some way of both reliably computing the parameter estimates and dealing with noise-related bias in these estimates. Some of them can be shown to produce estimates that coincide with the MLE estimate. However, none of these methods provide a means of finding the global minimum of the OE (or PEM) cost function in a general setting.

If the user wishes to find the global minimum of the OE cost function, it is often suggested that he use the parameter estimates obtained from one of the above methods as an initial guess for the iterative OE minimization procedure. There is no guarantee, of course, that the initial guess is inside the convergence region of

the global minimum. It should also be noted that using one of the methods that requires a specific model structure for finding an initial guess is not very useful for the problem of minimizing the OE cost function associated with grey-box models because such methods dictate the parameterization. That is, these methods are designed for particular parameterizations and any initial guesses they produce will not, in general, be initial guesses for the parameters of the grey-box models. To use these initial guesses would require that they first be transformed (e.g., via a similarity transformation for state-space systems) to the chosen parameterization in the grey-box model.

Practically, grey-box model users seeking good (MLE) parameter estimates are left with few options but to try and minimize the OE (or PEM) cost function despite its nonlinearity. This is why, despite the vast number of techniques available for producing "good" models, OE and PEM remain very popular.

The most common method for dealing with the nonlinearity is to simply improve the initial guess. If the initial guess is inside the basin of attraction of the global minimum, then standard NLLS solvers can be used to find the global minimum. Producing initial guesses can be as simple as monitoring the minimization process and restarting it with different initial guesses should it fail or converge to a local minimum. As the number of parameters grows, so does the dimension of the search space and it becomes unlikely that a user could find an initial guess within the basin of attraction of the global minimum by chance.

As mentioned in the previous section, some of the black-box techniques can be used to produce initial guesses for grey-box models provided that a similarity transformation from the black-box to the grey-box parameterization can be found. In [31] this is described for certain state-space structures. A related approach appears in [40].

Another method for improving initial guesses depends on further physical insight about the system under test. Certain systems admit specialized tests that isolate parameters and make the estimation of those parameters much easier. For instance, in power transformer modeling, short-circuit and open-circuit tests can be performed and the estimation of some of the transformer parameters are reduced to simple calculations. In [51], the authors present an extensive review of techniques for the important problem of estimating induction motor parameters. The results of the many experiments described therein can provide good initial guesses for OE minimization algorithms.

Instead of attempting to improve the initial guess for a local search method, the user could also attempt to use a global search method. Global search methods are generally heuristic in nature, are not provably convergent, and can be complicated and difficult to apply, especially when the number of parameters is large. For these reasons, global search methods have not gained much traction in the system identification literature (for example, they are not even mentioned in the popular textbook [28]). Nonetheless, such methods exist and have been applied to system identification problems. Some of the more common global search methods (also called non-convex optimization methods) are genetic algorithms [5], particle swarm optimization [8], simulated annealing [55], Tabu search [12], and ant colony optimization [7]. Efforts to apply these methods to system identification can be found in [17, 18, 20, 24, 36, 50].

Many global optimization methods are most effective when the dimensionality is so high that there are many acceptable solutions and the global optimum is not significantly better than these many other local optima. There exists a class of optimization problems which have a high enough number of dimensions that brute force global search is computationally prohibitive, but for which there still exists

a single acceptable global optimum (see [10], Ch. 13 for a lucid discussion of this important point). It is often the case that nonlinear parameter estimation falls into this class of problems.

In the present work, another method for finding the global minimum of the OE cost function is explored which relies on the idea that it is possible to produce a general cost function for which the OE and PE cost functions are special cases and which is continuously deformable. These notions will be made precise and clearly explained in the following chapters.

Ch. 2 explores the specific problem of sinusoid frequency estimation and uses it as a vehicle for introducing some of the novel insights in the present research. Ch. 3 forms the theoretical heart of this research and presents the general framework and a method for finding the global minimum of the cost function. Ch. 4 covers some of the practical, computation aspects of applying the method. Ch. 5 revisits the sinusoid frequency estimation problem and applies the insights of Ch. 3 and 4. Ch. 6 provides several examples of applying the method to some more complicated problems. Finally, Ch. 7 offers concluding remarks and directions for future research.

19

SINUSOID FREQUENCY ESTIMATION

Estimating the frequency of a sinusoidal signal provides a good entry point for understanding the difficulty associated with minimizing OE cost functions. Sinusoids are conceptually simple and their analysis proves more analytically tractable than more complicated systems while still exhibiting the complex behavior of the OE cost function.

In addition to being useful for theoretical purposes, sinusoid frequency estimation is also of considerable practical interest. Applications vary from ADC characterization [11, 61] to diverse applications for power systems. Frequency measurements are used directly for equipment protection [9, 14]. The dynamic response of the power system frequency to disturbances is a critical indicator of grid stability. Therefore, wide-area frequency measurements [62] are important for monitoring and control. Real-time frequency measurements are planned to be used for a feedback controller that will improve the stability of the western grid [52, 53].

Due to the theoretical and practical importance of sinusoid frequency estimation, it is a very well-studied problem. Many excellent methods exist that produce accurate measurements. Here the goal will be to provide a small but broad sample of existing work that reflects the range of available techniques. It is useful to separate these techniques into groups according to their methods for dealing with the bias/estimate accuracy and cost function complexity tradeoff.

First, online algorithms [3, 16, 32, 37] constantly update the frequency estimate as new data points arrive. These algorithms can generally prove asymptotic global convergence (regardless of the accuracy of the initial guess). That is, they will always converge to the right answer as the number of data points approaches infinity. For

finite-length data sequences, the properties of the estimates are often difficult to characterize.

Second, offline or batch methods act on some finite-length sequence of measurements all at once and either directly solve a linear problem [21] or iteratively minimize a nonlinear cost function to compute the estimate. The methods which solve a linear problem usually include some type of regularization to deal with bias problems [54]. Although they can be effective, the linear methods are generally considered to be less accurate than nonlinear methods. The methods that minimize a nonlinear cost function usually include a procedure for computing a pre-estimate of the frequency [13, 46] which improves the likelihood that the iterative procedure will converge to the global minimum. The statistical properties of the estimates obtained by the nonlinear methods are well-understand, and under certain conditions, optimal in the sense that they achieve the Cramer-Rao lower bound (CRB) [47]. However, convergence is not always guaranteed.

The approach taken here is a batch method that uses a kind of hybrid of the linear and nonlinear methods described above. This method offers reliable convergence to the global minimum of the OE cost function. The estimate thus obtained corresponds to the maximum likelihood estimate in the presence of white Gaussian noise. For more general systems and with colored noise, OE estimates are not as accurate as PEM estimates [28], which include a model of the noise process [43]. However, it has been shown that both PEM and OEM asymptotically approach the CRB even in the presence of colored noise for the sinusoid parameter estimation problem [47]. Therefore, there is clearly value in being able to reliably minimize the OE cost function.

This chapter begins by deriving a useful new result for the spacing of the minima of the OE cost function. Next, the OE and PE problems are recast in a unified

framework. Finally, a new method is applied that provides reliable convergence to the OE global minimum that makes use of this unified framework. This method is generalized in subsequent chapters.

The OE Cost Function and Its Minima

This section shows that the least-squares OE cost function for sinusoid frequency estimation has many local minima and therefore presents a significant challenge for iterative minimization algorithms such as Gauss-Newton. A previously unknown formula for the location of the local minima is derived. One use for this formula is to produce bounds on the accuracy of the initial guess required for global convergence.

Consider a signal described by

$$y_k = \alpha e^{jk\phi_0} \tag{2.1}$$

$$= \alpha \left(\cos(\phi_0 k) + j \sin(\phi_0 k)\right). \tag{2.2}$$

where α is a complex number including phase and magnitude information, and $k = 1 \ldots N$. With noise, which we temporarily omit other than by framing the problem in terms of least squares, (2.2) could represent a sequence of measurements of a continuous-time sinusoidal signal. The amplitude and phase parameter α in (2.2) appears *linearly*, and thus contributes quadratic terms to the cost function that do not increase the difficulty of the associated least-squares estimation problem. Therefore, for the purpose of studying the more complicated dependence of the problem on ϕ_0, we rewrite (2.2) as

$$y_k = e^{jk\phi_0} = a^k \tag{2.3}$$

and introduce an estimate for this signal

$$\hat{y}_k = e^{jk(\phi_0+\delta)} = a^k e^{jk\delta}, \tag{2.4}$$

where δ is the difference between the estimated frequency and the true value of ϕ_0. The OE cost function associated for the signal procedure seeks to minimize the cost function,

$$V = \frac{1}{2N}\sum_{k=1}^{N}(\hat{y}_k - y_k)(\hat{y}_k - y_k)^* \tag{2.5}$$

where $(\cdot)^*$ is the complex conjugate and N is the total number of points. Manipulation of (2.5) yields,

$$\begin{aligned} V &= \frac{1}{2N}\sum_{k=1}^{N}(\hat{a}^k - a^k)(\hat{a}^k - a^k)^* \\ &= \frac{1}{2N}\sum_{k=1}^{N}[e^{j\phi_0 k}(e^{j\delta k} - 1)][e^{-j\phi_0 k}(e^{-j\delta k} - 1)] \\ &= 1 - \frac{1}{2N}\sum_{k=1}^{N} e^{j\delta k} + e^{-j\delta k}. \end{aligned} \tag{2.6}$$

Note that if δ is zero, V is also zero, and if δ is not zero, the real parts of both of the terms inside the sum are less than 1. Thus, familiar formulas can be applied to eliminate the sums. Recall [45],

$$\sum_{k=1}^{N} e^{k\beta} = \frac{e^{\beta} - e^{(N+1)\beta}}{1 - e^{\beta}}. \tag{2.7}$$

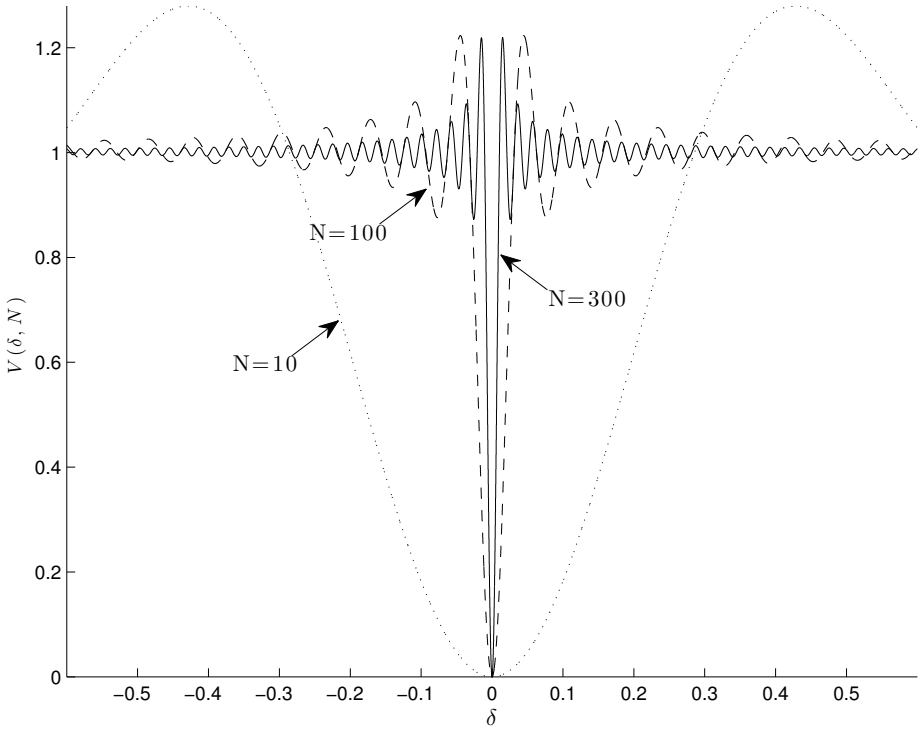

Figure 2.1: Cost function as a function of frequency error for several values of N from (2.8).

Applying (2.7) to both of the terms in the sum in (2.6) and explicitly writing the arguments yields,

$$\begin{aligned} V(\delta, N) &= 1 - \frac{1}{2N}\left[\frac{e^{j\delta} - e^{j(N+1)\delta}}{1 - e^{j\delta}} + \frac{e^{-j\delta} - e^{-j(N+1)\delta}}{1 - e^{-j\delta}}\right] \\ &= 1 + \frac{1}{2N}\left[1 + \frac{\cos((N+1)\delta) - \cos(N\delta)}{1 - \cos(\delta)}\right]. \end{aligned} \tag{2.8}$$

Equation (2.8) is plotted for several values of N in Fig. 2.1. The cost function has many local minima and for the NLLS algorithm to converge to the correct frequency, the initial guess must be between the two peaks nearest to 0. Since the cost function is symmetric about 0, the positive δ where the first peak occurs is also the maximum tolerable error. To find the location of the first peak, begin by computing the partial

derivative of V with respect to δ,

$$\frac{\partial V(\delta, N)}{\partial \delta} = \frac{(N+1)\sin(N\delta) - N\sin((N+1)\delta)}{4N\sin^2\left(\frac{\delta}{2}\right)} \tag{2.9}$$

which follows from the quotient rule and repeated application of trigonometric sum and difference formulas. Now, set (2.9) equal to 0 and rearrange,

$$\frac{N+1}{N}\sin\left(N\delta\right) = \sin\left((N+1)\delta\right). \tag{2.10}$$

Unfortunately, (2.10) does not have a closed form solution for δ in terms of N. However, the equation simplifies if an expansion of the following form is considered,

$$\delta(\epsilon) = c_0\epsilon + c_1\epsilon^2 + c_2\epsilon^3 + ... \tag{2.11}$$

where $\epsilon = \frac{1}{N}$ and c_k are coefficients to be determined using (2.10). Note that for large N, ϵ becomes small, and (2.11) looks like a traditional Taylor series. To compute the coefficients for the series, substitute (2.11) into (2.10) and replace N with $\frac{1}{\epsilon}$,

$$(1+\epsilon)\sin(c_0 + c_1\epsilon + c_2\epsilon^2 + ...) = \sin(c_0 + (c_0+c_1)\epsilon + (c_1+c_2)\epsilon^2 + ...). \tag{2.12}$$

Now, apply trigonometric sum formulas to separate the c_0 term from rest of the series, leaving a series of only small terms,

$$\begin{aligned}(1+\epsilon)&[\sin(c_0)\cos(c_1\epsilon + c_2\epsilon^2 + ...) + \cos(c_0)\sin(c_1\epsilon + c_2\epsilon^2 + ...)] \\ &= \sin(c_0)\cos((c_0+c_1)\epsilon + (c_1+c_2)\epsilon^2 + ...) \\ &\quad + \cos(c_0)\sin((c_0+c_1)\epsilon + (c_1+c_2)\epsilon^2 + ...).\end{aligned} \tag{2.13}$$

Now the functions containing only small terms can themselves be expanded with the familiar Maclaurin series for sin and cos. After dividing through by $\cos(c_0)$ and expanding, (2.13) becomes,

$$\begin{aligned}
&(1+\epsilon)\Bigg[\tan(c_0)\left(1-\frac{(c_1\epsilon+c_2\epsilon^2+...)^2}{2!}+...\right)\\
&\quad+\left((c_1\epsilon+c_2\epsilon^2+...)-\frac{(c_1\epsilon+c_2\epsilon^2+...)^3}{3!}+...\right)\Bigg]\\
&=\tan(c_0)\left(1-\frac{(c_0+c_1)\epsilon+(c_1+c_2)\epsilon^2+...)^2}{2!}+...\right)\\
&\quad+\Bigg(\big((c_0+c_1)\epsilon+(c_1+c_2)\epsilon^2+...\big)\\
&\quad-\frac{\big((c_0+c_1)\epsilon+(c_1+c_2)\epsilon^2+...\big)^3}{3!}+...\Bigg).
\end{aligned}\tag{2.14}$$

To compute the coefficients c_k, equate like powers of ϵ and solve. The result is,

$$c_k=\frac{c_0(-1)^k}{k+1},\qquad k=1,2,...\tag{2.15}$$

where c_0 is a solution to,

$$c_0=\tan(c_0).\tag{2.16}$$

Equation (2.16) is plotted in Fig. 2.2. There are clearly many solutions, and each solution corresponds to a point in δ where the derivative in (2.9) is zero (i.e. a local min or max of the cost function). For instance, if the smallest positive solution ($c_0\approx 4.4934$) is used, $\delta(N)$ will be the value for which the cost function has its first peak. The next smallest solution to (2.16) corresponds to the second local min or max of the cost function, etc. The expression for $\delta(N)$ is now,

$$\delta(N)=c_0\left(\frac{1}{N}-\frac{1}{2N^2}+\frac{1}{3N^3}-...\right).\tag{2.17}$$

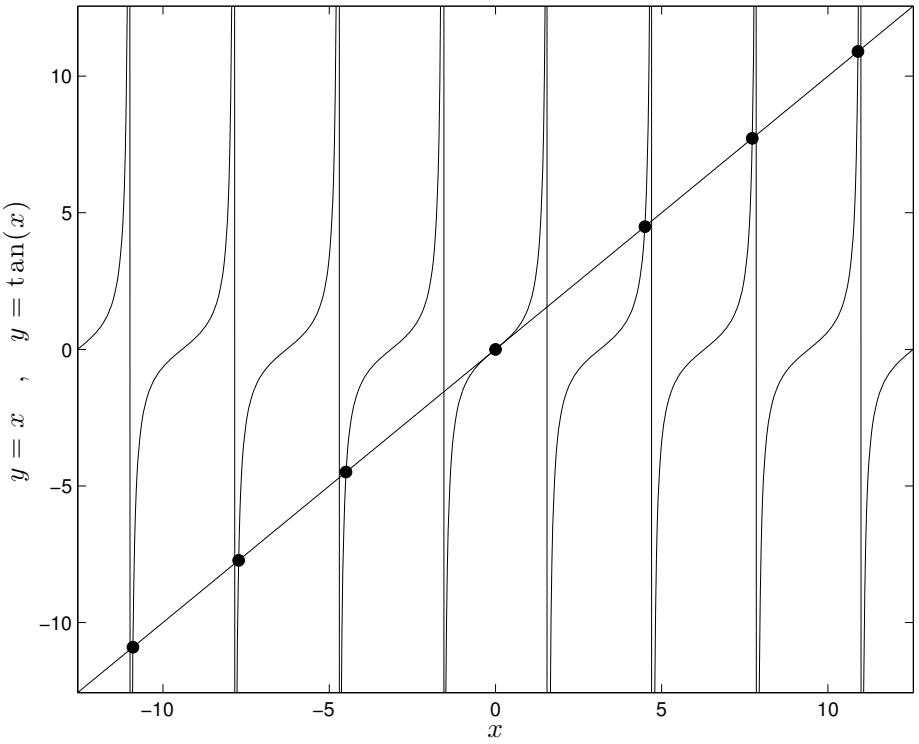

Figure 2.2: $y = \tan(x)$ and $y = x$ with solutions to (2.16) indicated by dots which correspond to local maxima and minima of V.

Since the series has alternating signs, $\delta(N)$ is guaranteed to be larger that the first two terms of the series, so taking the maximum tolerable error to be $\frac{4.4934(2N-1)}{2N^2}$ will guarantee convergence for the NLLS algorithm. The sum in (2.17) also has a closed form for $N > 1$,

$$\delta(N) = c_0 \ln\left(\frac{N+1}{N}\right). \tag{2.18}$$

Fig. 2.3 is a log-log plot of the absolute error between $\delta(N)$ as defined in (2.18) and the value of δ at the first peak of the loss function as computed by Matlab's *fzero* function. It shows that the sequence defined in (2.17) converges to a solution to (2.10) for large N. The complicated fine structure for $N > 1000$ is due to numerical

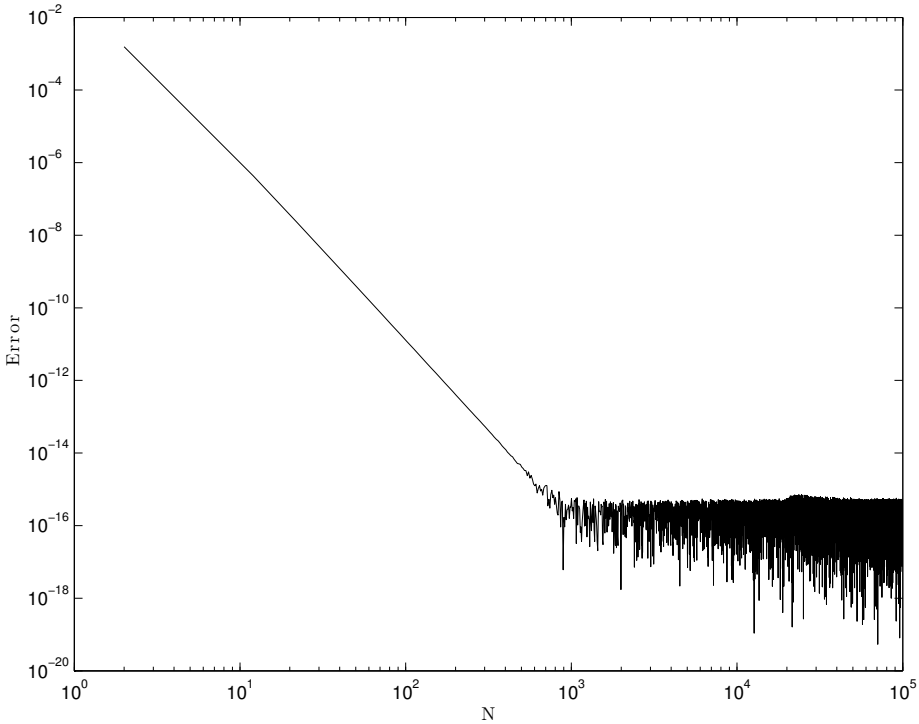

Figure 2.3: Log-log plot of the absolute error between $\delta(N)$ as defined in (2.18) and the value of δ at the first peak of the loss function as computed by Matlab's *fzero* function. Fine structure at the right is due to numerical error because the subtraction operation produces values near machine ϵ.

error because the subtraction operation produces values near machine ϵ. If greater precision were used, the downward trend would continue further.

OE, PE and Beyond for Sinusoid Frequency Estimation

In the last section the OE cost function for sinusoid frequency estimation was discussed and characterized. In the absence of an accurate initial guess (the precise required accuracy is given by (2.18)), the OE cost function will be difficult to minimize. In this section, a method for reliably obtaining the OE global minimum will be presented. First, the sinusoidal frequency estimation problem must be reformulated. In the previous section, the sinusoids were written as explicit functions

of the frequency variable ϕ and the index variable k in (2.3) and (2.4). In this section, the signals will be written as the outputs of state-space systems which will allow us to analyze the problem with a wider variety of techniques than in the previous section. The OE problem will be set up first, followed by the PE problem and a hybrid technique that allows convergence to the OE global minimum without the need for an accurate initial guess.

To begin, consider the state-space representation of a sampled sinusoid with unity amplitude and phase zero,

$$x_{k+1} = A_0 x_k = \begin{bmatrix} \cos(\phi_0) & \sin(\phi_0) \\ -\sin(\phi_0) & \cos(\phi_0) \end{bmatrix} x_k \quad , \quad x_0 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{2.19}$$

$$y_k = C x_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k, \tag{2.20}$$

where ϕ_0 is the frequency of the sinusoid. Note that the initial state x_0 fixes the magnitude and phase of the signal. In general, the initial state will be unknown and will need to be estimated. However, as in the previous section, the magnitude and phase only contribute linear terms to the estimation problem which do not increase the difficulty. Therefore, they will again be assumed to be fixed and known for clarity of presentation.

Output Error

As described in the previous section, OE minimization seeks to minimize the cost function,

$$V(\phi) = \frac{1}{2N} \sum_{k=1}^{N} (y_k - \hat{y}_k(\phi))^2 \tag{2.21}$$

where ϕ is understood to be the variable over which the minimization is carried out and where,

$$\begin{aligned} \hat{x}_{k+1} = A\hat{x}_k = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix} \hat{x}_k \quad , \quad \hat{x}_0 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \hat{y}_k(\phi) = C\hat{x}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \hat{x}_k. \end{aligned} \tag{2.22}$$

In (2.21) the complex conjugate has been omitted because the signals y_k and $\hat{y}_k$ are assumed to be real. Notice that, as described in Ch. 1, the only information required for producing the sequence $\hat{y}_k$ is a simulation model and a guess for the parameter ϕ. However, there is no obvious procedure for carrying out the minimization of (2.21). Iterative methods are generally required to carry out the minimization of the cost function and these methods cannot distinguish between local and global minima. Fig. 2.1 in the previous section plots several representative examples of (2.21) and clearly shows that an accurate initial guess will be required for convergence to the global minimum.

Prediction Error

PE minimization seeks to minimize the same cost function as OE (2.21), but the sequence $\hat{y}_k$ is defined differently. For PE, $\hat{y}_k$ needs to be written as a function of previous measured values y_{k-1}, y_{k-2}, etc. Begin by rewriting (2.20) as,

$$x_{k+1} = A_0^k x_0. \tag{2.23}$$

The characteristic equation for the matrix A_0 is,

$$\lambda^2 + 1 = 2\lambda\cos(\phi_0). \tag{2.24}$$

The Cayley-Hamilton [15] theorem implies that the matrix A_0 must itself satisfy (2.24),

$$A_0^2 + I = 2A_0\cos(\phi_0), \tag{2.25}$$

where I is the 2×2 identity matrix. Equation (2.25) implies,

$$CA_0^kA_0^2x_0 + CA_0^kx_0 = 2\cos(\phi_0)CA_0^kA_0x_0 \tag{2.26}$$

Substituting (2.20) and (2.23) into (2.26) gives,

$$y_{k+2} + y_k = 2\cos(\phi_0)y_{k+1}. \tag{2.27}$$

Shifting indices and rearranging (2.27) gives a method to write the prediction sequence $\hat{y}_k(\phi)$ as a function of the previous measurements of the system output y. Explicitly,

$$\hat{y}_k(\phi) = 2\cos(\phi)y_{k-1} - y_{k-2}. \tag{2.28}$$

The advantage of the PE approach is that the predictor (2.28) (and, therefore, the cost function) is a much simpler function of the parameter ϕ. In fact, if ϕ were not embedded in the cosine function in (2.28), the predictor would be linear in ϕ and the cost function would be quadratic. Said another way, if the change of variables, $a = \cos(\phi)$, is made, the cost function can be minimized uniquely by solving a linear regression problem. In practice, this change of variables is not necessary because the

cosine function is well-behaved and if the initial guess is within $\pm\pi$ of ϕ_0, minimizing the cost function directly via an iterative method should not be difficult.

A Continuation from PE to OE

In the previous two subsections, the OE and PE problems were formulated for sinusoid frequency estimation. However, the discussion in the Introduction chapter implies that neither minimization procedure will likely yield useful estimates for ϕ_0; the PE estimate will suffer from bias in the presence of measurement noise and the OE cost function will be difficult to minimize globally without a very accurate initial guess. These two issues are illustrated clearly in Fig. 2.4. The OE cost function is unbiased but difficult to minimize and the PE cost function is easy to minimize but biased.

This subsection introduces (via the sinusoid example) the core insight of this dissertation. Namely, that both the OE and PE problems can be viewed as special cases of a more general framework and that cost functions sharing some of the properties of both PE and OE are available in this framework. By solving a sequence of these intermediate problems, the estimate that globally minimizes the OE cost function can be reliably obtained. Consider the modified estimator $\hat{y}_k$ defined by the following state-space system,

$$\begin{aligned} \hat{x}_{k+1} &= A\hat{x}_k + L(y_k - \hat{y}_k) \\ \hat{y}_k &= C\hat{x}_k \end{aligned} \tag{2.29}$$

with A and C defined as in (2.22). When $L = [0\ 0]^T$, clearly this system has the same result as the output error equations (2.22). With other choices of L, the modified estimator (2.29) incorporates measurements y_k, which modify the internal

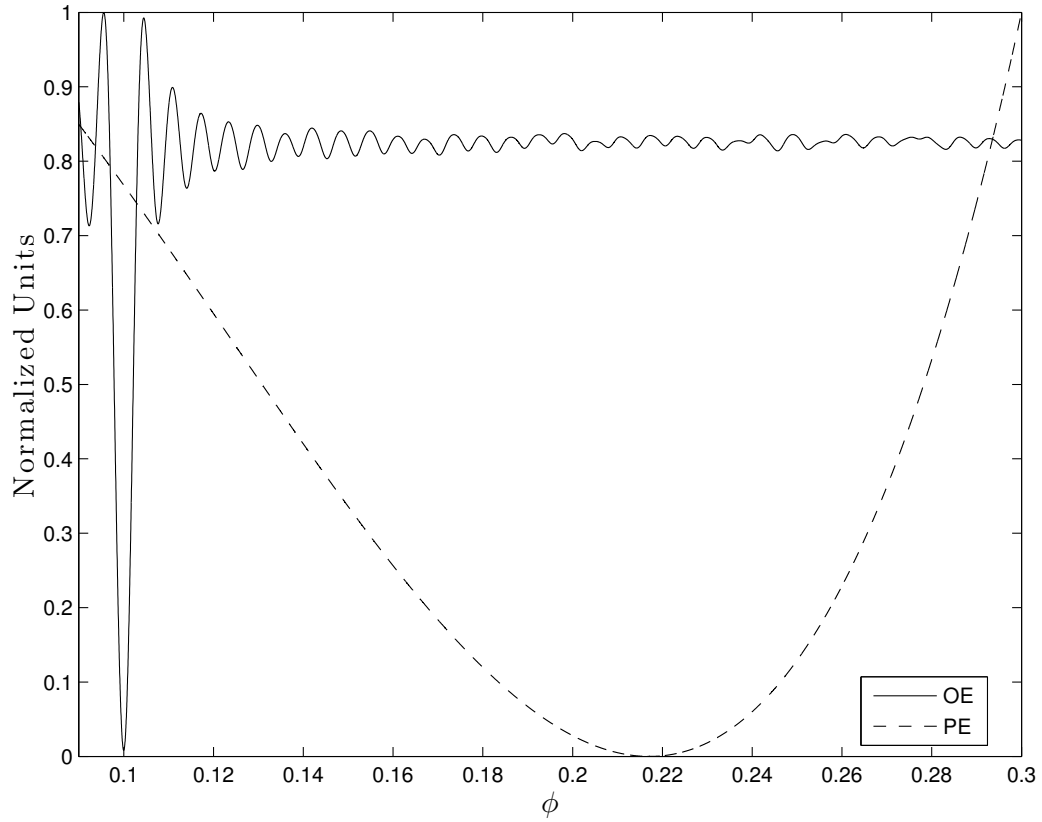

Figure 2.4: OE and PE cost functions against estimated frequency ϕ for $\phi_0 = 0.1$ with a modest amount of white measurement noise. The cost functions have been shifted vertically and scaled so that both are clearly visible.

state estimate $\hat{x}$. To examine this, consider the error dynamics,

$$x_{k+1} - \hat{x}_{k+1} = (A + \Delta)x_k - A\hat{x}_k - LC(x_k - \hat{x}_k), \tag{2.30}$$

where $A_0 = A + \Delta$ is a matrix containing the estimated underlying parameters of the system plus the error due to ϕ not being equal to ϕ_0. If we define a state estimate error $e_k = x_k - \hat{x}_k$, the system can be written as

$$e_{k+1} = (A - LC)e_k + \Delta x_k. \tag{2.31}$$

Thus, the error dynamics are determined by $A-LC$ driven by an input corresponding to the dynamics of the true underlying system and the parameter error Δ.

A feedback gain L_0 which eliminates e_k in two steps can be determined by setting $(A - L_0C)(A - L_0C) = 0$ (resulting in a deadbeat system with all eigenvalues at 0). It is

$$L_0 = \begin{bmatrix} 2\cos(\phi) \\ \frac{\cos^2(\phi)-\sin^2(\phi)}{\sin(\phi)} \end{bmatrix}. \tag{2.32}$$

With this feedback gain, the errors are driven entirely by the measurement sequence.

To examine this, write the following equations

$$\hat{x}_{k-1} = (A - LC)\hat{x}_{k-2} + Ly_{k-2} \tag{2.33}$$

$$\hat{x}_k = (A - LC)\hat{x}_{k-1} + Ly_{k-1} \tag{2.34}$$

$$\hat{y}_k = C\hat{x}_k. \tag{2.35}$$

Eliminating $\hat{x}_k$ and $\hat{x}_{k-1}$ yields

$$\hat{y}_k = C\left[(A - LC)((A - LC)\hat{x}_{k-2} + Ly_{k-2}) + Ly_{k-1}\right]. \tag{2.36}$$

Since we've found an L_0 that ensures $(A-L_0C)(A-L_0C) = 0$, this equation becomes

$$\hat{y}_k = C\left[(A - L_0C)L_0y_{k-2} + L_0y_{k-1}\right] = 2\cos(\phi)y_{k-1} - y_{k-2} \tag{2.37}$$

which is explicitly in the form of the prediction error. Moreover, this equation is identical to the prediction error formed previously (2.28). Therefore, with the modified estimator formulation, we can choose $L = 0$ to obtain OE and choose $L = L_0$ as above to obtain PE. It seems plausible, then, that if we choose $L = \alpha L_0$ for some

constant α from 0 to 1, the predictor obtained will have some combination of the properties of OE and PE. Fig. 2.5 shows the cost function for several values of α. Note that the cost functions for values of α between 1 and 0 achieve intermediate behavior between OE and PE. Fig. 2.5 suggests that it may be possible to minimize a sequence

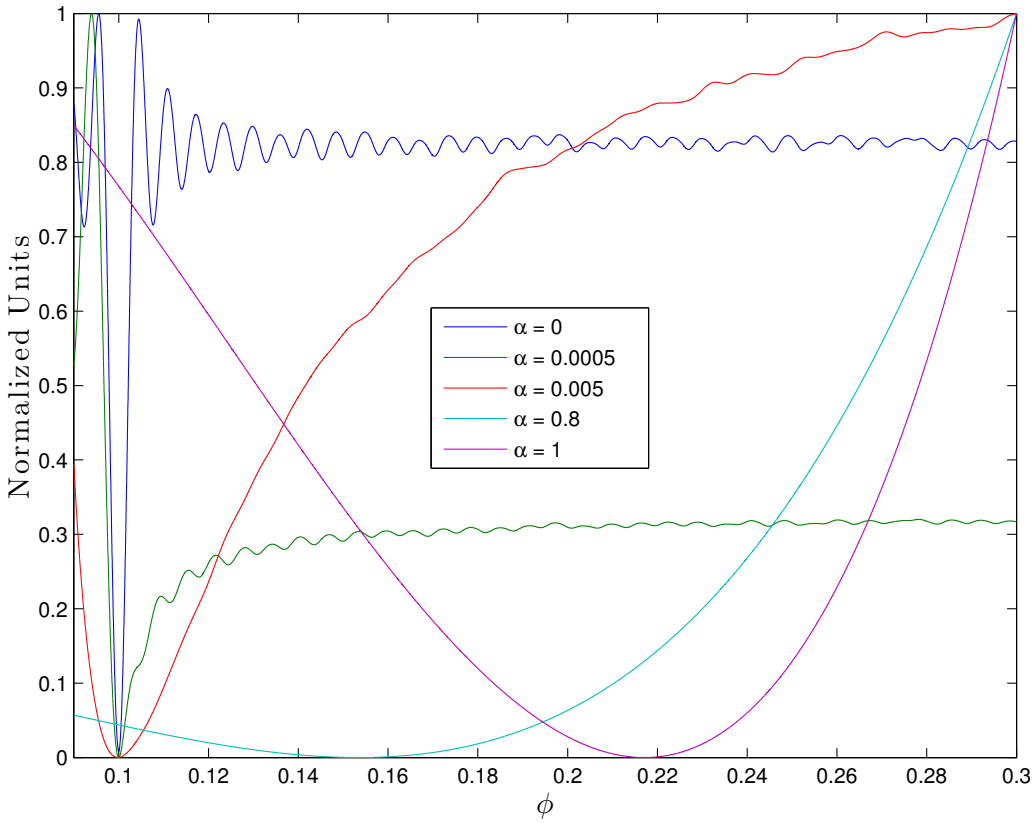

Figure 2.5: Cost functions against estimated frequency ϕ several values of α and $\phi_0 = 0.1$. The cost functions have been shifted vertically and scaled so that all are clearly visible.

of cost functions globally and eventually converge to the OE global minimum. It is useful to think of the cost function as a function of both ϕ and α, $V(\phi, \alpha)$. Let $\bar{\phi}(\alpha)$ be the value of ϕ that globally minimizes of cost function. With this new notation, the goal is to find $\bar{\phi}(\alpha_{oe}) = \bar{\phi}(0)$ and a procedure for doing so is roughly the following: 1) calculate $\bar{\phi}(\alpha_{pe})$ (which is easy because the PE cost function is easy to minimize), 2)

calculate $\bar{\phi}(\alpha_1)$ where $\alpha_1 = \alpha_{pe} + \delta$ and δ is some small step using $\bar{\phi}(\alpha_{pe})$ as the initial guess, 3) continue stepping along in α, each step calculating $\bar{\phi}(\alpha_j)$ using $\bar{\phi}(\alpha_{j-1})$ as the initial guess until $\alpha_j = \alpha_{oe}$.

The success of this procedure rests on the assumption that the the user can also make a step in α for which last global minimum in the sequence is within the convergence region of the global minimum of the next cost function. It seems reasonable that this should be possible. Fig. 2.6 shows the sequence of global minima as a function of α for the frequency estimation problem. The procedure outlined above is simply to carefully traverse this curve from $\alpha = 1$ to $\alpha = 0$.

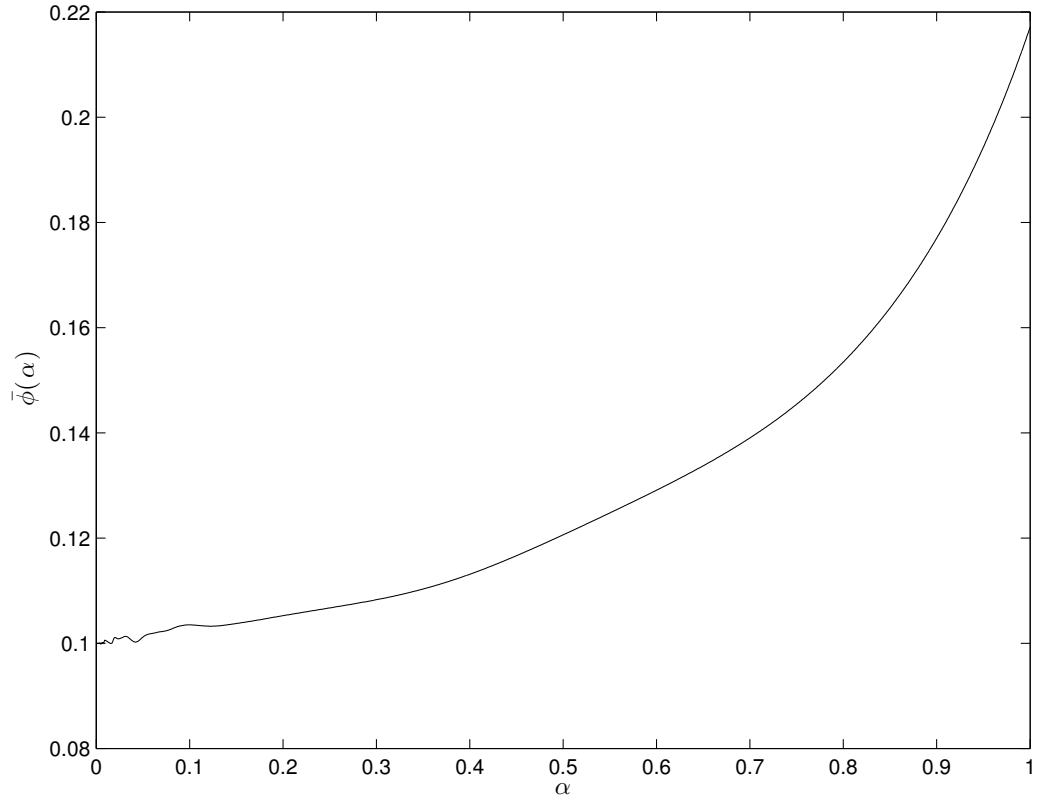

Figure 2.6: Cost function minimum, $\bar{\phi}(\alpha)$, against α for $\phi_0 = 0.1$.

Conclusions

This chapter has analyzed the problem of sine wave frequency estimation. A new result for the spacing of the local minima was derived. OE and PE minimization were presented in a novel, general framework for which intermediate cost functions are also available and it was suggested that the OE global minimum could be obtained by solving a sequence of minimization problems, starting at PE and ending at OE.

This technique for minimizing the OE cost function seems reasonable but also brings up many questions. Is there any way to prove that the technique will always find the OE global minimum? Does the method generalize to other state-space systems? While the OE and PE ends of the continuation seem fixed, the method for moving between them (slowly reducing the gain of the observer L) seemed somewhat arbitrary. Are there other continuations that might also achieve the same performance?

These and other questions will be taken up in the ensuing chapters.

GENERALIZATION

In Ch. 2, a method was developed for producing a sequence of cost functions between PE and OE. It was argued that minimizing a sequence of these cost functions should not be difficult and that the end result would be the global minimum of the OE cost function. This method was designed specifically in the context of sinusoid frequency estimation and it was noted that the heuristic for moving from PE to OE was somewhat arbitrary.

In this chapter the goal is to broaden the presentation. First, the continuation method will be viewed as a modification of the cost function via a general linear transformation of the OE residuals. From this viewpoint, both the arbitrary nature of the continuation and possible alternatives are more clearly exposed. Second, a more rigorous exploration of the properties of the continuation is undertaken and sufficient conditions for success of the method are proposed.

A General View of the Cost Function

Begin by introducing some new notation. Collect the N observations into the vector

$$Y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \end{bmatrix}. \tag{3.1}$$

This capital letter notation will be used in general for vectors. For instance,

$$\hat{Y}(\theta) = \begin{bmatrix} \hat{y}_0(\theta) \\ \hat{y}_1(\theta) \\ \vdots \\ \hat{y}_{N-1}(\theta) \end{bmatrix} \tag{3.2}$$

is the vector of simulated outputs. Consider a SISO state space system,

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned} \tag{3.3}$$

for $k = 0 \dots N-1$. Using the vector notation above and carrying out the operations in (3.3), the vector output of the system can be written,

$$Y = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} x_0 + \begin{bmatrix} D & 0 & 0 & 0 & & 0 \\ CB & D & 0 & 0 & \cdots & 0 \\ CAB & CB & D & 0 & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ CA^{N-3}B & \cdots & CAB & CB & D & 0 \\ CA^{N-2}B & \cdots & CA^2B & CAB & CB & D \end{bmatrix} U \tag{3.4}$$

Recall from Ch. 1 that the general cost function was defined as,

$$V(\theta) = \frac{1}{2N} \sum_{k=0}^{N-1} (y_k - \hat{y}_k(\theta))^2, \tag{3.5}$$

where, as in Ch. 2, $\hat{y}_k(\theta)$ is defined as the output of the state-space system,

$$\hat{x}_{k+1} = A\hat{x}_k + L(y_k - \hat{y}_k) = (A - LC)\hat{x}_k + Ly_k \tag{3.6}$$

$$\hat{y}_k(\theta) = C\hat{x}_k. \tag{3.7}$$

Note in (3.6) that the observations y_k can be viewed as inputs to the estimator $\hat{y}_k(\theta)$. Following (3.4), write,

$$\begin{aligned}\hat{Y}(\theta) &= \underbrace{\begin{bmatrix} C \\ C(A-LC) \\ \vdots \\ C(A-LC)^{N-1} \end{bmatrix}}_{G} x_0 + \underbrace{\begin{bmatrix} 0 & 0 & 0 & & 0 \\ CL & 0 & 0 & \cdots & 0 \\ C(A-LC)L & CL & 0 & & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ C(A-LC)^{N-2}L & \cdots & C(A-LC)L & CL & 0 \end{bmatrix}}_{M} Y \\ &= Gx_0 + MY \end{aligned} \tag{3.8}$$

Next, recall that the OE estimator is simply a simulation of the system outputs without making use of the observations. Explicitly, the vector $\hat{Y}_{OE}(\theta)$ can be written in the same form as (3.8) but with $L = 0$,

$$\hat{Y}_{OE}(\theta) = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} x_0 \tag{3.9}$$

Define the vector sequence of residuals by,

$$\begin{aligned} R(\theta) &= Y - \hat{Y}(\theta) \\ &= I_N Y - G x_0 - MY \\ &= (I_N - M)Y - G x_0 \end{aligned} \tag{3.10}$$

where I_N is the $N \times N$ identity matrix. Similarly, define the OE residuals as $R_{OE}(\theta) = Y - \hat{Y}_{OE}(\theta)$. Some fairly tedious matrix multiplication and algebra can establish the identity,

$$G = (I_N - M) \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} x_0 = (I_N - M)\hat{Y}_{OE}(\theta) \tag{3.11}$$

Substituting (3.11) into (3.10) yields,

$$\begin{aligned} R(\theta) &= (I_N - M)Y - (I_N - M)\hat{Y}_{OE}(\theta) \\ &= (I_N - M)(Y - \hat{Y}_{OE}(\theta)) \\ &= F(\theta) R_{OE}(\theta) \end{aligned} \tag{3.12}$$

where the matrix $F(\theta) = (I_N - M)$ and the dependence on θ is written explicitly as a reminder. Finally, using the vector notation, the cost function can be written as,

$$V(\theta) = \frac{1}{2N} R^T(\theta) R(\theta) = \frac{1}{2N} R_{OE}^T(\theta) F^T(\theta) F(\theta) R_{OE}(\theta) \tag{3.13}$$

where $(\cdot)^T$ denotes the vector transpose. The last two expressions offer two new (equivalent) interpretations for the way that the continuation method modifies the cost function. Equation (3.12) says that the method can be interpreted as producing

a new residual sequence by passing the OE residuals through a linear transformation (which can usefully be thought of as a filter). Equation (3.13) says that the method can be interpreted as producing a new cost function by applying a weighting matrix to the OE cost function.

It is worth writing out the matrix $F(\theta)$ explicitly for the state estimator formulation that was used for the sinusoid frequency estimation problem,

$$F(\theta) = I - M = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ -CL & 1 & 0 & \cdots & 0 \\ -C(A-LC)L & -CL & 1 & & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ -C(A-LC)^{N-2}L & \cdots & -C(A-LC)L & -CL & 1 \end{bmatrix} \tag{3.14}$$

As mentioned above, the state estimator formulation for the continuation was arbitrary to some degree. Equation (3.12) offers a new perspective on the full range of choices that are available. Any linear transformation of a vector can be written as a matrix multiplication. Therefore, (3.12) shows that the state estimator formulation is just one possible linear transformation of the OE residual vector and that *any* method that attempts to modify the OE cost function by some linear procedure can be written in the form of (3.13). Said another way, any linear method can be thought of as simply designing the matrix $F(\theta)$.

From this new perspective, it is clear that there are many more degrees of freedom available to the designer. In principle, every entry in $F(\theta)$ could be designed separately. Examining (3.14) makes clear exactly what structure the state estimator method imposes on $F(\theta)$ and these structural impositions are related to standard properties of the state estimator (3.7). Namely, $F(\theta)$ is lower triangular because the

estimator is causal, $F(\theta)$ has diagonals with equal elements because the estimator is time-invariant, and, as mentioned above, the reason that the residual sequence can be written in this matrix multiplication form at all is because the estimator is linear.

At this point it should be clear that many other methods for designing $F(\theta)$ might exist with very different effects on the cost function. In light of this newly discovered freedom, it is important to re-establish the goal of the method. Ideally, the method would lead to guaranteed global minimization of the OE cost function. The obvious question, then, is what are the design criteria for $F(\theta)$ that will lead to this guaranteed global minimization.

The remainder of this chapter is devoted to providing sufficient conditions for guaranteed convergence to the OE global minimum.

Conditions for Guaranteed Convergence to OE Global Minimum

Loosely speaking, the argument for guaranteed convergence to the OE global minimum proceeds as follows: first, the OE cost function is modified via (3.13) into some form that is more easily globally minimized (e.g. by transforming to a PE cost function); second, this global minimum is found via some numerical procedure; third, the matrix $F(\theta)$ is slowly modified in a way that the next global minimum is always near the previous global minimum; last, the final cost function to be minimized is the OE cost function (i.e. when $F(\theta)$ is the identity matrix).

Later in this section the above argument will be made more rigorous, but some preliminary ideas must be established first.

The Gauss-Newton Algorithm for NLLS

The Gauss-Newton algorithm [6] is a method for solving NLLS problems that is based on Newton's method for finding roots of functions. Generally speaking, Newton's iterative method, which finds zeros of functions, is applied to the derivative of the least squares cost function to find a (local) minimum.

Begin by differentiating the cost function with respect to θ,

$$\frac{\partial V(\theta)}{\partial \theta} = \begin{bmatrix} \frac{\partial V(\theta)}{\partial \theta_1} \\ \vdots \\ \frac{V(\theta)}{\partial \theta_n} \end{bmatrix} \tag{3.15}$$

and differentiating again,

$$\frac{\partial^2 V(\theta)}{\partial \theta \partial \theta^T} = \begin{bmatrix} \frac{\partial^2 V(\theta)}{\partial \theta_1^2} & \frac{\partial^2 V(\theta)}{\partial \theta_1 \theta_2} & \cdots & \frac{\partial^2 V(\theta)}{\partial \theta_1 \theta_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 V(\theta)}{\partial \theta_n \theta_1} & \frac{\partial^2 V(\theta)}{\partial \theta_n \theta_2} & \cdots & \frac{\partial^2 V(\theta)}{\partial \theta_n^2} \end{bmatrix} \tag{3.16}$$

The Gauss-Newton algorithm begins by approximating the cost function locally about the current guess $\theta^{(i)}$ (where i denotes the Gauss-Newton iteration number) by its second-order Taylor series,

$$V(\theta) \approx V(\theta^{(i)}) + \frac{\partial V^T(\theta^{(i)})}{\partial \theta}(\theta - \theta^{(i)}) + \frac{1}{2}(\theta - \theta^{(i)})^T \frac{\partial^2 V(\theta^{(i)})}{\partial \theta \partial \theta^T}(\theta^{(i)})(\theta - \theta^{(i)}) \tag{3.17}$$

where the derivative notation is understood to mean $\frac{\partial V^T(\theta^{(i)})}{\partial \theta} = \left.\frac{\partial V^T(\theta)}{\partial \theta}\right|_{\theta=\theta^{(i)}}$ and all higher order derivatives are assumed to be 0 (i.e. the cost function is assumed to be locally quadratic). Equation (3.17) is then minimized directly by differentiating with

respect to θ and setting equal to 0,

$$\frac{\partial V^T(\theta^{(i)})}{\partial \theta} + \frac{\partial^2 V(\theta^{(i)})}{\partial \theta \partial \theta^T}(\theta - \theta^{(i)}) = 0 \tag{3.18}$$

The so-called normal equations (3.18) are solved for the updated parameter estimate $\theta^{(i+1)}$, via,

$$\theta^{(i+1)} = \theta^{(i)} - \left(\frac{\partial^2 V(\theta^{(i)})}{\partial \theta \partial \theta^T}\right)^{-1} \frac{\partial V^T(\theta^{(i)})}{\partial \theta} \tag{3.19}$$

Equations (3.17)-(3.19) could apply to any function $V(\theta)$. The Gauss-Newton algorithm takes advantage of the special structure of the least squares cost function to compute the derivative quantities in (3.19). Define the Jacobian of the residual vector as,

$$J(\theta) = \frac{\partial R(\theta)}{\partial \theta^T} = \begin{bmatrix} \frac{\partial r_1(\theta)}{\partial \theta_1} & \cdots & \frac{\partial r_1(\theta)}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_N(\theta)}{\partial \theta_1} & \cdots & \frac{\partial r_N(\theta)}{\partial \theta_n} \end{bmatrix} \tag{3.20}$$

Then with the cost function defined as in the last section as $V(\theta) = \frac{1}{2N} R^T(\theta) R(\theta)$, the derivative in (3.15) can be written as,

$$\frac{\partial V(\theta)}{\partial \theta} = \frac{1}{N} J^T(\theta) R(\theta). \tag{3.21}$$

The Gauss-Newton algorithm sets the second order derivatives of $R(\theta)$ with respect to θ to 0 (i.e., $\frac{\partial J(\theta)}{\partial \theta} = 0$) in calculating the derivative (3.16). This assumption is valid near the local minimum but can cause problems far from the minimum. With this assumption, (3.16) can be calculated by differentiating (3.21) using the product rule,

$$\frac{\partial^2 V(\theta)}{\partial \theta \partial \theta^T} = \frac{1}{N} J^T(\theta) J(\theta). \tag{3.22}$$

Finally, substituting (3.21) and (3.22) into (3.18), the normal equations become,

$$J^T(\theta)R(\theta) + J^T(\theta)J(\theta)\left(\theta^{(i+1)} - \theta^{(i)}\right) = 0 \tag{3.23}$$

and the explicit solution for the parameter update is,

$$\theta^{(i+1)} = \theta^{(i)} - \left(J^T(\theta)J(\theta)\right)^{-1} J^T(\theta)R(\theta). \tag{3.24}$$

Clearly, the algorithm ceases to change the parameter estimate when $\theta^{(i+1)} = \theta^{(i)}$. This implies that all stationary points (minima) must satisfy,

$$J^T(\theta)R(\theta) = 0. \tag{3.25}$$

Equation (3.25) is the primary result that will be used in the argument that follows. It is important that most practical NLLS solvers use a technique more sophisticated. For instance, Matlab's *lsqnonlin* function has two built-in options - the trust-region reflective and Levenberg-Marquardt algorithms [4, 33]. These differ from Gauss-Newton primarily in that they have some method for restricting the step size. These algorithms are designed to be more robust than Gauss-Newton far from the local minima. However, near the minima, all of the algorithms perform similarly and, most importantly, they all converge on stationary points that satisfy (3.25). As will be discussed in detail later, (3.25) can be viewed as an implicit definition for the global minimum of the cost function.

The Implicit Function Theorem

Another important piece of the continuity argument is the Implicit Function Theorem (IFT). The IFT is an elementary result from analysis that will be stated

without proof and explored by way of a simple example. Many calculus texts contain rigorous proofs, e.g. [35]. A form of the theorem (adapted from [35]) is stated formally below.

Formal Statement of Implicit Function Theorem Let G be an open set in $\mathbb{R}^{n+1}$ containing the point $(\bar{\alpha}, \bar{\theta})$ where $\bar{\alpha}$ is a scalar and $\bar{\theta}$ is a $n \times 1$ vector. Suppose some continuously differentiable function $F(\alpha, \theta)$ exists that maps $\mathbb{R}^{n+1}$ to $\mathbb{R}^n$ with $F(\bar{\alpha}, \bar{\theta}) = 0$ and that the matrix of first partial derivatives of F with respect to θ is invertible on G. Then there exists a continuously differentiable function f and a closed region H inside G such that $F(\alpha, f(\alpha)) = 0$ for $(\alpha, f(\alpha))$ on H. Further, if D is the matrix of first partial derivatives of F with respect to θ,

$$D = \begin{bmatrix} \frac{\partial F_1}{\partial \theta_1} & \cdots & \frac{\partial F_1}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial \theta_1} & \cdots & \frac{\partial F_n}{\partial \theta_n} \end{bmatrix} \tag{3.26}$$

then,

$$\frac{\partial f}{\partial \alpha} = -D^{-1} \frac{\partial F}{\partial \alpha}. \tag{3.27}$$

In words, the theorem states the conditions under which a continuously differentiable function F of two variables, α and θ, implicitly defines θ as a continuously differentiable function of α - namely that the matrix D must be invertible. Finally, the theorem gives an explicit formula to calculate the partial derivative of the function f with respect to α. Note that the theorem makes a local statement about the existence of f, not a global one. As we will see in the example below, there need not be a single f over all possible points (α, θ) that satisfy $F(\alpha, \theta) = 0$, and there may be some points where no continuously differentiable f exists at all.

IFT Example The canonical example to illustrate the IFT is the equation for a circle, which can also be analyzed directly using algebra. Consider the unit circle,

$$F(x, y) = x^2 + y^2 - 1 = 0 \tag{3.28}$$

Solving for y algebraically gives $y = \pm\sqrt{1 - x^2}$. Thus, there is no function $y = f(x)$ that defines y globally. Locally, however, there are two functions, $f_1(x) = \sqrt{1 - x^2}$ and $f_2(x) = -\sqrt{1 - x^2}$ that define y as a function of x over the top and bottom halves of the circle respectively. These functions are continuously differentiable except when $y = 0$, where the derivative does not exist (the tangent is vertical at these points).

Application of the IFT should confirm the above algebraic observations. First, the continuity conditions of $F(x, y)$ and its partial derivatives are clearly met. The final condition for the existence of a continuously differentiable function $f(x)$ that satisfies $F(x, f(x)) = 0$ is that the matrix D in (3.26) be invertible. For the unit circle,

$$D = \frac{\partial F(x, y)}{\partial y} = 2y. \tag{3.29}$$

Clearly, D is invertible except when $y = 0$; therefore, the IFT implies that the unit circle locally defines y as a function of x everywhere except when $y = 0$. Further, (3.27) implies that,

$$\frac{\partial f(x)}{\partial x} = -D^{-1}\frac{\partial F}{\partial x} = -\frac{1}{2y}2x = -\frac{x}{y} \tag{3.30}$$

This result can be compared to differentiating either of the functions above directly. For example,

$$\frac{df_1(x)}{dx} = -\frac{x}{\sqrt{1 - x^2}} = -\frac{x}{f_1(x)} = -\frac{x}{y} \tag{3.31}$$

which agrees with (3.30). Thus, all of the algebraic observations are confirmed by the IFT. The power of the IFT is that the existence of the implicitly defined function $f(x)$

can be established even when it is not possible to solve for it directly. The existence of this function is the key feature for the continuation argument that follows.

The Continuation Argument

The goal of this section is to prove that if certain conditions are met, then the global minimum of the OE cost function can be found by solving a series of minimization problems, each of which are guaranteed to converge. To make this argument precise, some new notation is required. As in the sinusoid example in Ch. 2, the continuation will be parameterized by the variable α. In the sinusoid example, a specific formulation for how the continuation depends on α was given (αL_0 where L_0 is defined in (2.32) and α varies from 1 to 0). For the present argument, no specific dependence on α will be defined, except to define α_{oe} as the value that produces the OE cost function and α_0 as the value that produces a modified cost function for which the global minimum is "easily" obtainable. For the sinusoid example, the cost function for α_0 is the PE cost function described in detail in Ch. 2.

Next, the dependence of the cost function on α will be made explicit, $V(\alpha, \theta)$. The important equations from the beginning of this chapter will be repeated here with α shown explicitly. The residuals are,

$$R(\alpha, \theta) = F(\alpha, \theta) R_{OE}(\theta) \tag{3.32}$$

the Jacobian is,

$$J(\alpha, \theta) = \frac{\partial R(\alpha, \theta)}{\partial \theta^T} = \begin{bmatrix} \frac{\partial r_1(\alpha,\theta)}{\partial \theta_1} & \cdots & \frac{\partial r_1(\alpha,\theta)}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_N(\alpha,\theta)}{\partial \theta_1} & \cdots & \frac{\partial r_N(\alpha,\theta)}{\partial \theta_n} \end{bmatrix} \tag{3.33}$$

and the cost function is,

$$V(\alpha,\theta) = \frac{1}{2N} R_{OE}^T(\theta) F^T(\alpha,\theta) F(\alpha,\theta) R_{OE}(\theta). \tag{3.34}$$

The point at the start of this chapter that modification of the cost function amounts to designing the matrix $F(\alpha,\theta)$ is now explicit in (3.34). The OE residuals $R_{OE}(\theta)$ are unaffected by the continuation parameter, so the only effect α has on the cost function is via $F(\alpha,\theta)$. Note that $F(\alpha_{oe},\theta) = I_N$ since then $V(\alpha_{oe},\theta) = \frac{1}{2N} R_{OE}^T(\theta) R_{OE}(\theta)$ is the OE cost function.

Finally, let $\bar{\theta}$ be the global minimum of the cost function. The continuation argument can now be stated. In brief, it proceeds as follows. The global minimum must be a fixed point of the cost function and, therefore, must satisfy,

$$J^T(\alpha,\bar{\theta}) R(\alpha,\bar{\theta}) = 0. \tag{3.35}$$

If (3.35) satisfies the conditions of the IFT, then it can be thought of as an implicit definition for the global minimum as a function of α, which will be denoted $\bar{\theta}(\alpha)$, and the IFT guarantees that this function will be continuous in α. This continuity implies that a non-zero step size in α can always be chosen so that the global minimum $\bar{\theta}(\alpha)$ moves an arbitrarily small amount. As long as a neighborhood around the next global minimum exists for which the Gauss-Newton (or similar) algorithm will converge, a step size can always be found for which the previous global minimum is within this neighborhood (because the continuity of $\bar{\theta}(\alpha)$ means that the successive minima can be made arbitrarily close to each other). Assuming the global minimum has been found for some initial value α_0, a sequence of minimizations can be carried out (using

the previous minimum as the initial guess for the next minimization) each of which is guaranteed to converge and which terminates at the OE global minimum.

This argument is now stated formally.

Theorem 3.0.1. *Let α_0 and $\bar{\theta}_0$ exist and be known such that $J^T(\alpha_0, \bar{\theta}_0)R(\alpha_0, \bar{\theta}_0) = 0$ and $V(\alpha_0, \bar{\theta}_0) \leq V(\alpha_0, \bar{\theta})$ for all $\bar{\theta}$ with $R(\alpha, \bar{\theta})$, $J^T(\alpha, \bar{\theta})$, and $V(\alpha, \bar{\theta})$ defined in (3.32), (3.33), and (3.34), respectively. If there is a continuous variable α between fixed values α_0 and α_{oe} such that $J^T(\alpha, \bar{\theta})R(\alpha, \bar{\theta}) = 0$ satisfies the conditions of the IFT at each point, then $\bar{\theta}$ is defined implicitly as a function of α and sequences $\alpha_0, \alpha_1, \cdots, \alpha_{oe}$ and $\bar{\theta}(\alpha_0), \bar{\theta}(\alpha_1), \cdots, \bar{\theta}(\alpha_{oe})$ exists for which the Gauss-Newton algorithm is guaranteed to converge to $\bar{\theta}(\alpha_{k+1})$ starting from the initial point $\bar{\theta}(\alpha_k)$. Thus, a sequence of minimizations can be carried out which ends in calculating $\bar{\theta}(\alpha_{oe})$, the global minimum of the OE cost function.*

Proof. Since $J^T(\alpha_0, \bar{\theta}_0)R(\alpha_0, \bar{\theta}_0) = 0$ satisfies the conditions of the IFT, the function $\bar{\theta}(\alpha)$ is defined and continuous in the neighborhood of $(\alpha_0, \bar{\theta}_0)$. Continuity implies that for every $\epsilon > 0$ there exists $\delta > 0$ such that $||\bar{\theta}(\alpha_0+\delta)-\bar{\theta}(\alpha_0)|| < \epsilon$. Dennis has shown [6] (Theorem 10.2.1) that a neighborhood of convergence exists for the Gauss-Newton algorithm. The conditions of this existence are implied by the conditions of the IFT. Therefore, there exists some ϵ_k such that Gauss-Newton is guaranteed to converge to $\bar{\theta}(\alpha_{k+1})$ starting from the initial point $\bar{\theta}(\alpha_k)$ as long as $||\bar{\theta}(\alpha_{k+1}) - \bar{\theta}(\alpha_k)|| < \epsilon_k$. Choose $\alpha_1 = \alpha_0 + \delta_0$ such that $||\bar{\theta}(\alpha_1) - \bar{\theta}(\alpha_0)|| < \epsilon_0$. Then the Gauss-Newton algorithm is guaranteed to converge to $\bar{\theta}(\alpha_1)$ starting from $\bar{\theta}(\alpha_0)$. Since $\bar{\theta}_0 = \bar{\theta}(\alpha_0)$ is assumed to be known, $\bar{\theta}(\alpha_1)$ can be found. This procedure can be repeated for $\bar{\theta}(\alpha_2) \cdots \bar{\theta}(\alpha_{oe})$ with convergence guaranteed at each step. Thus, $\bar{\theta}(\alpha_{oe})$ can be found with certainty. □

Theorem 3.0.1 states a set of sufficient conditions which, if satisfied, guarantee that the OE global minimum can be found by solving a sequence of Gauss-Newton minimizations. These conditions are worth laying out again for clarity:

1. There must be some α_0 for which the *global* minimum of the cost function $V(\alpha_0, \theta)$ is known. This implies that some transformation of the OE cost function has been made such that the transformed cost function is convex, or nearly so. Transforming to the PE cost function has been shown to achieve this for the sinusoid problem, and other potential transformations will be explored in ensuing chapters.

2. The function defining the stationary point $J^T(\alpha, \bar{\theta})R(\alpha, \bar{\theta}) = 0$ must satisfy the conditions of the IFT at every point along the continuation from α_0 to α_{oe}. Specifically, the stationary point must be continuously differentiable and the matrix of partial derivatives with respect to θ must be invertible. As will be detailed later, $J^T(\alpha, \bar{\theta})R(\alpha, \bar{\theta})$ and its derivatives are complicated functions of θ and it is very unlikely that any global statements can be made about them. Rather, the best that can be achieved is to numerically check the conditions of the IFT at each point along the continuation. Obviously, the conditions cannot be checked before the sequence of minimizations is carried out (because the values of $\bar{\theta}$ that satisfy $J^T(\alpha, \bar{\theta})R(\alpha, \bar{\theta}) = 0$ are not known in advance - if they were, the problem would be solved). Therefore, in practice, the user will not have a guarantee in advance that their procedure will end in global minimization of the OE cost function. They will, however, have the means to diagnose problems during the procedure and, at the end, determine with confidence whether the minimum has been found.

3. The continuation must end at the OE cost function (i.e. $V(\alpha_{oe}, \theta) = \frac{1}{2N} R_{OE}^T(\theta) R_{OE}(\theta)$). Using the linear transformation idea from the beginning of this chapter, the matrix $F(\alpha_{oe}, \theta)$ in (3.12) and (3.13) must be the identity matrix, I_N.

Having laid out the theoretical conditions for success of the continuation method, the next chapter will explore some of the practical and numerical aspects of applying it.

COMPUTATIONAL ASPECTS

In Ch. 3 conditions were presented that guarantee convergence of a continuation procedure to the OE global minimum. From a practical perspective this procedure involves solving a sequence of nonlinear minimization problems and checking the conditions of the Implicit Function Theorem at each step. These calculations, in turn, require certain derivatives to be computed. In many optimization problems, analytical calculations of the necessary derivatives are forbiddingly complex and numerical approximations via finite differences are used instead. For the problem structure described in Ch. 3 and discussed in detail in this chapter, it turns out that analytical expressions for the derivatives can be written and accurate values can be computed efficiently and easily. This chapter shows which derivatives must be computed and discusses efficient methods for their determination.

Necessary Derivatives

In Ch. 3 it was shown that if the OE residuals are modified by any linear procedure, the resulting residuals can be written,

$$R(\alpha, \theta) = F(\alpha, \theta) R_{OE}(\theta) \tag{4.1}$$

and the cost function can be written as

$$V(\alpha, \theta) = \frac{1}{2N} R^T(\alpha, \theta) R(\alpha, \theta) = \frac{1}{2N} R_{OE}^T(\theta) F^T(\alpha, \theta) F(\alpha, \theta) R_{OE}(\theta). \tag{4.2}$$

Here $F(\alpha,\theta)$ is an $N \times N$ matrix that performs the modification, $R_{OE}(\theta) = Y - \hat{Y}(\theta)$ is the $N \times 1$ vector of OE residuals, Y is the $N \times 1$ vector of measured values and $\hat{Y}(\theta)$ is the $N \times 1$ vector of simulated outputs for the $n \times 1$ parameter vector θ.

The Jacobian matrix of partial derivatives of the residuals with respect to the parameters is required for both the Gauss-Newton minimization update (see 3.24) and the check on the conditions of the IFT. This Jacobian is the $N \times n$ matrix (one column for each parameter),

$$J(\alpha,\theta) = \frac{\partial R(\alpha,\theta)}{\partial \theta^T} = \left[\frac{\partial R(\alpha,\theta)}{\partial \theta_1} \quad \dots \quad \frac{\partial R(\alpha,\theta)}{\partial \theta_n} \right]. \tag{4.3}$$

Introduce $J_k(\alpha,\theta)$ to denote the k'th column of the Jacobian (4.3). As will be explained shortly, it is easiest to write the equations for the Jacobian one column at a time. Differentiate (4.1) with respect to θ_k (using the product rule) to get the $N \times 1$ vector $J_k(\alpha,\theta)$,

$$J_k(\alpha,\theta) = \frac{\partial R(\alpha,\theta)}{\partial \theta_k} = \left(\frac{\partial F(\alpha,\theta)}{\partial \theta_k} R_{OE}(\theta) - F(\alpha,\theta) \frac{\partial \hat{Y}(\theta)}{\partial \theta_k} \right). \tag{4.4}$$

Note that the term $\frac{\partial F(\alpha,\theta)}{\partial \theta_k}$ is an $N \times N$ matrix. Therefore, there is no way to write the entire Jacobian, $J(\alpha,\theta)$, in a similar form as (4.4) compactly without resorting to Kronecker product notation [60]. Rather than use Kronecker notation, the Jacobian will be written one column at a time when necessary.

In Ch. 3 it was argued that the stationary points of the Gauss-Newton algorithm, $J^T(\alpha,\theta)R(\alpha,\theta) = 0$, define $\theta(\alpha)$ implicitly as long as $J^T(\alpha,\theta)R(\alpha,\theta)$ satisfies the conditions of the IFT. Notice that $J^T(\alpha,\theta)R(\alpha,\theta) = 0$ is a set of n equations. To satisfy the conditions of the IFT, $J^T(\alpha,\theta)R(\alpha,\theta)$ must be a continuously differentiable function of α and θ and the matrix of partial derivatives of $J^T(\alpha,\theta)R(\alpha,\theta)$ with

respect to θ must be invertible. Therefore, to check these conditions, the partial derivatives of $J^T(\alpha,\theta)R(\alpha,\theta)$ must be calculated. As noted above, $J^T(\alpha,\theta)R(\alpha,\theta)$ is $n \times 1$. Since α is assumed to be scalar, $\frac{\partial J^T(\alpha,\theta)R(\alpha,\theta)}{\partial \alpha}$ is an $n \times 1$ vector. Since θ is an $n \times 1$ vector, the matrix of partial derivatives, $D_\theta(\alpha,\theta) = \frac{\partial J^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta^T}$ is $n \times n$. Expressions for each of these partial derivatives can be found by applying the product rule to $J^T(\alpha,\theta)R(\alpha,\theta)$,

$$\frac{\partial J_k^T(\alpha,\theta)R(\alpha,\theta)}{\partial \alpha} = \frac{\partial J_k^T(\alpha,\theta)}{\partial \alpha}R(\alpha,\theta) + J_k^T(\alpha,\theta)\frac{\partial R(\alpha,\theta)}{\partial \alpha} \tag{4.5}$$

and

$$\frac{\partial J_k^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta_j} = H_{kj}^T(\alpha,\theta)R(\alpha,\theta) + J_k^T(\alpha,\theta)J_j(\alpha,\theta), \tag{4.6}$$

where the $N \times 1$ Hessian vector $H_{kj}(\alpha,\theta)$ is

$$\begin{aligned} H_{kj}(\alpha,\theta) = \frac{\partial J_k(\alpha,\theta)}{\partial \theta_j} = \Bigg(& \frac{\partial^2 F(\alpha,\theta)}{\partial \theta_j \partial \theta_k}R_{OE}(\theta) - \frac{\partial F(\alpha,\theta)}{\partial \theta_k}\frac{\partial \hat{Y}(\theta)}{\partial \theta_j} \\ & - \frac{\partial F(\alpha,\theta)}{\partial \theta_j}\frac{\partial \hat{Y}(\theta)}{\partial \theta_k} - F(\alpha,\theta)\frac{\partial^2 \hat{Y}(\theta)}{\partial \theta_j \theta_k}\Bigg). \end{aligned} \tag{4.7}$$

Two terms on the right side of (4.5) also require some unpacking. First,

$$\frac{\partial J_k(\alpha,\theta)}{\partial \alpha} = \frac{\partial^2 F(\alpha,\theta)}{\partial \alpha \partial \theta_k}R_{OE}(\theta) - \frac{\partial F(\alpha,\theta)}{\partial \alpha}\frac{\partial \hat{Y}(\theta)}{\partial \theta_k} \tag{4.8}$$

Second,

$$\frac{\partial R(\alpha,\theta)}{\partial \alpha} = \frac{\partial F(\alpha,\theta)}{\partial \alpha}R_{OE}(\theta) \tag{4.9}$$

While (4.5)-(4.9) may seem imposing, they follow from straightforward application of the product rule. Also, note that terms like $\frac{\partial \hat{Y}(\theta)}{\partial \alpha}$ drop out because the simulated output $\hat{Y}(\theta)$ does not depend on α.

The matrix of partial derivatives of the stationary point with respect to θ is populated by (4.6) with k and j evaluated over $1 \cdots n$,

$$D_\theta(\alpha,\theta) = \begin{bmatrix} \frac{\partial J_1^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta_1} & \cdots & \frac{\partial J_1^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial J_n^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta_1} & \cdots & \frac{\partial J_n^T(\alpha,\theta)R(\alpha,\theta)}{\partial \theta_n} \end{bmatrix}. \tag{4.10}$$

With all of the derivatives written explicitly, the conditions of the IFT can be stated compactly: (4.5) and (4.6) must be continuous for all k and j and (4.10) must be invertible.

Define one final piece of notation, the $n \times 1$ vector of partial derivatives of the stationary point with respect to α,

$$D_\alpha(\alpha,\theta) = \begin{bmatrix} \frac{\partial J_1^T(\alpha,\theta)R(\alpha,\theta)}{\partial \alpha} \\ \vdots \\ \frac{\partial J_n^T(\alpha,\theta)R(\alpha,\theta)}{\partial \alpha} \end{bmatrix}. \tag{4.11}$$

Then the IFT implies - see (3.27) - that,

$$\frac{d\bar{\theta}(\alpha)}{d\alpha} = -D_\theta^{-1}(\alpha,\bar{\theta})D_\alpha(\alpha,\bar{\theta}). \tag{4.12}$$

Equation (4.12) suggests that if the conditions of the IFT are satisfied at every point along the continuation, then an *explicit* n'th order system of (nonlinear) ODEs can be

written that describes the behavior of the global minimum as α varies. This intriguing observation will be explored in a later chapter.

Before closing this section, it should be re-emphasized that the reason that these derivatives need to be calculated is twofold: first, the Jacobian (4.3) is required by the Gauss-Newton (or similar) algorithm to compute the parameter updates; second, the continuation method relies on the stationary point satisfying the conditions of the IFT. Therefore, these conditions must be checked at each point along the continuation. Namely, (4.5) and (4.6) must be continuous for all k and j and (4.10) must be invertible.

Calculating the Necessary Derivatives

Careful examination of (4.4)-(4.9) reveals that all of the necessary derivatives described in the last section are linear combinations of the OE residual sequence, $R_{OE}(\alpha, \theta)$, and $F(\alpha, \theta)$, $\hat{Y}(\theta)$ and their first and second order partial derivatives. Therefore, if these first and second order derivatives are known exactly, the Gauss-Newton iterations and checks of the IFT conditions are trivial to compute. This section describes well-known methods to calculate these derivatives.

Partial Derivatives of $\hat{Y}(\theta)$

Throughout this document, $\hat{Y}(\theta)$ has been assumed to be the output of a simulation model - specifically a LTI state-space model,

$$\hat{Y}(\theta) = \begin{bmatrix} \hat{y}_0(\theta) \\ \hat{y}_1(\theta) \\ \vdots \\ \hat{y}_{N-1}(\theta) \end{bmatrix} \tag{4.13}$$

where the individual outputs $\hat{y}_k$ are defined by

$$\begin{aligned} \hat{x}_{k+1}(\theta) &= A(\theta)\hat{x}_k(\theta) + Bu_k \\ \hat{y}_k(\theta) &= C\hat{x}_k(\theta). \end{aligned} \tag{4.14}$$

Notice that only the A matrix is shown as a function of θ, which amounts to assuming that B and C are known. This choice was made to simplify the results, but the method outlined below applies with obvious modifications if the B and C matrices also contain unknown parameters to be estimated.

The goal is to calculate both $\frac{\partial \hat{Y}(\theta)}{\partial \theta_i}$ and $\frac{\partial^2 \hat{Y}(\theta)}{\partial \theta_j \partial \theta_i}$ which can be achieved by calculating the sequences $\frac{\partial \hat{y}_k(\theta)}{\partial \theta_i}$ and $\frac{\partial^2 \hat{y}_k(\theta)}{\partial \theta_j \partial \theta_i}$ for $k = 1 \cdots N$ and collecting them into vectors in the same way as in (4.13). Begin by differentiating both equations in (4.14) (making use of the product rule),

$$\begin{aligned} \frac{\partial \hat{x}_{k+1}(\theta)}{\partial \theta_i} &= A(\theta)\frac{\partial \hat{x}_k(\theta)}{\partial \theta_i} + \frac{\partial A(\theta)}{\partial \theta_i}\hat{x}_k(\theta) \\ \frac{\partial \hat{y}_k(\theta)}{\partial \theta_i} &= C\frac{\partial \hat{x}_k(\theta)}{\partial \theta_i}. \end{aligned} \tag{4.15}$$

If the change of variables $\gamma_k^{(i)} = \frac{\partial \hat{x}_k(\theta)}{\partial \theta_i}$ is made, then

$$\begin{aligned} \gamma_{k+1}^{(i)} &= A(\theta)\gamma_k^{(i)} + \frac{\partial A(\theta)}{\partial \theta_i}\hat{x}_k(\theta) \\ \frac{\partial \hat{y}_k(\theta)}{\partial \theta_i} &= C\gamma_k^{(i)} \end{aligned} \tag{4.16}$$

which makes it clear that the derivative sequence $\frac{\partial \hat{y}_k(\theta)}{\partial \theta_i}$ is simply the output of a state space system (4.16) with input vector $\hat{x}_k(\theta)$. Thus, $\frac{\partial \hat{Y}(\theta)}{\partial \theta_i}$ can be computed by first simulating the system (4.14) to get the sequence of state vectors $\hat{x}_k$, and then simulating the system (4.16). This process must be repeated for $i = 1 \cdots n$; therefore, a total of $n+1$ simulations are required to compute all first order partial derivatives of

$\hat{Y}(\theta)$ with respect to θ. The only hand calculation required of the user is $\frac{\partial A(\theta)}{\partial \theta_i}$ which can be very simple because the system matrices are often linear in the parameters.

It should be noted that the above procedure for calculating the derivatives is well-known. For instance, it is described in detail in [60] and Ljung uses it in his System Identification Toolbox for Matlab [27].

The various second order derivatives are calculated in a very similar manner. Simply differentiate (4.15) again,

$$\begin{aligned} \frac{\partial^2 \hat{x}_{k+1}(\theta)}{\partial\theta_j \partial\theta_i} &= A(\theta)\frac{\partial^2 \hat{x}_k(\theta)}{\partial\theta_j \partial\theta_i} + \frac{\partial A(\theta)}{\partial\theta_j}\frac{\partial \hat{x}_k(\theta)}{\partial\theta_i} + \frac{\partial^2 A(\theta)}{\partial j \partial\theta_i}\hat{x}_k(\theta) + \frac{\partial A(\theta)}{\partial\theta_i}\frac{\partial \hat{x}_k(\theta)}{\partial\theta_j} \\ \frac{\partial^2 \hat{y}_k(\theta)}{\partial\theta_j \theta_i} &= C\gamma_k^{(i)} \end{aligned} \tag{4.17}$$

Again, making a change of variables $\gamma_k^{(ij)} = \frac{\partial^2 \hat{x}_k(\theta)}{\partial\theta_j \partial\theta_i}$ gives,

$$\begin{aligned} \gamma_{k+1}^{(ij)} &= A(\theta)\gamma_k^{(ij)} + \frac{\partial A(\theta)}{\partial\theta_j}\frac{\partial \hat{x}_k(\theta)}{\partial\theta_i} + \frac{\partial^2 A(\theta)}{\partial\theta_j \partial\theta_i}\hat{x}_k(\theta) + \frac{\partial A(\theta)}{\partial\theta_i}\frac{\partial \hat{x}_k(\theta)}{\partial\theta_j} \\ \frac{\partial^2 \hat{y}_k(\theta)}{\partial\theta_j \theta_i} &= C\gamma_k^{(ij)} \end{aligned} \tag{4.18}$$

Thus, the second partial derivatives are also the output of a state-space system that can be simulated. Note that the system (4.18) has 3 inputs - $\hat{x}_k(\theta)$ (which is the same as for the first order derivatives), $\frac{\partial \hat{x}_k(\theta)}{\partial\theta_i}$, and $\frac{\partial \hat{x}_k(\theta)}{\partial\theta_j}$. All of these input state sequences must be simulated before (4.18) can be simulated. Fortunately, these state sequences are exactly the same sequences as in (4.15) which have already been simulated for the first order partial derivatives and just need to be saved.

While there is a certain amount of careful bookkeeping required, computationally the first and second order derivatives of $\hat{Y}(\theta)$ are simply the output of certain state-space systems (4.16) and (4.18) which can be simulated without difficulty. Matlab's built-in function *ltitr* (release R2014a) is a good, high-speed option.

Partial Derivatives of $F(\alpha, \theta)$

The point of describing the cost function modification procedure as a linear transformation of the OE residual sequence was to emphasize the amount of freedom available to the user. As mentioned above, the user is, in principle, free to specify every element of $F(\alpha, \theta)$ independently. This freedom is clearly an asset, but makes it difficult to make any general statements about the first and second order partial derivatives of $F(\alpha, \theta)$ with respect to θ and α. That said, a few general statements will be made and an expression for the derivative in the special case where $F(\alpha, \theta)$ takes the state estimator form of (3.14) will be derived.

Despite the apparent freedom in $F(\alpha, \theta)$, any candidate must comply with the assumptions of Theorem 3.0.1. These assumptions are discussed in detail at the end of Ch. 3. In brief, $F(\alpha, \theta)$ must be designed such that the sequence of global minima starts at a point where the global minimum can be reliably found given initial information, ends with the OE cost function (where $F(\alpha, \theta) = I_N$), and is a continuous function of α in between.

Clearly, if $F(\alpha, \theta)$ is designed so that it is not a function of θ, then $\frac{\partial F(\alpha,\theta)}{\partial \theta_j} = 0$ for all j and several of the terms in (4.5)-(4.9) drop out. This would significantly simplify the required computations and any *a priori* analysis that might be attempted. However, it is not clear that such an $F(\alpha)$ exists that still meets the design requirements. It has been hinted that the state estimator formulation of (3.14) appears to meet all of the requirements, at least for the sinusoid frequency estimation problem, but it is not obvious how the dependence of $F(\alpha, \theta)$ on θ might be removed while still meeting the requirements. The requirement that $F(\alpha_{oe}) = I_N$ is obviously achievable without dependence on θ. Likewise, continuity need not pose a significant challenge. The difficulty, then, is likely to be finding such an $F(\alpha)$ for which the

initial cost function is made convex to such a degree that the global minimum can be found reliably. As will be seen later, structures do indeed exist that can be expected to perform reasonably well.

As has been discussed, the state estimator formulation seems to meet the requirements of Theorem 3.0.1, at least for the sinusoid problem. For other problems, any formulation that is based on designing a state estimator can be written in the form of (3.14), the only possible difference is how L is designed. Thus, it seems useful to have a general way of calculating the required derivatives of $F(\alpha, \theta)$ for the specific form of (3.14).

For ease of presentation, (3.14) is repeated here,

$$F(\alpha,\theta) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -CL & 1 & 0 & \cdots & 0 \\ -C(A-LC)L & -CL & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ -C(A-LC)^{N-2}L & \cdots & -C(A-LC)L & -CL & 1 \end{bmatrix} \tag{4.19}$$

Reviewing (4.5)-(4.9) reveals that the necessary derivatives of $F(\alpha, \theta)$ are the first order partials, $\frac{\partial F(\alpha,\theta)}{\partial \theta_j}$ for $j = 1 \cdots n$ and $\frac{\partial F(\alpha,\theta)}{\partial \alpha}$ and the second order partials $\frac{\partial^2 F(\alpha,\theta)}{\partial \alpha \partial \theta_j}$ and $\frac{\partial^2 F(\alpha,\theta)}{\partial \theta_k \partial \theta_j}$ for $k, j = 1 \cdots n$.

In the general state-estimator formulation, L is a function of both α and θ (for the sinusoid problem, $L(\alpha, \theta) = \alpha L_0(\theta)$ where $L_0(\theta)$ is given by (2.32)). As above, A will be assumed to be a function of θ, but C will be assumed to be known and not a function of θ.

It would be tedious to find the partial derivatives listed above by differentiating each element of (4.19) directly. A better way is to first notice that each element is

part of the impulse response of the state-space system,

$$\begin{aligned} x_{k+1}(\alpha,\theta) &= (A(\theta) - L(\alpha,\theta)C)x_k + L(\alpha,\theta)u_k \quad , \quad x_0 = 0 \\ y_k(\alpha,\theta) &= -Cx_k(\alpha,\theta), \end{aligned} \tag{4.20}$$

where u_k is the unit impulse function. It is easy to see that $y_0 = -CL$, $y_1 = -C(A - LC)L$, $y_2 = -C(A - LC)^2L$, etc. Thus, the outputs of (4.20) are exactly the elements of $F(\alpha,\theta)$. Having written the elements of $F(\alpha,\theta)$ as the output of a state-space system, the partial derivatives can be calculated using the same method as in the previous section. Once the derivatives are calculated using this method, the matrix $F(\alpha,\theta)$ can be populated directly. For instance, $\frac{\partial F(\alpha,\theta)}{\partial \theta_j}$ is populated by the terms $\frac{\partial y_k(\alpha,\theta)}{\partial \theta_j}$ where $y_k(\alpha,\theta)$ is defined in (4.20).

Omitting the derivations (which proceed exactly as in the previous section), the state-space systems associated with each of the partial derivatives listed above will be shown. The first order partial with respect to θ_j can be calculated using,

$$\begin{aligned} \gamma_{k+1}(\alpha,\theta) &= (A(\theta) - L(\alpha,\theta)C)\gamma_k + \left(\frac{\partial A(\theta)}{\partial \theta_j} - \frac{\partial L(\alpha,\theta)}{\partial \theta_j}C\right)x_k + \frac{\partial L(\alpha,\theta)}{\partial \theta_j}u_k \\ \frac{\partial y_k(\alpha,\theta)}{\partial \theta_j} &= -C\gamma_k(\alpha,\theta), \end{aligned} \tag{4.21}$$

where the state sequence x_k is an input to (4.21) and must be first generated by simulating (4.20). The first order partial with respect to α can be calculated using,

$$\begin{aligned} \gamma_{k+1}(\alpha,\theta) &= (A(\theta) - L(\alpha,\theta)C)\gamma_k - \frac{\partial L(\alpha,\theta)}{\partial \alpha}Cx_k + \frac{\partial L(\alpha,\theta)}{\partial \alpha}u_k \\ \frac{\partial y_k(\alpha,\theta)}{\partial \alpha} &= -C\gamma_k(\alpha,\theta). \end{aligned} \tag{4.22}$$

The second order partial with respect to θ can be calculated using,

$$\begin{aligned}
\gamma_{k+1} &= (A(\theta) - L(\alpha,\theta)C)\gamma_k + \left(\frac{\partial A(\theta)}{\partial \theta_j} - \frac{\partial L(\alpha,\theta)}{\partial \theta_j}C\right)\frac{\partial x_k}{\partial \theta_i} \\
&+ \left(\frac{\partial A(\theta)}{\partial \theta_i} - \frac{\partial L(\alpha,\theta)}{\partial \theta_i}C\right)\frac{\partial x_k}{\partial \theta_j} + \left(\frac{\partial^2 A(\theta)}{\partial \theta_i \partial \theta_j} - \frac{\partial^2 L(\alpha,\theta)}{\partial \theta_i \partial \theta_j}C\right)x_k + \frac{\partial^2 L(\theta)}{\partial \theta_i \partial \theta_j}u_k \qquad (4.23) \\
\frac{\partial^2 y_k(\theta)}{\partial \theta_i \theta_j} &= C\gamma_k,
\end{aligned}$$

where the state sequence x_k is, again, an input and must be first generated by simulating (4.20) and $\frac{\partial x_k(\theta)}{\partial \theta_j}$ are also inputs which have already been generated if the first order partial system (4.21) has been simulated for $j = 1 \cdots n$.

Finally, the second order partial with respect to θ and α can be calculated using,

$$\begin{aligned}
\gamma_{k+1} &= (A(\theta) - L(\alpha,\theta)C)\gamma_k + \left(\frac{\partial A(\theta)}{\partial \theta_j} - \frac{\partial L(\alpha,\theta)}{\partial \theta_j}C\right)\frac{\partial x_k}{\partial \alpha} \\
&- \frac{\partial L(\alpha,\theta)}{\partial \alpha}C\frac{\partial x_k}{\partial \theta_j} - \frac{\partial^2 L(\theta)}{\partial \alpha \partial \theta_j}Cx_k + \frac{\partial^2 L(\theta)}{\partial \alpha \partial \theta_j}u_k \qquad (4.24) \\
\frac{\partial^2 y_k(\theta)}{\partial \alpha \theta_j} &= C\gamma_k,
\end{aligned}$$

where the required input state sequences are generated by simulating (4.20)-(4.22).

It has been argued that the state estimator formulation for $F(\alpha,\theta)$ in (4.19) is generally useful for modifying the OE cost function such that the global minimum can be found easily at one end of the continuation and slowly reducing the gain of $L(\alpha,\theta)$ causes the cost function to approach OE. Clearly, the design of $L(\alpha,\theta)$ is the key step. While this section does not offer any direct guidelines for this design, it provides methods for calculating the necessary derivatives to apply the IFT check. That is, once $L(\alpha,\theta)$ is designed, this section describes the computations that must be performed to establish the continuity of $\bar{\theta}(\alpha)$ as α moves from α_0 to α_{oe}. As was

shown in Ch. 3, if continuity can be established, the method is guaranteed to converge to the OE global minimum.

Conclusions

This chapter applies the partial derivative equations of the IFT to the implicit definition of the global minimum $J^T(\alpha,\theta)R(\alpha,\theta) = 0$ for the general situation described in Ch. 3 where $R(\alpha,\theta) = F(\alpha,\theta)R_{OE}(\theta)$. This results in several equations (4.5)-(4.9) for derivatives that need to be calculated before the IFT check can be applied. These equations are high level expressions and it is not immediately clear how they should be computed. A method is derived for calculating the derivatives as the output of certain state-space systems. Finally, it is observed that $F(\alpha,\theta)$ is a design choice for the continuation method and, therefore, general statements about its various derivatives cannot not be made. However, the state estimation method that arises in the context of sinusoid frequency estimation does give a specific form for $F(\alpha,\theta)$ and formulas for its derivatives are derived.

It is important to note that, while this chapter derives formulas for derivatives that must be calculated in order to apply the IFT check, no design criteria have yet been given for the transformation matrix $F(\alpha,\theta)$ that would, *a priori*, ensure success of the IFT method. That said, the main requirement of the IFT - that the matrix (4.10) be invertible - is a kind of indirect design criteria on $F(\alpha,\theta)$ in that $F(\alpha,\theta)$ and its derivatives show up in (4.10). Unfortunately, because the relationship between $F(\alpha,\theta)$ and its derivatives and the matrix (4.10) is quite complicated, any direct design requirements on $F(\alpha,\theta)$ are likely to be problem-dependent and difficult to ascertain. Consequently, other arguments for good choices for $F(\alpha,\theta)$ will likely

be required. One such argument leads to the state estimator formulation and another good choice will be described in subsequent chapters.

SINUSOID FREQUENCY ESTIMATION REVISITED

Given the theoretical developments in the previous two chapters, it is useful to return to the sinusoid frequency estimation problem. In this chapter, the state estimation method will be examined in light of the IFT results. Also, another method for designing the transformation matrix $F(\alpha, \theta)$ will be introduced and tested.

Applying the IFT to Sinusoid Frequency Estimation

The two main results of the IFT are conditions for the existence of a continuous function defining the stationary point as a function of α and an explicit representation for the derivative of this function with respect to α. Both of these require the computation of various derivatives as described in Ch. 4. This section begins by setting up the state-space systems that must be simulated in order to calculate these derivatives.

Calculating the Derivatives

Recall from Ch. 2 that the state-space simulation model for the family of sinusoids with frequency parameter ϕ is

$$\hat{x}_{k+1}(\phi) = A(\phi)\hat{x}_k(\phi) = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix} \hat{x}_k(\phi) \quad , \quad \hat{x}_0 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{5.1}$$

$$\hat{y}_k(\phi) = C\hat{x}_k(\phi) = \begin{bmatrix} 1 & 0 \end{bmatrix} \hat{x}_k(\phi). \tag{5.2}$$

As in Ch. 4, the simulated output samples will be collected in to the $N \times 1$ column vector $\hat{Y}(\phi)$. The output sequence of (5.2) is known exactly - $\hat{y}_k(\phi) = \sin(\phi k)$ - and

so its derivatives can be computed easily. Namely,

$$\frac{\partial \hat{y}_k(\phi)}{\partial \phi} = k\cos(\phi k), \tag{5.3}$$

and

$$\frac{\partial^2 \hat{y}_k(\phi)}{\partial \phi^2} = -k^2\sin(\phi k). \tag{5.4}$$

The remaining required derivatives are the terms populating $F(\alpha, \phi)$. For the state estimation method, the matrix $F(\alpha, \phi)$ takes the form

$$F(\alpha,\phi) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -CL & 1 & 0 & \cdots & 0 \\ -C(A-LC)L & -CL & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ -C(A-LC)^{N-2}L & \cdots & -C(A-LC)L & -CL & 1 \end{bmatrix} \tag{5.5}$$

where the dependence of A and L on α and ϕ was suppressed for visual clarity and the observer is given by $L(\alpha, \phi) = \alpha L_0(\phi)$ and

$$L_0(\phi) = \begin{bmatrix} 2\cos(\phi) \\ \frac{\cos^2(\phi)-\sin^2(\phi)}{\sin(\phi)} \end{bmatrix}. \tag{5.6}$$

Application of the equations of Ch. 4 requires the computation of certain derivatives of the matrices $A(\phi)$ and $L(\alpha, \phi)$. These derivatives are,

$$\frac{\partial A(\phi)}{\partial \phi} = \begin{bmatrix} -\sin(\phi) & \cos(\phi) \\ -\cos(\phi) & -\sin(\phi) \end{bmatrix}, \tag{5.7}$$

$$\frac{\partial^2 A(\phi)}{\partial \phi^2} = \begin{bmatrix} -\cos(\phi) & -\sin(\phi) \\ \sin(\phi) & -\cos(\phi) \end{bmatrix}, \tag{5.8}$$

$$\frac{\partial L(\alpha,\phi)}{\partial \phi} = \alpha \begin{bmatrix} -2\sin(\phi) \\ \frac{\cos(\phi)(2\cos^2(\phi)-3)}{\sin^2(\phi)} \end{bmatrix}, \tag{5.9}$$

$$\frac{\partial L(\alpha,\phi)}{\partial \alpha} = L_0, \tag{5.10}$$

$$\frac{\partial^2 L(\alpha,\phi)}{\partial \phi^2} = \alpha \begin{bmatrix} -2\cos(\phi) \\ \frac{2\sin^4(\phi)-\sin^2(\phi)+2}{\sin^3(\phi)} \end{bmatrix}, \tag{5.11}$$

and

$$\frac{\partial^2 L(\alpha,\phi)}{\partial \alpha \partial \phi} = \begin{bmatrix} -2\sin(\phi) \\ \frac{\cos(\phi)(2\cos^2(\phi)-3)}{\sin^2(\phi)} \end{bmatrix}. \tag{5.12}$$

Equations (5.7)-(5.12) can be substituted directly into the state-space systems in (4.21)-(4.24). These systems can then be simulated and the outputs used to populate the various derivatives of $F(\alpha,\phi)$ required in (4.5)-(4.9). Finally, these equations are used to calculate $D_\phi(\alpha,\phi)$ and $D_\alpha(\alpha,\phi)$ - defined in (4.10) and (4.11), respectively - which are both scalars for the sinusoid frequency estimation problem.

Let $\bar{\phi}(\alpha)$ be the value of ϕ that minimizes the cost function globally and is defined implicitly by $J^T(\alpha,\phi)R(\alpha,\phi) = 0$. Then the IFT implies that $\bar{\phi}(\alpha)$ is a continuously differentiable function anywhere $D_\phi(\alpha,\phi) \neq 0$ and is continuous and, in turn, Theorem 3.0.1 implies that the global minimum of the OE cost function can be found by minimizing a sequence of cost functions starting at $\alpha = 1$ (PE) and ending at $\alpha = 0$ (OE).

Simulation Results

For continuity of presentation, simulations will be performed with the same test conditions as in Ch. 2. The frequency of the signal is $\phi = 0.1$, the number of points is $N = 2000$, the noise is additive, white Gaussian with a modest signal-to-noise ratio (SNR) of about 17 dB. The measured signal y_k is shown in Fig. 5.1. The simplest

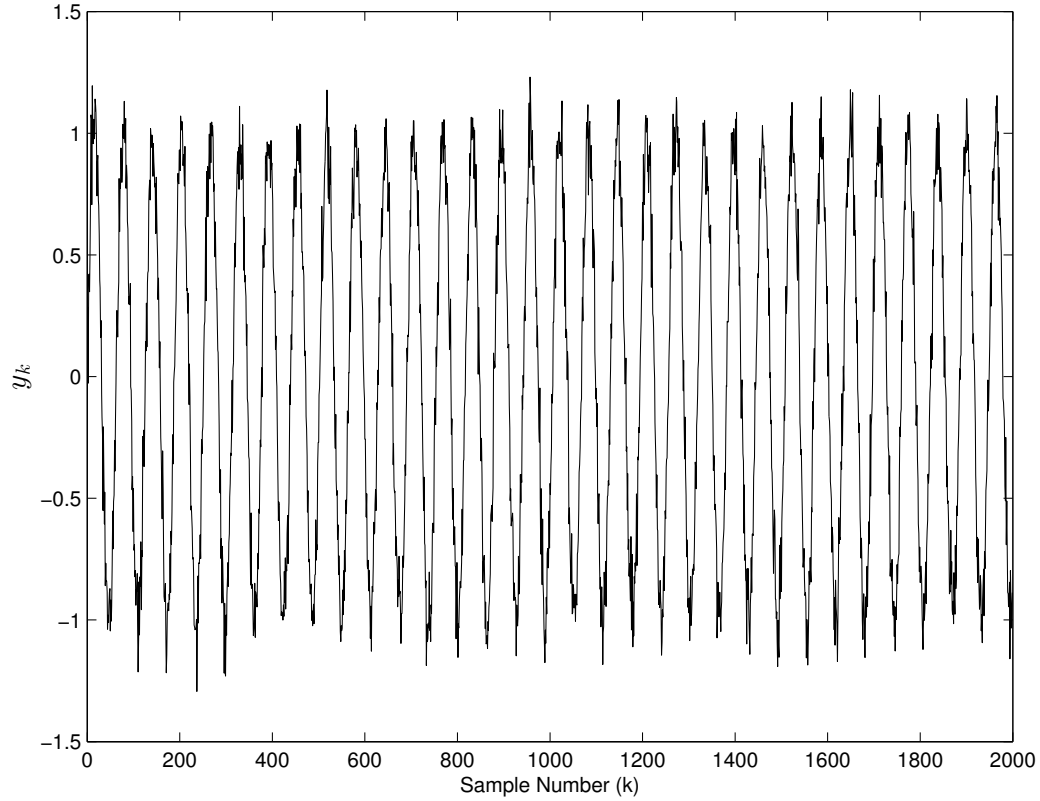

Figure 5.1: Measured signal sequence whose frequency is to be estimated.

test of the IFT ideas of Ch. 3 is to take small steps in α starting at $\alpha = 1$ (PE), and invoking the NLLS solver at each point with the solution $\bar{\phi}$ from the previous step as the initial guess for the next step. This was done in Ch. 2. The additional step that can guarantee this procedure will converge to the OE global minimum is to apply the IFT check at each step. As long as $D_\phi(\alpha, \phi)$ is continuous and non-zero at each step,

the conditions of the IFT are satisfied. Fig. 5.2 shows the sequence $\bar{\phi}(\alpha)$ that results from taking 2000 equally-spaced steps in α. For the beginning of the sequence from $\alpha = 1$ to roughly $\alpha = 0.1$, $\bar{\phi}(\alpha)$ varies smoothly, indicating continuity. Near OE, as the enlarged portion of Fig. 5.2 shows, $\bar{\phi}(\alpha)$ makes some rapid movements, indicating potential continuity problems. These rapid movements are almost certainly inherent to the problem structure (i.e., there is some inherent sensitivity to the measurement noise) and not due to numerical problems. As will be seen below, the step size can be dramatically decreased and the rapid variations persist.

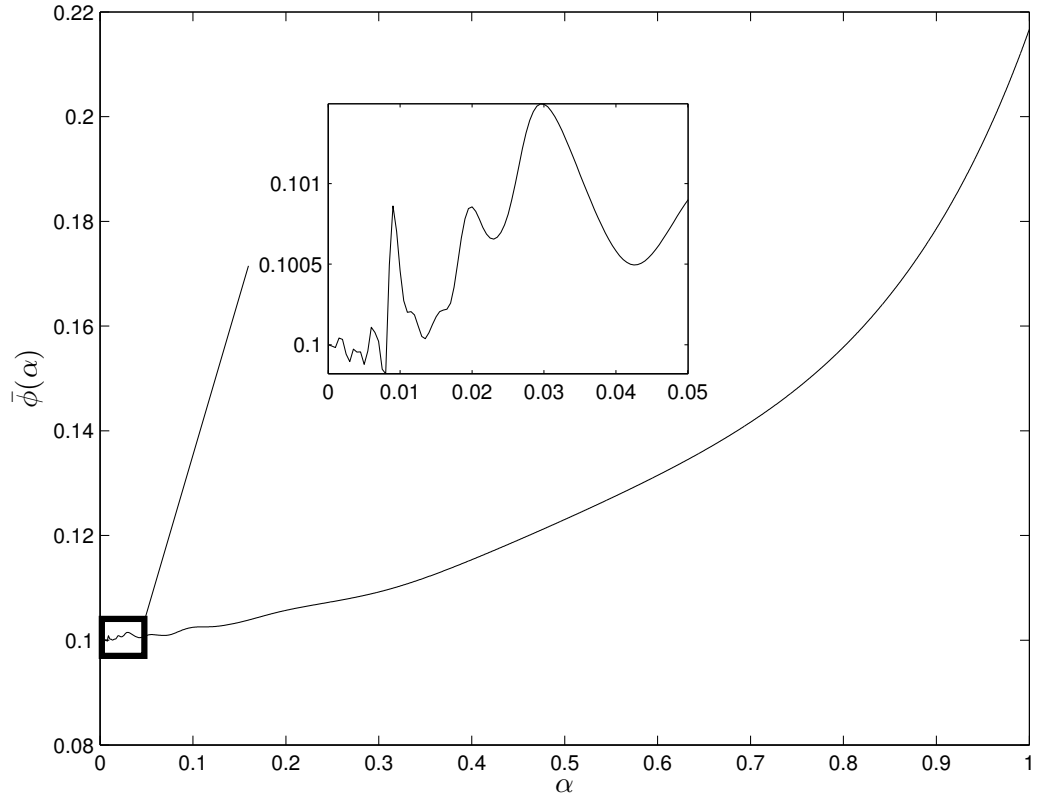

Figure 5.2: Global minimum of cost function as α moves from PE ($\alpha = 1$) to OE ($\alpha = 0$). The enlarged region shows potential problems because the minimum is moving rapidly.

The true test of continuity is to ensure that $D_\phi(\alpha, \phi)$ - a scalar - is non-zero. Fig. 5.3 shows $D_\phi(\alpha, \phi)$ evaluated at every solution point in Fig. 5.2. The plot was broken into two regions for visual clarity because $D_\phi(\alpha, \phi)$ gets very large for small α. Because $D_\phi(\alpha, \phi)$ is large for all points along the continuation, $\bar{\phi}(\alpha)$ is always

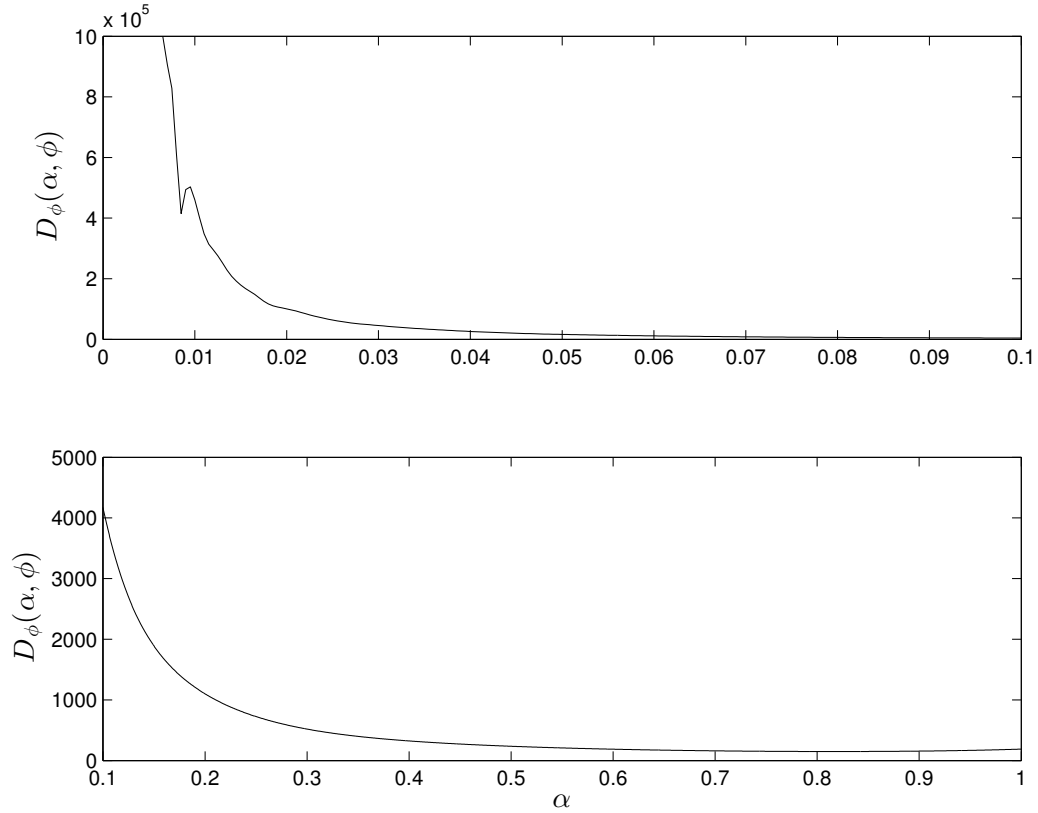

Figure 5.3: $D_\phi(\alpha, \phi)$ evaluated at every solution calculated by the NLLS solver. It is always large and never gets close to 0.

continuous. However, Fig. 5.3 seems to imply that $\bar{\phi}(\alpha)$ should be moving slowly for small α because, as explained in Ch. 4, the IFT says that the derivative of $\bar{\phi}(\alpha)$ with respect to α is

$$\frac{d\bar{\phi}(\alpha)}{d\alpha} = -D_\phi^{-1}(\alpha, \bar{\phi}) D_\alpha(\alpha, \bar{\phi}). \tag{5.13}$$

Even though $D_\phi(\alpha, \phi)$ is large near OE, $\bar{\phi}(\alpha)$ can still move rapidly if $D_\alpha(\alpha, \bar{\phi})$ is also large. Fig. 5.4 shows (5.13) for each point along the continuation. Even though the curve starts to behave strangely near OE, the magnitude is always small, which means that $\bar{\phi}(\alpha)$ never makes any unreasonable steps. The final point of the continuation

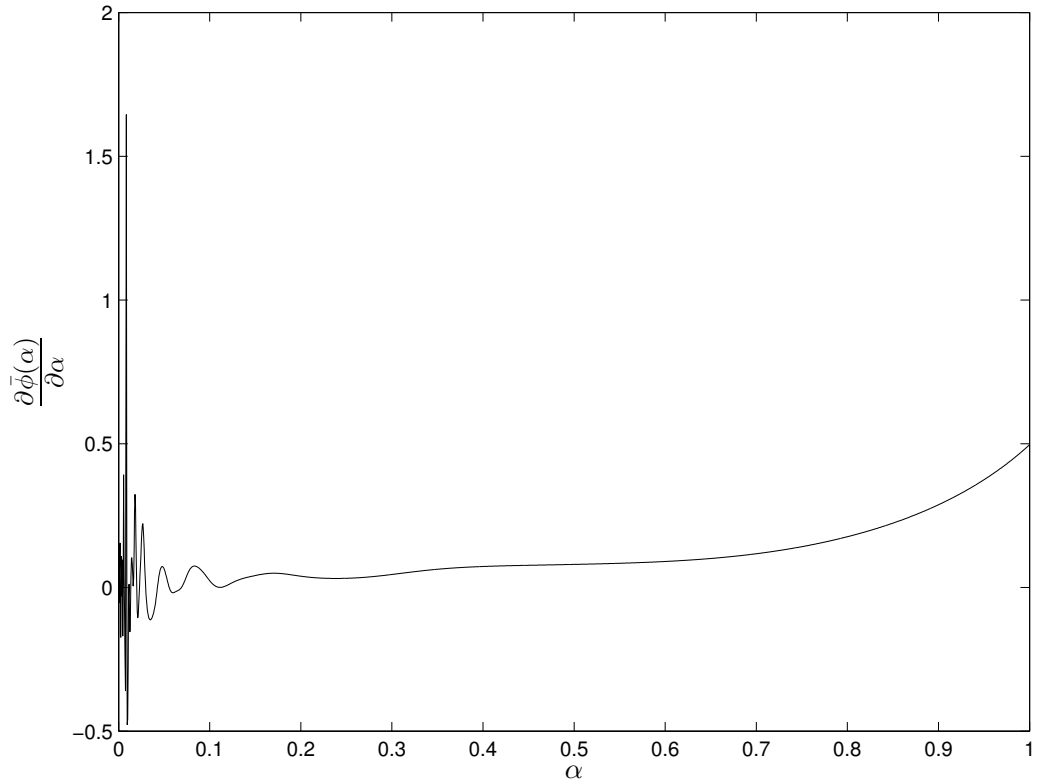

Figure 5.4: Plot of (5.13) at each point along the continuation. Small magnitude at all points implies likely success of the method.

is $\bar{\phi}(0) = 0.1000000182$ which is indeed the global minimum of the OE cost function. Therefore, the method was successful; further, this success was predictable using the diagnostic tools of the IFT.

Solving the ODE Directly

As briefly noted in Ch. 4, (5.13) presents an intriguing alternative possibility for following the continuation. Recall that explicit expressions are available for both of the terms on the right side of (5.13) which means that (5.13) is a differential equation describing the dynamic behavior of $\bar{\phi}(\alpha)$. Once initialized at $\bar{\phi}(1)$, this differential equation can be integrated numerically from $\alpha = 1$ to $\alpha = 0$.

The change in perspective here is significant. The problem has shifted from running an NLLS solver at many steps to solving a nonlinear differential equation. Consequently, there is potential for applying the tools of nonlinear dynamics to the problem. This potential is mostly left for future work, but a few remarks will be made. First, it is natural to ask, from the perspective of nonlinear dynamics, does the OE global minimum represent a stable attractor for the ODE (5.13)? Asked another way, does the ODE still converge to the OE global minimum if it is initialized in some neighborhood of, but not exactly at, $\bar{\phi}(1)$? Numerical simulations shown in Fig. 5.5 suggest this is the case. The blue curve was initialized at $\bar{\phi}(1)$ as computed by the NLLS solver and consequently follows the curve in Fig. 5.2. All of the other curves were initialized elsewhere but still converged to $\bar{\phi}(0)$ (to within the tolerance of the ODE solver). This rather remarkable result deserves emphasis. Once the differential equation is known, all that is needed is a rather poor initial guess to ensure convergence to the OE global minimum. Without ever invoking a NLLS solver, a user can, nonetheless, have a reasonable expectation of finding the global minimum of the OE cost function by numerically solving an ODE. It should be noted that, while Fig. 5.5 suggests that the convergence region of the ODE is quite large, convergence is not global. Initialization points can be found from which the ODE diverges.

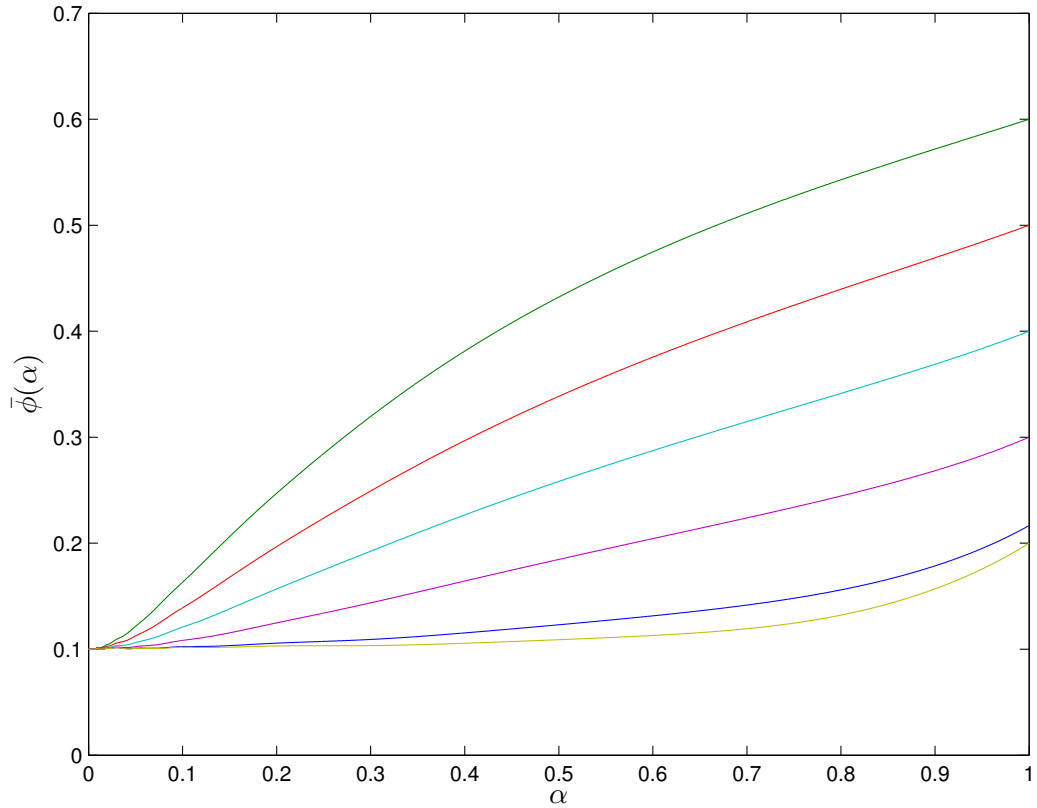

Figure 5.5: Plot of numerical solutions to (5.13) initialized at several different points. The OE global minimum is apparently an attractor for a wide range of initial conditions.

Analysis of a Failure

This section induces a failure of the conditions of the IFT and attempts to explain why it occurred. Recall that the conditions of the IFT were presented as *sufficient* conditions for success of the continuation method. As such, failure of the conditions does not necessarily imply failure of the method. In the case examined in this section, the conditions fail while the method still succeeds and the reason is apparent.

Decreasing the SNR can exacerbate the somewhat pathological behavior of $\bar{\phi}(\alpha)$ shown in the enlarged portion of Fig. 5.2. In this section, the SNR was decreased to

roughly 11 dB. Fig. 5.6 shows the sequence of global minima produced by the NLLS solver for the same setup as in Fig. 5.2, except the increased noise. At a glance, the behavior in the enlarged portion does not seem much more severe in the lower SNR case. However, examination of $D_\phi(\alpha, \phi)$ at each point along the continuation proves otherwise. Fig. 5.7 shows $D_\phi(\alpha, \phi)$ focusing on the narrow range of α where continuity problems appear to be occurring. Notice that, unlike in the higher SNR case, $D_\phi(\alpha, \phi)$ crosses zero as α gets small. This zero-crossing violates the conditions of the IFT, but $\bar{\phi}(\alpha)$ still converges to the OE global minimum.

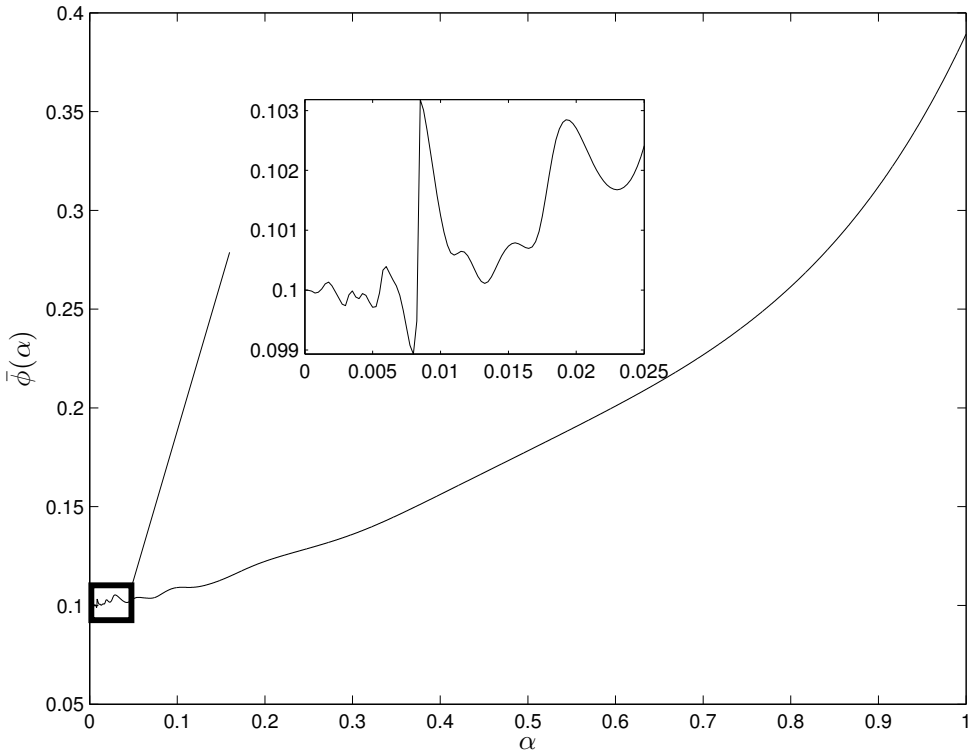

Figure 5.6: Global minimum of cost function as α moves from PE ($\alpha = 1$) to OE ($\alpha = 0$) for higher SNR case.

A plot of the cost function in the immediate vicinity of the discontinuity reveals both the reason the curve is discontinuous and why the sequence of NLLS solutions

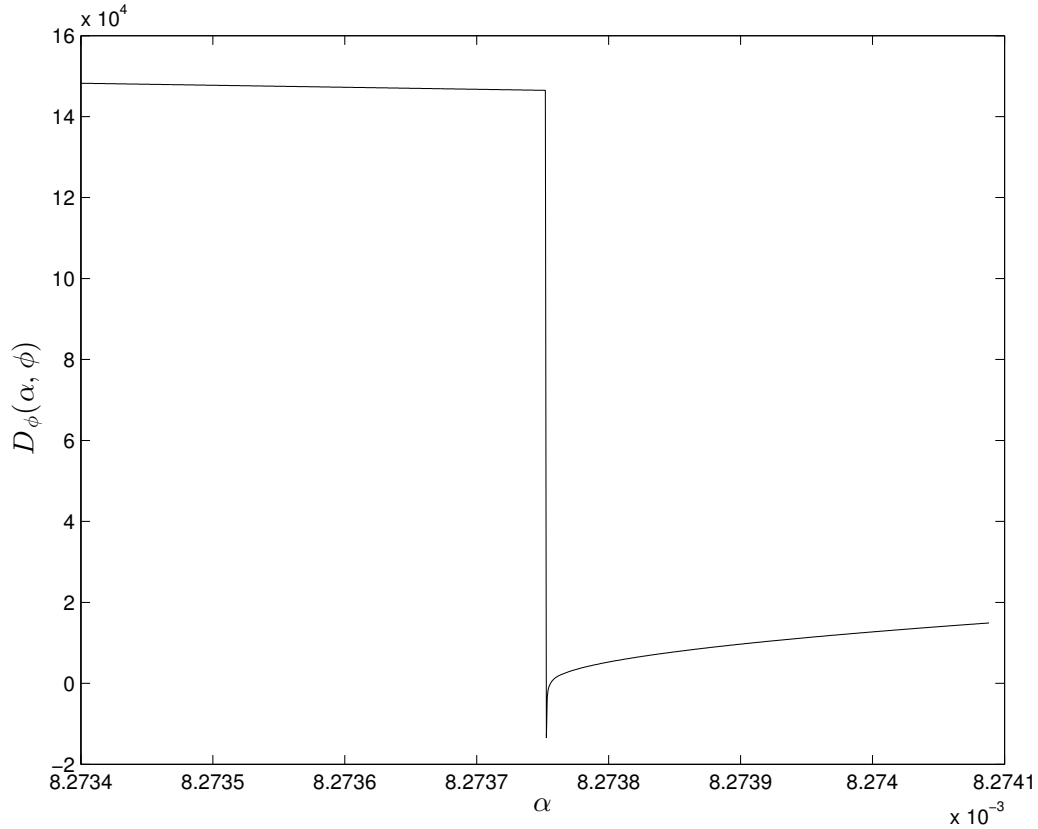

Figure 5.7: $D_\phi(\alpha, \phi)$ evaluated at solutions calculated by the NLLS solver over a narrow range of α. It crosses 0 and appears to be discontinuous, both violations of the IFT conditions.

still converges to the OE global minimum. Fig. 5.8 shows the cost function for a range of values of α in the region of the discontinuity. The color bar gives the values of α. Before the discontinuity, at the black end of the color scale, the global minimum is on the right of the figure. At some point, the minimum on the left dips lower than the minimum on the right and becomes the global minimum. Notice, however, that because of the local maximum in the middle of the figure, the transition from the right minimum to the left minimum is discontinuous. Also notice that by the green end of the color scale, the minimum on the right has disappeared, and the global minimum on the left is the only extremal point (at least over the range of ϕ shown).

Imagine how the NLLS solver proceeds through this sequence. At the black end of the scale, the solver easily finds the minimum on the right. Near the pink portion, the solver will still find the minimum on the right (because it was initialized somewhere very close to it), even though it is no longer the global minimum. As α decreases further, there is a point where the minimum on the right is no longer an extremal point of the curve (derivative is not 0), and the NLLS solver will be forced to leave the right side of the curve and simply follow it downward and find the minimum on the left. This explains why the NLLS method can still succeed in converging to the OE global minimum even when $\bar{\phi}(\alpha)$ is discontinuous and the conditions of the IFT are not met.

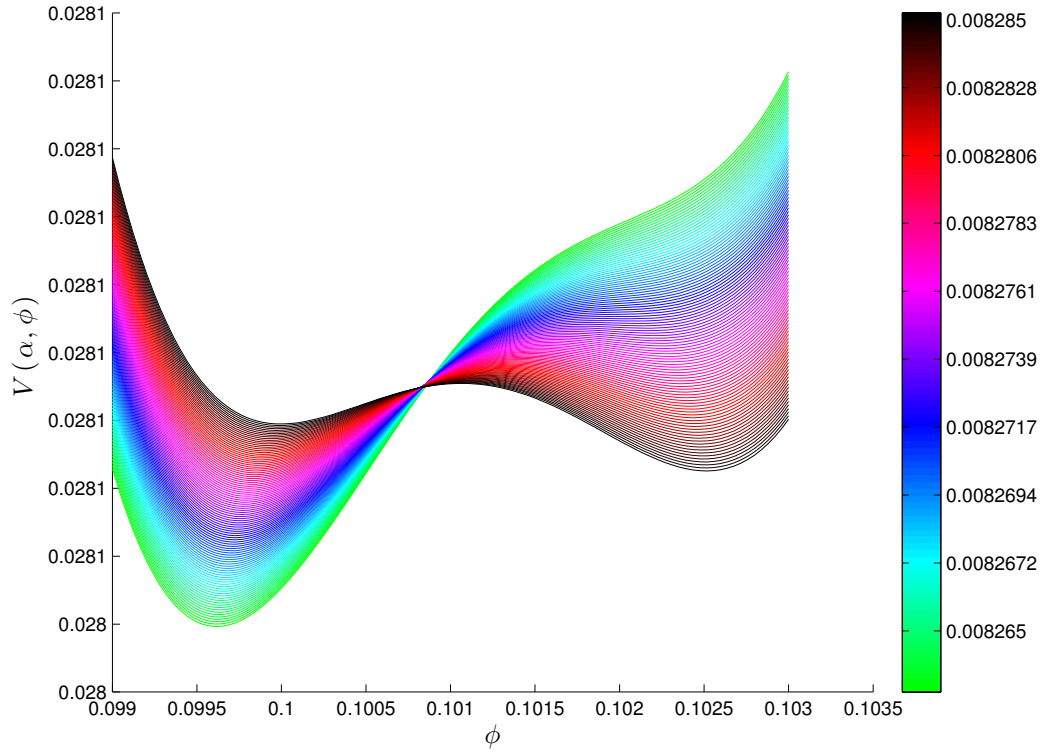

Figure 5.8: Cost function for many closely spaced values of α, all near in the region of the discontinuity of $\bar{\phi}(\alpha)$. The color bar indicates the values of α. At the black end of the color scale, the global minimum is on the right. At some point, the minimum passes through the inflection point and ends up on the left.

To reemphasize, the conditions of the IFT are sufficient, not necessary, conditions for the success of the continuation method and failure to meet them does not imply failure of the method. On the other hand, solving the ODE (5.13) directly *does* require the conditions of the IFT to be met because, in general, discontinuities in the solution are fatal to numerical methods. Therefore, while the ODE method may yield theoretical insights, as a practical method of solving the problem, it is less robust than the NLLS continuation method.

An Alternative Design for $F(\alpha, \phi)$

In Ch. 2 it was noted that the cost function becomes more complex and the basin of attraction of the global minimum becomes smaller as the number of data points N increases. This can be seen in Fig. 2.1 and is made precise in (2.18). While the present work has only considered this phenomena in the context of sinusoid frequency estimation, in [41] it is argued that the situation is much more general. For many systems, the cost function is more convex if the number of data points is smaller.

This observation leads naturally to a new qualitative design for $F(\alpha, \phi)$. At the beginning of the continuation (near α_0) $F(\alpha, \phi)$ should be mostly zeros except in the upper left corner. This will result in most of the OE residuals being multiplied by 0, effectively reducing the number of data points. The only residuals that will contribute to the cost function will be the first few. As α increases toward α_{oe}, $F(\alpha, \phi)$ should smoothly approach the identity matrix. Perhaps the simplest way to achieve these qualitative features is to make $F(\alpha, \phi)$ a diagonal matrix at all points along the continuation. Near α_{oe}, most of the diagonal elements should be 0, and a few at the beginning should be 1. As α moves from α_0 to α_{oe}, more and more of the diagonal elements should smoothly become non-zero until at α_{oe} they should all be 1.

An attractive function to achieve the smooth transition is the decaying exponential. Specifically, if the elements of $F(\alpha, \phi)$ are denoted by $f_{ij}(\alpha, \phi)$, then in the proposed scheme the diagonal elements of $F(\alpha, \phi)$, are given by

$$f_{kk}(\alpha) = \begin{cases} 1, & \text{if } k <= n_0; \\ e^{-\alpha(k-n_0-1)}, & \text{if } n_0 + 1 \leq k \leq N. \end{cases} \tag{5.14}$$

All of the elements of $F(\alpha, \phi)$ not on the main diagonal are 0. From (5.14) several useful observations can be made. The first n_0 diagonal elements of $F(\alpha, \phi)$ are 1, regardless of the value of α. This is to ensure that the NLLS solver is properly constrained, even at the beginning of the continuation. In practice, α_0 could be any positive number. The larger the number, the nearer to 0 all of the elements will be except the first n_0. Finally, when $\alpha = 0$, $F(\alpha, \phi)$ is the identity matrix, so the continuation starts at some positive number α_0 and ends at $\alpha_{oe} = 0$.

From the above discussion, it is clear that $F(\alpha, \phi)$ should meet the endpoint requirements of the continuation method (i.e., that $F(\alpha_0, \phi)$ should yield a cost function that can be globally minimized easily and that $F(\alpha_{oe}, \phi) = I_N$). The remaining requirement is that $F(\alpha, \phi)$ lead to a continuous global minimum $\bar{\phi}(\alpha)$. As in the state estimation design, this continuity cannot be established *a priori*. The conditions of the IFT must be checked at each point. However, the design scheme in (5.14) does have two attractive features related to the IFT. First, it leads to a transformation matrix that is not a function of ϕ, so it can be written $F(\alpha)$ and several of the derivatives required for the IFT are 0. Second, the derivative of $F(\alpha)$ with respect to α is easy to compute, does not require the state-space computations of Ch. 4, and the derivative is smooth and gets small as α approaches $\alpha_{oe} = 0$.

Simulation results for the exponential weighting design for $F(\alpha)$ will be given in the following section. The derivatives required for applying the IFT are omitted because most are the same as for the state estimation method. The only differences are regarding the derivatives of $F(\alpha, \phi)$ which are trivial for the exponential weighting method.

Simulation Results

The exponential weighting method was simulated for the same sinusoidal signal as was used for the state estimation method earlier in this chapter. The additional parameters for the exponential weighting method were $\alpha_0 = 1$ and $n_0 = 4$. Fig. 5.9 shows the sequence of minima calculated by the NLLS procedure with 2000 equally-spaced steps in α. Notice that the procedure converges to the OE global minimum when $\alpha = 0$ and the variation is smooth throughout. Because of this smoothness, continuity problems seem very unlikely, but $D_\phi(\alpha, \phi)$ should be assessed to be sure. Fig. 5.10 shows that $D_\phi(\alpha, \phi)$ is never near 0. Therefore, continuity is ensured. For comparison to the state-estimation method (see Fig. 5.4), Fig. 5.11 shows the derivative of $\bar{\phi}(\alpha)$ with respect to α, given by (5.13). Notice that the exponential weighting method yields a much smoother derivative. This smooth derivative indicates that the ODE is well-behaved and should be easy to solve numerically.

Fig. 5.12 shows the result of solving the ODE (5.13) directly for the exponential weighting method and for several different initial points. As in the state estimation method, the OE global minimum seems to be an attractor for a relatively wide range of initial conditions.

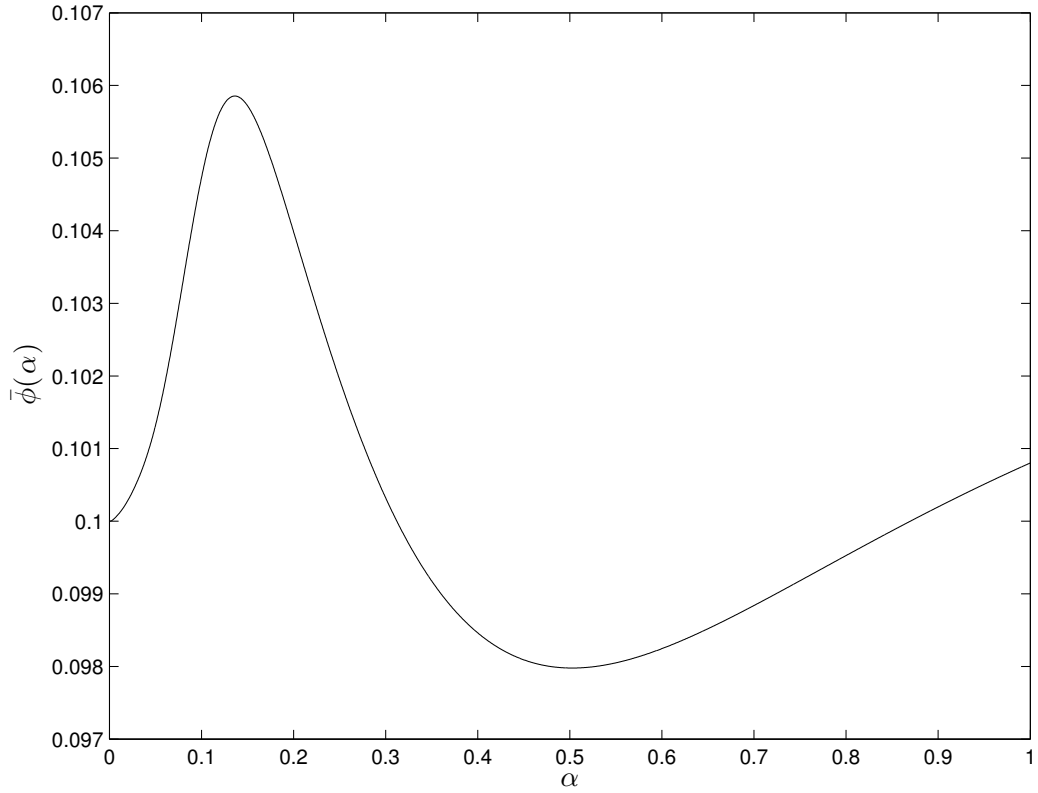

Figure 5.9: Sequence of global minima for exponential weighting method. Variation is small and very smooth, indicating that there are unlikely to be continuity issues.

Conclusions

This chapter shows the application of the IFT ideas to the sinusoid frequency estimation problem. Two distinct methods for the design $F(\alpha, \phi)$ are shown to result in convergence of the continuation method to the OE global minimum. It is shown that, remarkably, the OE global minimum can also be found by solving the ODE (5.13) numerically. The existence of this ODE is guaranteed by the IFT. Further it is shown that the convergence of this ODE to the OE global minimum is relatively insensitive to the point at which it is initialized. This implies the somewhat surprising result that the OE global minimum can be found without ever invoking a NLLS procedure.

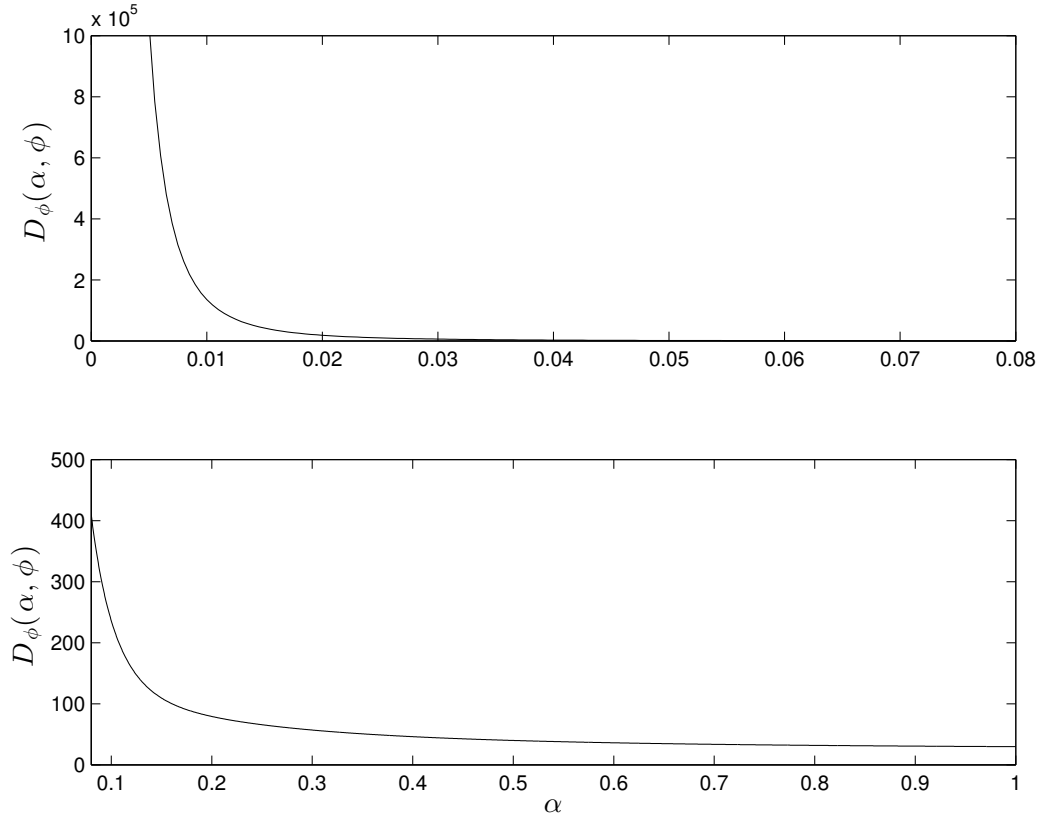

Figure 5.10: $D_\phi(\alpha, \phi)$ evaluated at every solution calculated by the NLLS solver for the exponential weighting method. It is never close to 0.

Comparing the results of the state estimation method with the exponential weighting method yields several useful insights. First, the two methods are quite different in their motivation and structure of $F(\alpha, \phi)$. The fact that they both work suggests that the transformation matrix framework and IFT ideas are quite general and admit many solutions. The exponential weighting method, in particular, was developed in a fairly ad hoc manner based on a few elementary observations, raising the exciting possibility that effective design methods for $F(\alpha, \theta)$ may be plentiful and all of them can be analyzed and understood using the tools provided by this work.

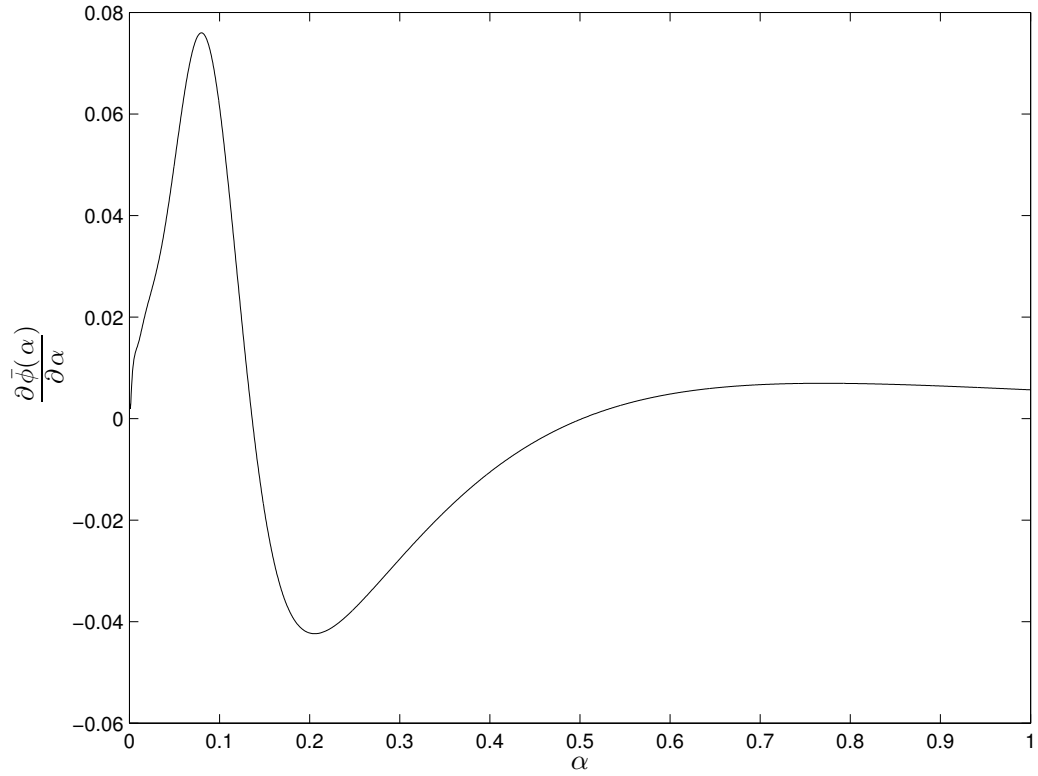

Figure 5.11: Plot of (5.13) at each point along the continuation for the exponential weighting method. Small magnitude and smooth variation indicates a well-behaved ODE.

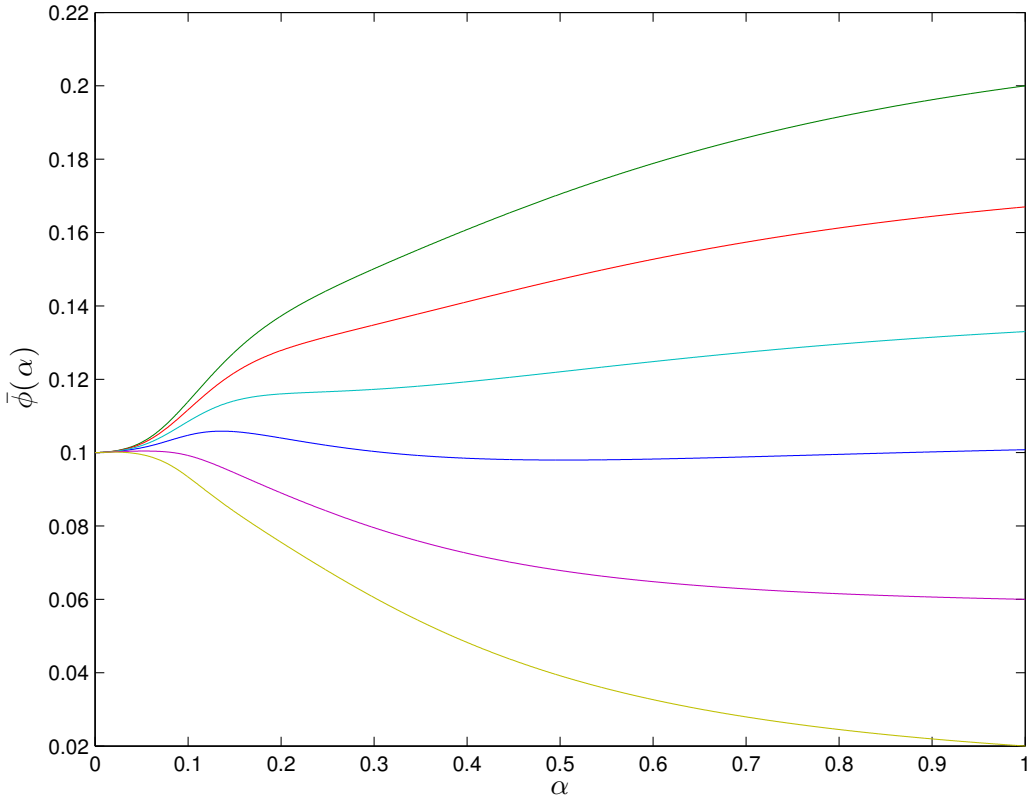

Figure 5.12: Plot of numerical solutions to (5.13) initialized at several different points for the exponential weighting method. The OE global minimum is apparently an attractor for a wide range of initial conditions.

EXAMPLES

All of the novel theoretical insights resulting from this work have been introduced and applied to the sinusoid frequency estimation problem. In this chapter, they will be applied to two more complicated parameter estimation problems. First, two parameters of a nonlinear system will be estimated. Second, four parameters of a 4th order linear model consisting of two underdamped modes will be estimated. Practical aspects of applying the continuation method will be discussed.

Duffing Oscillator

The Duffing Oscillator is a very well-studied (e.g., see [23]) second-order differential equation with a cubic nonlinearity whose general form is

$$\frac{d^2y}{dt^2} + \theta_3\frac{dy}{dt} + \theta_2 y + \theta_1 y^3 = f(y,t) \tag{6.1}$$

where $y(t)$ is interpreted as a displacement, θ_j are the model parameters, and $f(y,t)$ is a forcing function. Depending on the signs and values of the parameters, the Duffing Oscillator has different physical interpretations. In the present work, the undamped ($\theta_3 = 0$), unforced ($f(y,t) = 0$) Duffing Oscillator will be considered. Thus, the goal will be to estimate the parameters θ_1 and θ_2 of

$$\frac{d^2y}{dt^2} + \theta_2 y + \theta_1 y^3 = 0. \tag{6.2}$$

In [41] it was shown that, when θ_2 is negative and θ_1 is positive and much smaller than θ_2 (i.e., the nonlinearity is small), a complicated OE cost function results for which many initial guesses will not converge to the global minimum. These same conditions

will be adopted here. To apply the methods presented in the previous chapters, (6.2) should be converted to state-space form. First, reparameterize by changing the sign of θ_2 so that both parameters are positive. Introduce the state variables $x_1(t) = y(t)$ and $x_2(t) = \frac{dy(t)}{dt}$. Then (6.2) can be written

$$\begin{aligned} \frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= \theta_2 x_1(t) - \theta_1 x_1(t)^3 \\ y(t) &= x_1(t). \end{aligned} \tag{6.3}$$

Ch. 5 presented two possibilities for designing $F(\alpha, \theta)$ - one method based on designing a deadbeat observer for the system and another method based on ignoring some of the data at the beginning of the continuation. It is not obvious how a deadbeat observer might be designed for this nonlinear system. Further, as noted in Ch. 5, the exponential weighting design method is based on an observation in [41] where it was shown to be successful when applied to the Duffing Oscillator. Finally, the exponential weighting method is both computationally and conceptually simpler than the observer method. For all of these reasons, it makes sense to at least start with the exponential weighting method.

With the design for $F(\alpha, \theta)$ chosen, the next major step in applying the continuation method is computing the various derivatives required by the IFT. The computational techniques outlined in Ch. 4 were derived assuming a discrete-time system, but the Duffing Oscillator is a continuous-time system. In theory, the general method for writing the derivatives as the output of certain state-space systems can be extended to continuous-time systems with little difficulty. However, the simulations required to calculate these derivatives for continuous-time systems may be much more computationally expensive. To see why, recall that the state-space systems from Ch.

4 used as inputs the states of the original system and the procedure was to simulate the original system first so that these states were available. For discrete-time systems, all of the systems are simulated for exactly the same sampling instants. Once the original system is simulated, all of the samples that could be required as inputs for the derivative system are known.

The situation is more complicated for continuous-time systems. Simulation of the derivative systems will be carried out by an ODE solver (because they, like the original system, are continuous-time systems). An accurate ODE solver can, in general, require samples of the input at any point in the simulation interval. Since the inputs to the derivative systems are the states of the original system which must also be computed using the ODE solver, arbitrary accuracy can only be achieved by calling the solver for the original system while the solver is running for the derivative system. In other words, the ODE solver must be run in a nested fashion (in fact, the nesting must go one layer deeper for the second derivatives). The exception is, of course, when a closed-form solution to the ODE is known. In practice, this nesting is likely computationally prohibitive and some approximations must be made.

One obvious approximation method would be to simply simulate the original system at a finite number of points (possibly many more points than will ultimately be used in the cost function) and interpolate whenever the ODE solver for the derivative systems calls for a point not in the record. However, since an approximation must be made, there is no obvious reason not to simply forego the simulation of the derivative systems altogether and simply use a finite difference approximation. That is, simulate the original system (6.3) for the current parameter set, perturb the parameters slightly, and calculate the derivative based on the change observed in the output sequence. For the following simulation results, all of the derivatives for the IFT were calculated using finite differences.

Simulation Results

The Duffing Oscillator was simulated for $\theta_1 = 0.5$ and $\theta_2 = 5$ with $N = 300$ data points sampled at an interval of 0.1 s starting at $t = 0$. The initial conditions were $x_1(0) = 1$, $x_2(0) = 0$. All ODE solutions were computed using Matlab's built-in *ode45* (release R2014a) solver. Fig. 6.1 shows $x_1(t)$ and $x_2(t)$ for these simulation conditions and Fig. 6.2 shows the associated phase-plane plot. For the identification procedure, a small amount of Gaussian white noise was added to $y(t)$. The noisy output measurements are shown in Fig. 6.3.

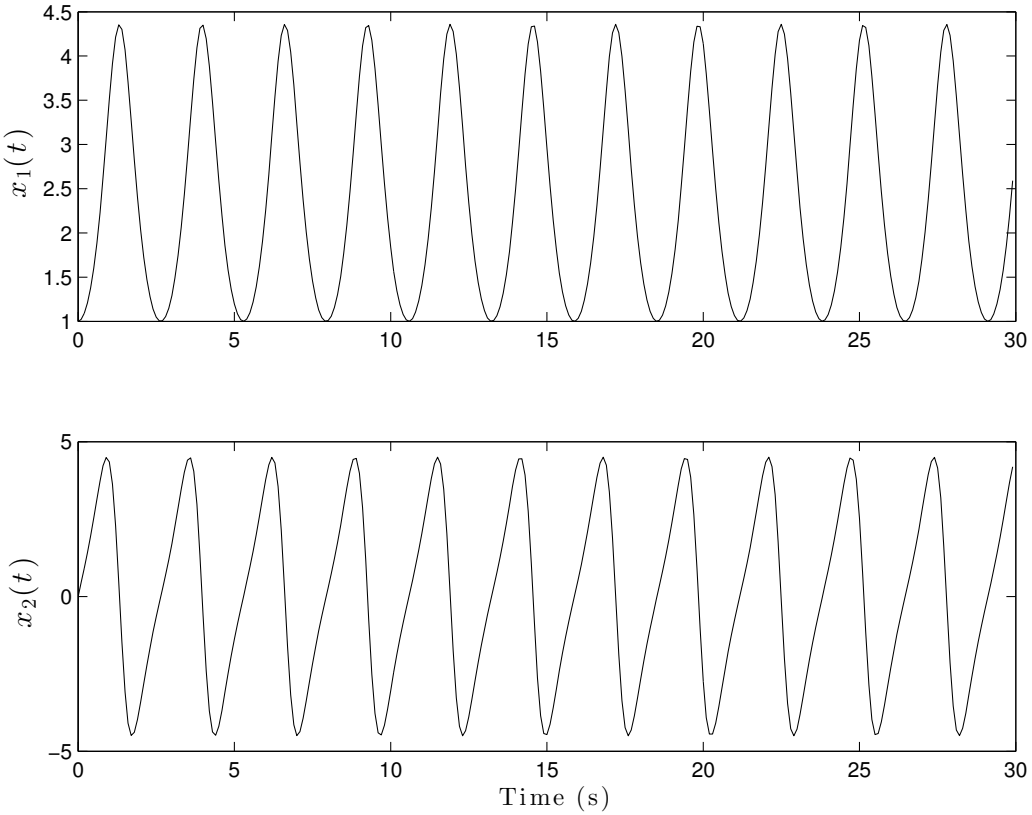

Figure 6.1: States of Duffing Oscillator.

As described in the previous section, the exponential weighting method was chosen for constructing $F(\alpha)$ (see (5.14) and the surrounding discussion). The

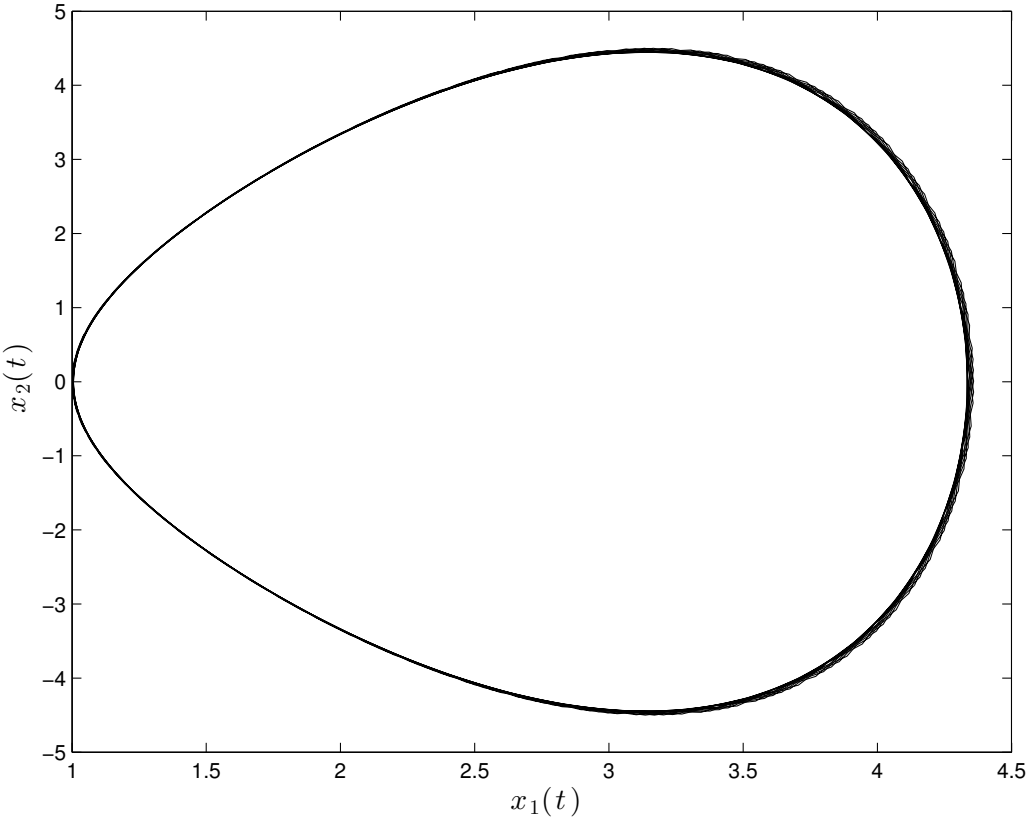

Figure 6.2: Phase plane plot of Duffing Oscillator.

parameters for this design were chosen to be $n_0 = 4$ and $\alpha_0 = 1$. Just as in the sinusoid problem, α will vary from 1 to 0. For the continuation method to be effective, the cost function must be much more convex at α_0 than at α_{oe}. Otherwise, either the method is unneeded or does not offer improvement over OE. Fig. 6.4 shows the logarithm of the OE cost function. The logarithm was used to improve the visual clarity of the plot. Because the logarithm is a monotone function, it preserves the relative position of each point and therefore does not alter the qualitative behavior of the cost function with respect to local minima or gradient directions. Fig. 6.4 clearly shows that a good initial guess will be required to reach the global minimum and the continuation method has room to improve the situation.

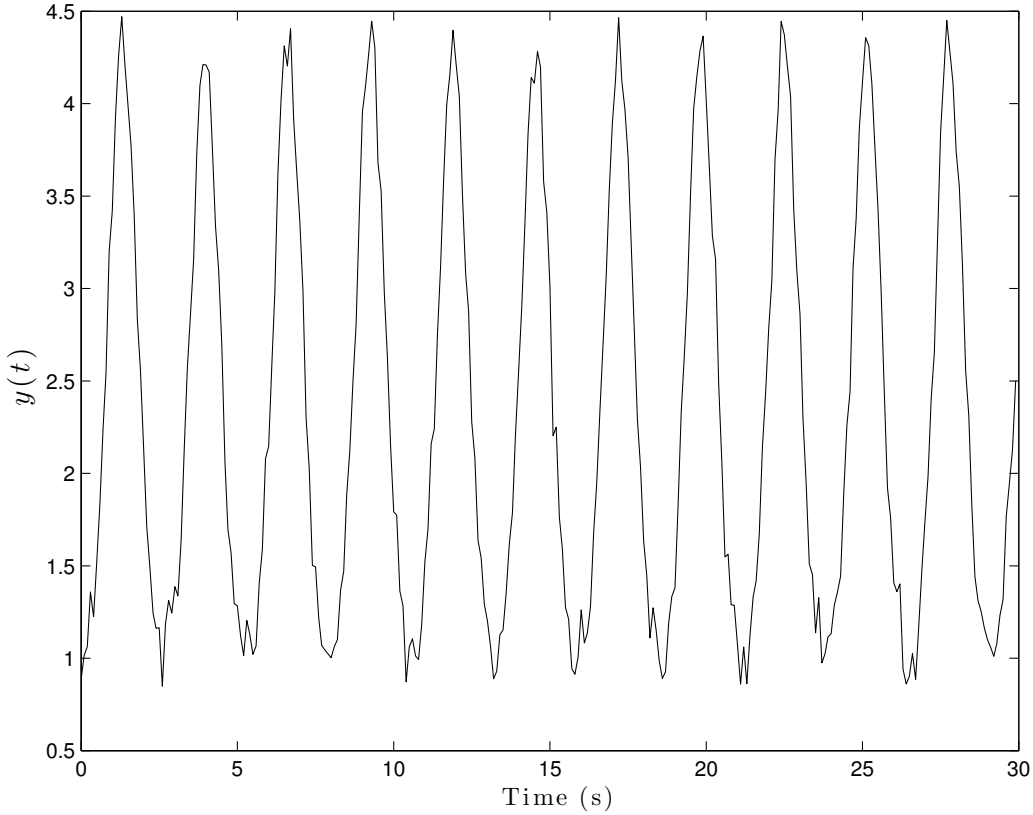

Figure 6.3: Noisy measurement data for Duffing Oscillator.

Fig. 6.5 shows the logarithm of the cost function when $\alpha = 1$. It is difficult to visually locate the minimum along the trough so it was marked with a red star. This cost function is much smoother than for OE and appears to have a single minimum over the region shown. The method has succeeded in producing a cost function that is much easier to globally minimize. Thus, the endpoint conditions of the continuation method are satisfied. All that remains is to establish the continuity of the global minimum as α varies. As with the sinusoid problem, this will be done by taking small steps in α, computing the minimum at each step via NLLS, and evaluating the invertibility of the matrix of partial derivatives $D_\theta(\alpha, \bar{\theta})$ in (4.10). If this matrix is invertible at each step, convergence to the OE global minimum is assured.

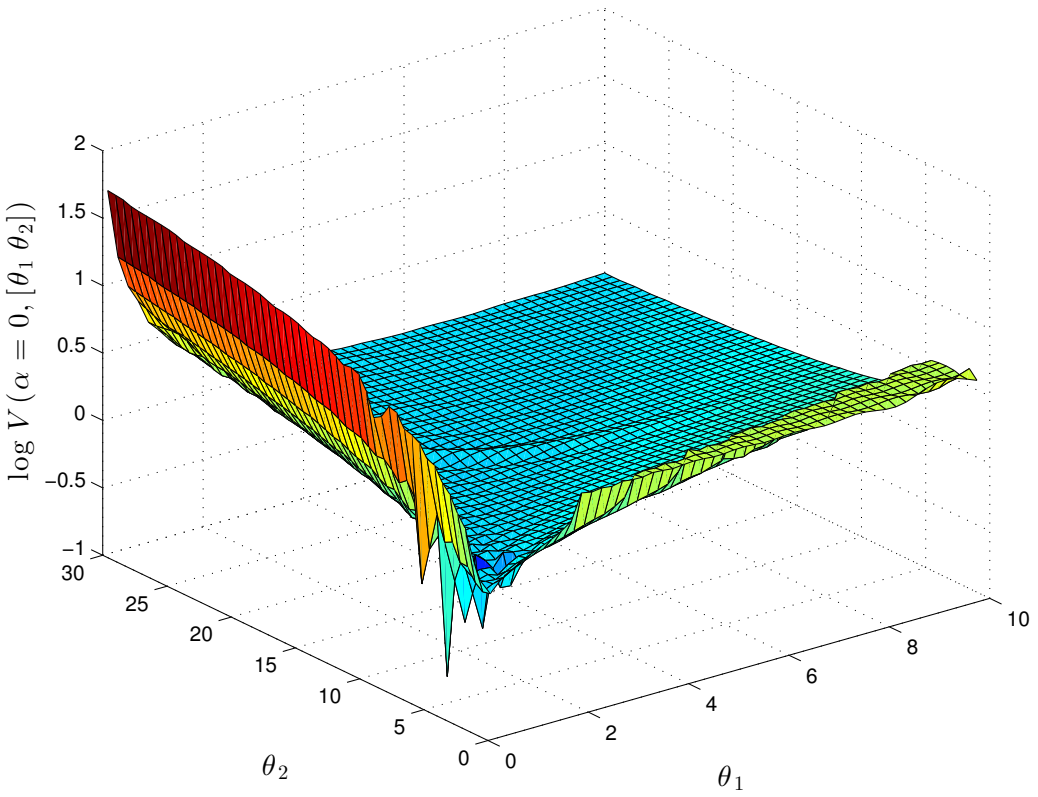

Figure 6.4: Logarithm of cost function at $\alpha = 0$ (OE).

Fig. 6.6 shows the globally minimizing parameter values as α varies from 1 to 0. Notice that the parameters start far from the correct values which indicates large bias near α_0. This should be expected because near α_0 only a few points contribute to the cost function so even a small amount of noise can have a large effect. Both curves appear smooth so continuity problems are not expected. Fig. 6.7 shows the condition number of $D_\theta(\alpha, \bar{\theta})$ whose invertibility will ensure continuity. Condition number is a useful proxy for invertibility [15]. If the condition number is very large, the matrix may be nearly singular.

As can be seen in Fig. 6.7, the condition number is reasonably small at all points indicating that the sequence of global minima is continuous. The rapid fluctuations at the beginning appear to be due to numerical issues associated with the finite difference

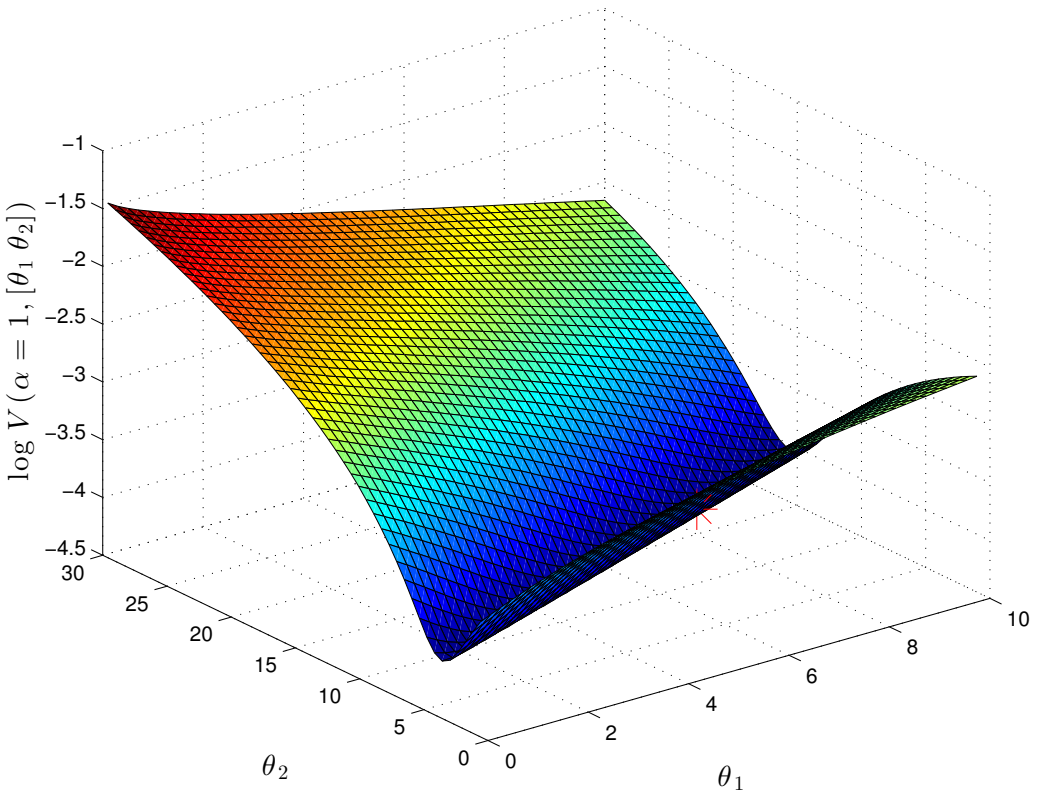

Figure 6.5: Logarithm of cost function at $\alpha = 1$.

approximation. If the location of the OE global minimum had not been known in advance, it still could have been concluded from Theorem 3.0.1 that the continuation procedure had succeeded.

The continuation procedure has been successfully applied to a nonlinear system with a fairly complicated OE cost function. A major difference from previous examples is that the Duffing Oscillator model is continuous-time. This led to the derivatives for the IFT being computed using finite differences instead of the state-space simulation method of Ch. 4.

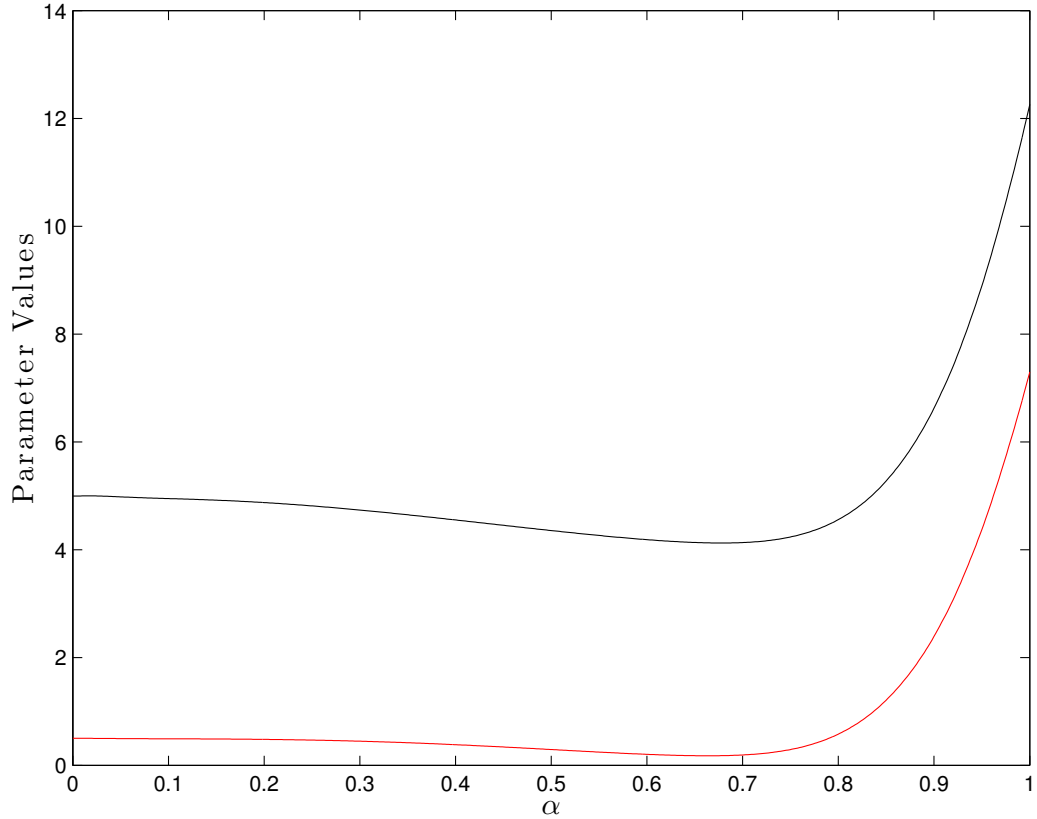

Figure 6.6: Parameter values as α varies. θ_1 is in red and θ_2 in black. Each converges to the correct value.

Two Underdamped Modes

A common data fitting problem is estimating the parameters of multiple decaying modes that have been summed. In this example, the parameters controlling the damping and oscillation frequency of two decaying modes will be estimated by fitting a discrete-time OE model to their sum. Four free parameters will be estimated - two for each mode. As will be seen, this problem has the usual difficulty that the OE cost function has a narrow convergence region, but also adds the additional complication that the eigenvalues are near the unit circle and bad steps can easily lead to instability.

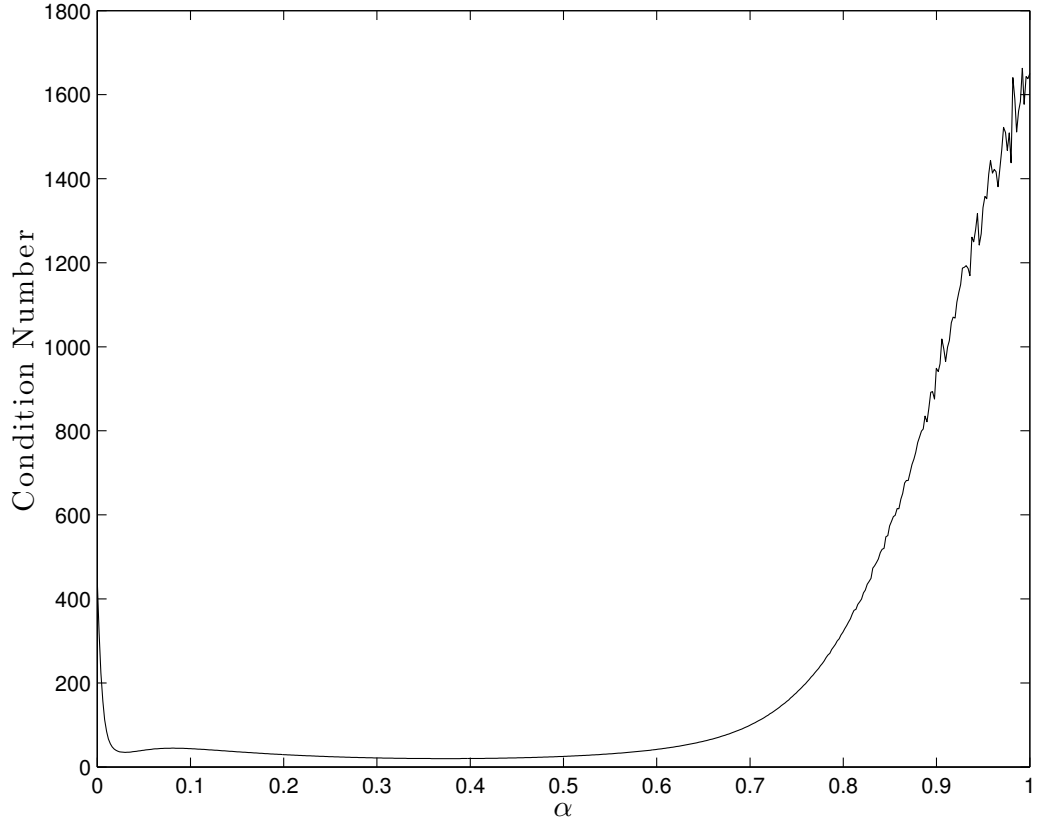

Figure 6.7: Condition number of $D_\theta(\alpha, \bar{\theta})$ as α varies. All values are of reasonable magnitude and do not threaten continuity. Larger condition numbers indicate difficulty in obtaining the inverse.

The state-space model is

$$x_{k+1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\theta_1 & -\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\theta_3 & -\theta_4 \end{bmatrix} x_k \quad , \quad x_0 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \tag{6.4}$$

$$y_k = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} x_k.$$

The parameter vector $\theta = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}^T$ was set to $\theta = \begin{bmatrix} 0.9940 & -1.9049 & 0.9216 & -1.8817 \end{bmatrix}^T$ so that the eigenvalues of the modes were $0.997e^{\pm i0.3}$ and $0.96e^{\pm i0.2}$. A plot of the output of (6.4) with white Gaussian noise added so that the SNR is roughly 21 dB is shown in Fig. 6.8.

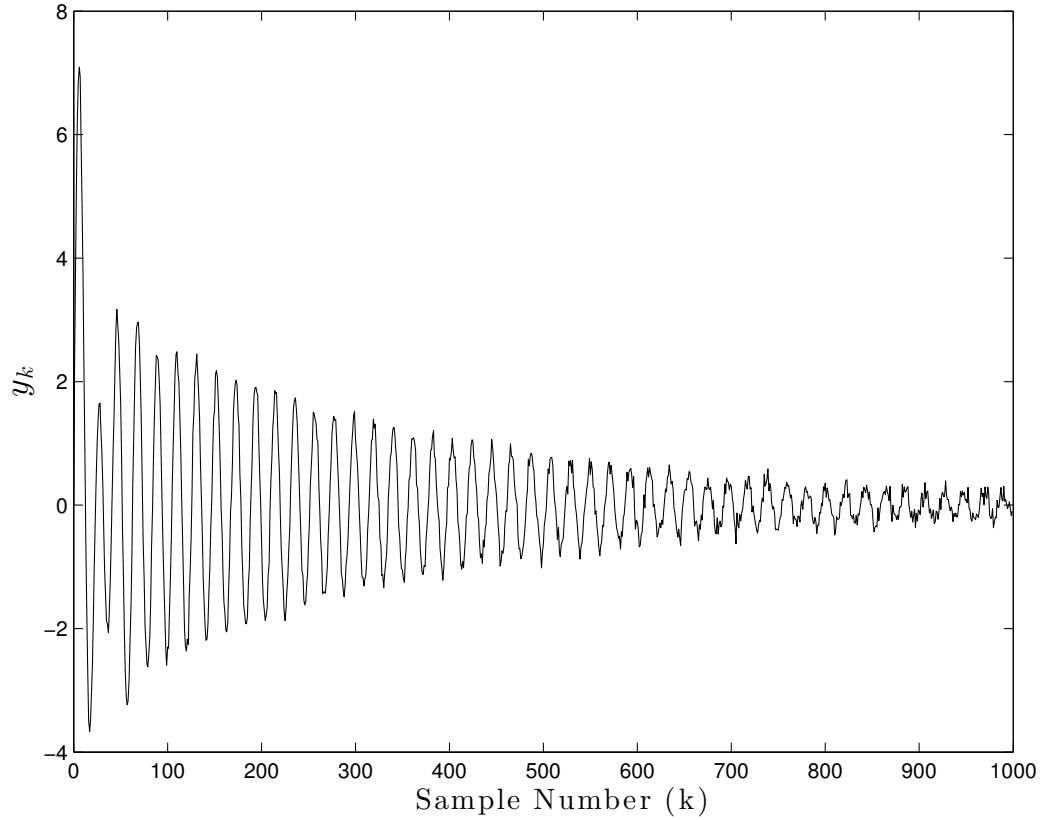

Figure 6.8: Output of two-mode system with SNR of 21 dB.

As with the Duffing Oscillator, the first step will be to choose a design for $F(\alpha, \theta)$. Because (6.4) is a linear system, the deadbeat observer design can be computed easily, so both design options are available. For the sinusoid problem, the continuation was executed by slowly reducing the gain of the deadbeat observer with good results. Unfortunately, for the two-mode problem, this technique leads to a region where two

of the eigenvalues are outside the unit circle and the system is unstable. Thus, the simple method used for the sinusoid problem is unsuitable for this problem. Of course, there are many other ways to design and parameterize an observer. A useful way to think about the design options is to plot - see Fig. 6.9 - the eigenvalues of $A-LC$ as α moves from OE to PE. For OE, $L = 0$ and the eigenvalues are simply the eigenvalues of A. For PE, L is a deadbeat observer which means that $A-LC$ is nilpotent and all of its eigenvalues are at the origin. The design choice is the path that the eigenvalues follow between these points. The red curve in Fig. 6.9 shows the path followed using the αL_0 method. As mentioned above, a portion of the curve is outside the unit circle (black curve) and unstable. The blue curve is another potential path - a straight line connecting the two endpoints.

One could certainly imagine other observer design methods as well. The goal is for the sequence of global minima to be continuous; it is not obvious how to design an observer that achieves it. Further, many design methods such as pole placement (which could be used to produce the blue curve) are numerical in nature and therefore do not offer a way to analytically calculate the derivatives of L required for the IFT. Finite differences are the only option in such a case. A handful of tests along these lines showed that, while convergence to the OE global minimum often occurred, there were points along the global minimum curve where it changed very rapidly and continuity may have been violated. The IFT checks using finite differences were inconclusive because of numerical problems. These results emphasize the point that Theorem 3.0.1 represents sufficient, not necessary conditions for convergence. A brave user may try any continuation that comes to mind and it may converge (although without guarantee).

The above discussion makes clear that while the continuation method may still succeed for the observer design method associated with this problem, there may

be no obvious way to draw solid conclusions from the IFT. On the other hand, the derivatives associated with the exponential weighting method can always be calculated using the methods of Ch. 4 for any linear, discrete-time system. Simulation results for this method are discussed in the following section.

Simulation Results

The parameters for the underlying system were described in the previous section. The only remaining parameters are associated with $F(\alpha)$. For the results that follow, $\alpha_0 = 1$ and $n_0 = 8$. Again, the first important test is to compare the OE cost function to the cost function at α_0 which should improve convexity significantly. Figs. 6.10 and 6.11 show the one-dimensional versions of the OE and $\alpha = 1$ (noise-free) cost functions respectively. Because the cost function is actually four-dimensional, it cannot be visualized in its entirety. The one-dimensional cost functions do not show the full story, but they give a rough idea of the relative convexity of each set of cost functions. While not perfectly convex, the $\alpha = 1$ cost function certainly seems to represent an improvement over OE. Assuming that the user was able to globally minimize the $\alpha = 1$ cost function (not as certain a result as in the sinusoid and Duffing problems), establishing continuity of the global minimum sequence is all that remains. As discussed above, for any discrete-time system using the exponential weighting method, analytical expressions for the derivatives are available. In this case, the user might as well proceed directly to solving the ODE (4.12) in the manner first described in Ch. 5 rather than first invoking the NLLS solver at a number of steps and checking the invertibility of $D_\theta(\alpha, \bar{\theta})$ at each step to guarantee continuity. Solving the ODE directly has two main advantages: first, it is an implicit check of continuity because the solver will fail if the solution sequence is not continuous;

second, the user need not choose the number of steps to take because many ODE solver routines can be configured in variable-step mode where the step size depends on the local characteristics of the ODE.

Fig. 6.12 shows the ODE solutions for each parameter as α varies along with the correct parameter values (dotted line). Because the solver succeeded, continuity is guaranteed. Note that (6.4) has two equally valid OE global minima because the same system output would result if the position of the second order blocks were swapped (i.e., if θ_1 and θ_2 were swapped with θ_3 and θ_4, respectively). Should the solver make this exchange, it should not be considered a failure. This swap has taken place in Fig. 6.12. The two-mode problem presents some unique difficulties. Because the eigenvalues are near the unit circle, even small steps can result in instability. Further, the exponential weighting method results in an initial cost function that is not globally convex. To minimize this cost function (and initialize the ODE solver), the user may need an NLLS routine that is more powerful than Gauss-Newton with step-size or restart heuristics. Alternatively, the user could restart the Gauss-Newton routine manually should it fail. Matlab's built-in *lsqnonlin* (release R2014a) solver with the 'Trust-Region Reflective' algorithm proved capable of handling the $\alpha = 1$ cost function for the above simulation conditions.

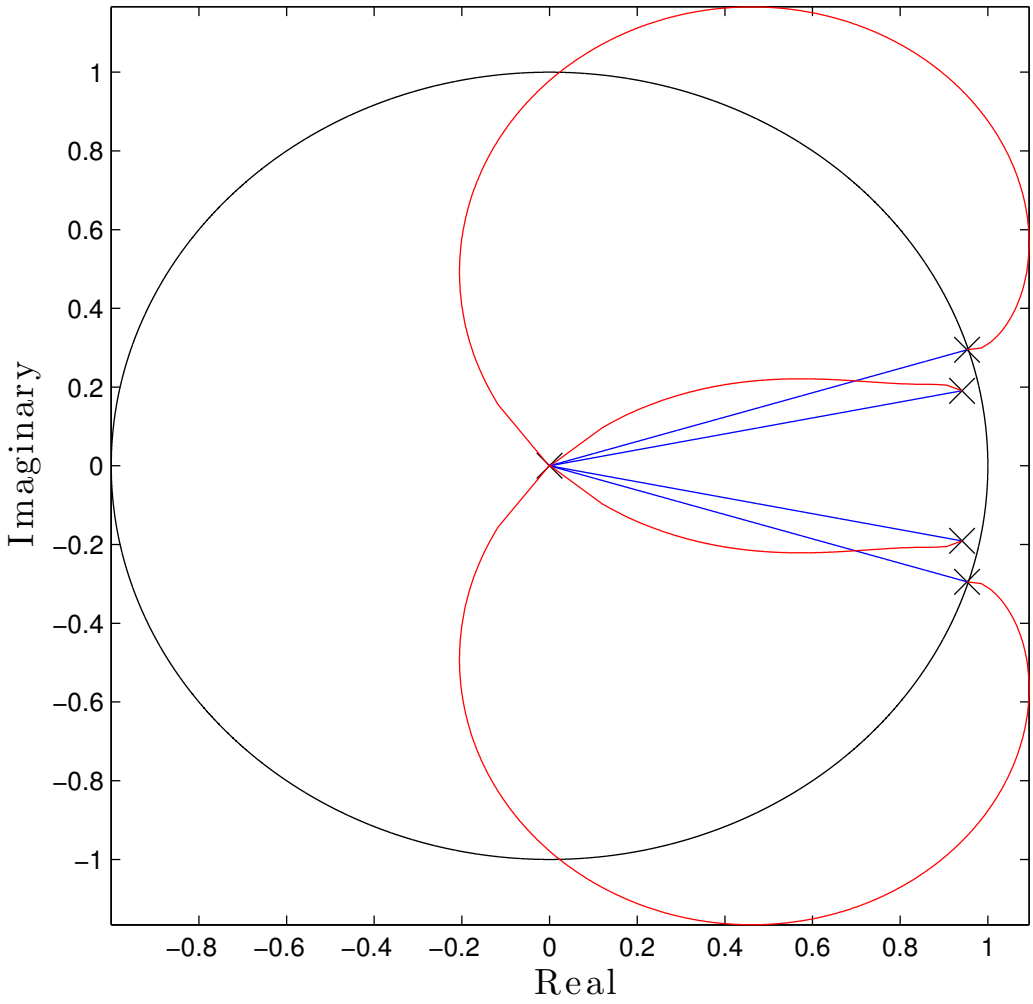

Figure 6.9: Eigenvalues of $A - LC$. Points marked with 'X' are the eigenvalues of A (the OE simulation system) and the origin (the PE simulation system). The blue curve follows a straight line from OE to PE, the red curve is the result of slowly reducing the gain of the deadbeat observer, and the black curve is the unit circle.

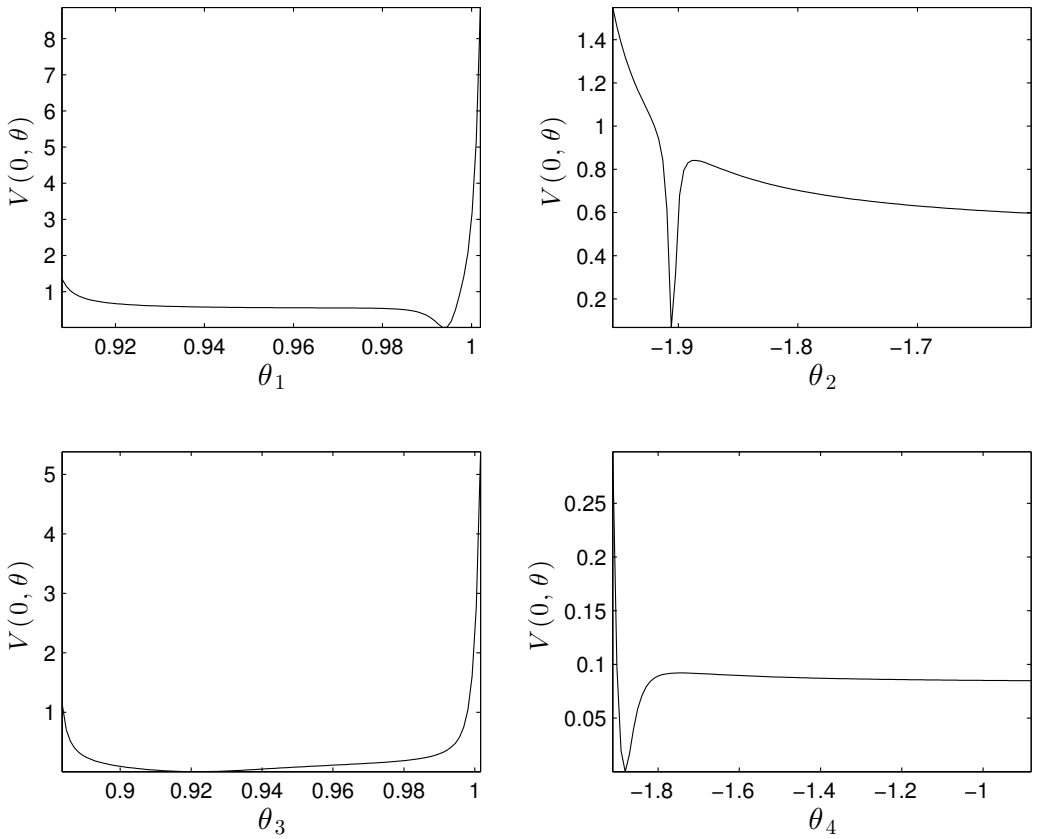

Figure 6.10: One-dimensional OE cost functions for each parameter. Most of the ranges shown are at the stability limits and each cost function rapidly approaches infinity outside these ranges.

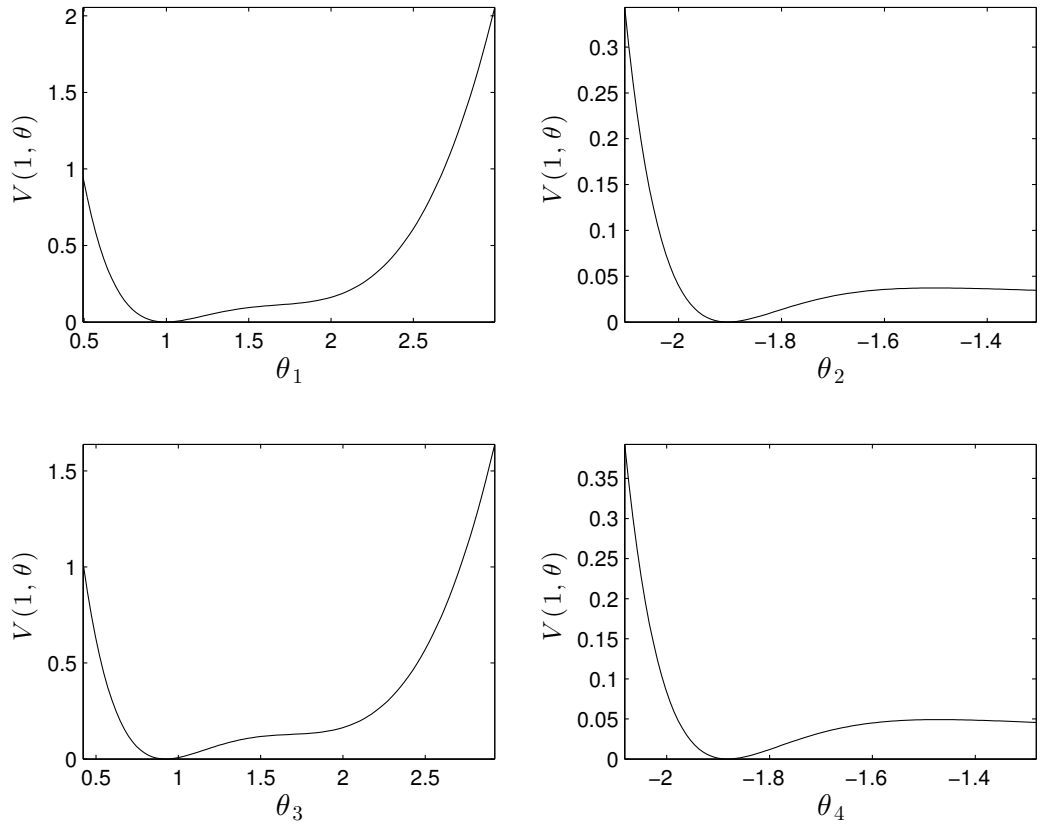

Figure 6.11: One-dimensional cost functions at $\alpha = 1$ for each parameter. Notice that the convexity is an improvement over OE, but there are certainly initial guesses for which the global minimum will not be found.

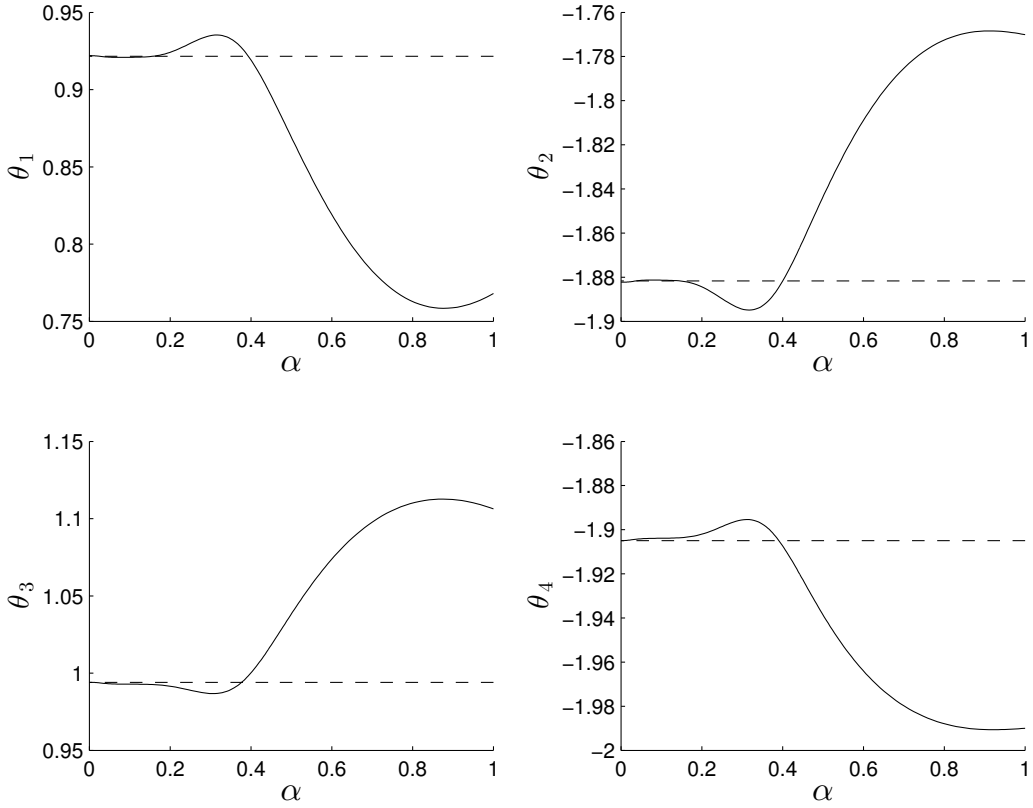

Figure 6.12: Sequence of global minimizing parameters as α varies. Dotted lines are the correct parameter values.

CONCLUSION

Mathematical modeling is fundamental to science and engineering. Often, prior information about the physical system under test is available in the form of physically parameterized equations describing the behavior of the phenomena, initial guesses for the parameters, and information about the noise associated with any measurements that might be taken. System identification is the process of refining the initial model using experimental data.

In the present work, models are confined to dynamic systems (in state-space form) with specific parameterizations. The process of system identification consists of simulating the model and adjusting the parameters such that the squared error between the simulation output and a sequence of noisy measurements is minimized. This procedure goes by the name output error (OE) minimization in the system identification literature and is known to produce good parameter estimates when the measurement noise is zero-mean white and the model is an accurate representation of the underlying system. The main difficulty with OE methods is that the cost function (sum of squared error) is a nonlinear function of the parameters; therefore, its minimization generally requires an iterative routine. Most iterative routines, such as Gauss-Newton, start from an initial guess, work on local information and, at best, converge to a local minimum. For convergence to the global minimum to occur, the initial guess must be within its basin of attraction.

While many other methods of producing good models exist, OE methods remain popular despite the difficulty in computing the optimal parameter estimates. The reason for their enduring popularity is twofold: first, as described above, they produce good estimates under reasonable conditions; second, the process of matching the

output of a simulation model to measurements is conceptually simple and appealing and only requires a simulation model to proceed.

In fact, OE methods are used often enough, and the minimization of OE-type cost functions difficult enough that Ljung, one of the founders of modern system identification, has recently described the problem of minimizing nonlinear cost functions as an important open problem in the field [29]. The present work focuses on this problem and describes a family of methods that involve a continuation from a more convex cost function to the non-convex OE cost function.

Review of Methods

The present work begins by describing (Ch. 1) the one-step prediction-error (PE) approach. For linear systems, the PE approach leads to a linear system of equations that can easily be solved for the unknown parameters without an iterative routine. In most cases, the resulting parameters have a noise-related bias. The first key observation of this work is that the PE approach can also be shown to result from a deadbeat observer for the measured outputs using the OE simulation model. By continuously varying the gain of this observer, residual sequences can be produced which exhibit some of the properties of both OE and PE. It is argued, via the example of sinusoid frequency estimation (Ch. 2), that, beginning by solving for the PE parameters and slowly varying the observer gain towards OE, a sequence of minimizations can be carried out using the previous answer as the next initial guess that will converge to the OE global minimum. If the steps in the sequence are small enough, none of the minimizations are difficult.

Next, it is shown (Ch. 3) that the PE residuals, along with all of the intermediate residuals produced by the observer, can be viewed as a particular

linear transformation of the OE residuals. That is, all of the residuals in the continuation described above can be written in the form $R(\alpha, \theta) = F(\alpha, \theta) R_{OE}(\theta)$ where θ is the vector of model parameters, α is the variable that parameterizes the continuation, and $F(\alpha, \theta)$ is the matrix that executes the linear transformation. The observer method results in a particular structure for $F(\alpha, \theta)$, but when viewed as a general linear transformation, it is clear that there is a great variety of potential transformations beyond the observer method. The obvious next question is, what are the characteristics of the transformation that will guarantee convergence to the OE global minimum by solving a sequence of minimization problems as α varies from some α_0 to α_{oe}?

In an attempt to answer this question, it is first observed that the stationary point of the Gauss-Newton algorithm (and other nonlinear least-squares (NLLS) routines) offers an implicit definition for the globally minimizing parameter vector as a function of α, denoted $\bar{\theta}(\alpha)$. Making use of the Implicit Function Theorem (IFT), sufficient conditions for guaranteed convergence to the OE global minimum are provided in the form of Theorem 3.0.1. These conditions are: 1) the cost function at the beginning of the continuation $V(\alpha_0, \theta)$ must be sufficiently convex so that the NLLS routine finds the global minimum; 2) the cost function at the end of the continuation $V(\alpha_{oe}, \theta)$ must be the OE cost function which means that $F(\alpha_{oe}, \theta)$ must be the identity matrix; 3) the sequence of global minima $\bar{\theta}(\alpha)$ must be continuous. If these conditions are met, convergence to the global minimum is guaranteed. However, these conditions do not provide direct design requirements for $F(\alpha, \theta)$. Other considerations are needed to suggest designs for $F(\alpha, \theta)$. Once a design is chosen, the conditions of the IFT can be checked at each point along the continuation to provide a *post hoc* proof that the OE global minimum has been found.

Checking the conditions of the IFT requires the computation of certain derivatives. Ch. 4 shows exactly what derivatives are required and analytical expressions that can be used to compute them efficiently (at least for discrete-time systems). Interestingly, it is also observed that, once these expressions are known, the IFT implies that $\frac{d\bar{\theta}(\alpha)}{d\alpha}$ is also known. That is, a (nonlinear) ordinary differential equation (ODE) is available that describes the behavior of the global minimum as α varies. If the NLLS solver is invoked once at α_0 to initialize the ODE, the OE global minimum can be potentially found by applying a standard numerical ODE solver. When this technique is applied to the ODE associated with the sinusoid frequency estimation problem (Ch. 5), the OE global minimum appears to be an attractor for the ODE from a wide range of initialization points, suggesting the surprising possibility that, provided the user has even a poor initial guess, the OE global minimum can be found without ever invoking a NLLS solver.

The remainder of this work consists of successfully applying the theoretical methods described above to the sinusoid frequency estimation problem, the continuous-time Duffing Oscillator (a system with nonlinear dynamics), and a discrete-time system whose output is the sum of two underdamped modes. An alternative to the observer-based design for $F(\alpha, \theta)$ is developed and successfully applied to each of these problems. Various observations about the practical aspects of the method are made.

Future Work

The present work suggests several lines along which further research may be conducted:

- The IFT implies the existence of a nonlinear ODE that describes the behavior of the global minimum along the continuation. It has been observed that the OE global minimum is an attractor for a range of initial conditions. It is possible that tools from nonlinear dynamics might be of use in better understanding the behavior of this ODE. An ideal result from the application of these tools would be suggestions for useful $F(\alpha, \theta)$ designs. For instance, if the attraction regions were better understood, it may be possible to design $F(\alpha, \theta)$ such that they are enlarged.

- The theoretical developments of the work are sound and the handful of practical applications successful. In order for the method to be broadly useful for users without technical expertise, a general purpose software package would be necessary. A partial list of developments required for such a package includes: automated routines for computing the analytical derivatives of Ch. 4, robust routines for computing the derivatives using finite differences when the analytical expressions are inappropriate, a wider variety of designs for $F(\alpha, \theta)$ (possibly drawn from developments described in the first bullet) and an understanding of the classes of systems for which each design is best, heuristics for selecting the meta-parameters such as n_0 and α_0 in the exponential weighting method, and heuristics for selecting the step size in α.

- The Duffing Oscillator example showed the application of the methods to a continuous-time system. It involved sampling the output of the system and applying the transformation matrix $F(\alpha, \theta)$ to these samples. An alternative method would be to apply the transformation in continuous-time before sampling. The continuous-time analog of the linear transformation executed by $F(\alpha, \theta)$ is the integral transformation $R(t, \alpha, \theta) = \int f(t, \tau, \alpha, \theta) R_{OE}(\tau, \theta) d\tau$.

It is possible that such an approach could yield both valuable theoretical insights and better practical performance for continuous-time systems.

REFERENCES CITED

[1] A. Betser and E. Zeheb. Modified output error identification-elimination of the spr condition. *Automatic Control, IEEE Transactions on*, 40(1):190–193, Jan 1995.

[2] P. E. Caines. Stationary linear and nonlinear system identification and predictor set completeness. *Automatic Control, IEEE Transactions on*, 23(4):583–594, 1978.

[3] G. O.-P. B. Castillo-Toledo and A. Loukianov. A globally convergent estimator for n-frequencies. *IEEE Trans. Autom. Control*, 47(5):857 –863, may 2002.

[4] T. F. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on optimization*, 6(2):418–445, 1996.

[5] L. Davis et al. *Handbook of genetic algorithms*, volume 115. Van Nostrand Reinhold New York, 1991.

[6] J. E. Dennis Jr and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall, 1983.

[7] M. Dorigo and C. Blum. Ant colony optimization theory: A survey. *Theoretical computer science*, 344(2):243–278, 2005.

[8] R. Eberhart and Y. Shi. Particle swarm optimization: developments, applications and resources. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 81–86 vol. 1, 2001.

[9] W. Fromm, A. Halinka, and W. Winkler. Accurate measurement of wide-range power system frequency changes for generator protection. In *Developments in Power System Protection, Sixth International Conference on (Conf. Publ. No. 434)*, pages 53–57, Mar 1997.

[10] N. A. Gershenfeld. *The nature of mathematical modeling*. Cambridge university press, 1999.

[11] N. Giaquinto and A. Trotta. Fast and accurate adc testing via an enhanced sine wave fitting algorithm. *IEEE Trans. Instrum. Meas.*, 46(4):1020 –1025, aug 1997.

[12] F. Glover. Tabu search-part i. *ORSA Journal on computing*, 1(3):190–206, 1989.

[13] P. Handel. Properties of the ieee-std-1057 four-parameter sine wave fit algorithm. *Instrumentation and Measurement, IEEE Transactions on*, 49(6):1189–1193, Dec 2000.

[14] D. Hart, D. Novosel, Y. Hu, B. Smith, and M. Egolf. A new frequency tracking and phasor estimation algorithm for generator protection. *Power Delivery, IEEE Transactions on*, 12(3):1064–1073, Jul 1997.

[15] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge university press, 2012.

[16] L. Hsu, R. Ortega, and G. Damm. A globally convergent frequency estimator. *IEEE Trans. Autom. Control*, 44(4):698 –713, apr 1999.

[17] C.-M. Huang, C.-J. Huang, and M.-L. Wang. A particle swarm optimization to identifying the armax model for short-term load forecasting. *Power Systems, IEEE Transactions on*, 20(2):1126–1133, 2005.

[18] I.-K. Jeong and J.-J. Lee. Adaptive simulated annealing genetic algorithm for system identification. *Engineering Applications of Artificial Intelligence*, 9(5):523–532, 1996.

[19] P. V. Kabaila. On output-error methods for system identification. *Automatic Control, IEEE Transactions on*, 28(1):12–23, 1983.

[20] N. Karaboga, A. Kalinli, and D. Karaboga. Designing digital iir filters using ant colony optimisation algorithm. *Engineering Applications of Artificial Intelligence*, 17(3):301–309, 2004.

[21] S. Kay and J. Marple, S.L. Spectrum analysis - a modern perspective. *Proceedings of the IEEE*, 69(11):1380–1419, Nov 1981.

[22] J. Kenney and C. Rohrs. The composite regressor algorithm for iir adaptive systems. *Signal Processing, IEEE Transactions on*, 41(2):617–628, Feb 1993.

[23] I. Kovacic and M. J. Brennan. *The Duffing equation: nonlinear oscillators and their behaviour*. John Wiley & Sons, 2011.

[24] K. Kristinsson and G. Dumont. Genetic algorithms in system identification. In *Intelligent Control, 1988. Proceedings., IEEE International Symposium on*, pages 597–602. IEEE, 1988.

[25] J.-N. Lin and R. Unbehauen. Bias-remedy least mean square equation error algorithm for iir parameter recursive estimation. *Signal Processing, IEEE Transactions on*, 40(1):62–69, Jan 1992.

[26] L. Ljung. Convergence analysis of parametric identification methods. *Automatic Control, IEEE Transactions on*, 23(5):770–783, 1978.

[27] L. Ljung. *MATLAB: System Identification Toolbox: User's Guide Version 4*. The Mathworks, 1995.

[28] L. Ljung. *System Identification: Theory for the User*. Prentice Hall PTR, Upper Saddle River, NJ, 1999.

[29] L. Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010.

[30] L. Ljung and P. E. Caines. Asymptotic normality of prediction error estimators for approximate system models. *Stochastics*, 3(1-4):29–46, 1980.

[31] C. Lyzell, M. Enqvist, and L. Ljung. Handling certain structure information in subspace identification. 2009.

[32] R. Marino and P. Tomei. Global estimation of n unknown frequencies. *IEEE Trans. Autom. Control*, 47(8):1324 – 1328, aug 2002.

[33] J. J. Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer, 1978.

[34] D. Norris and L. Snyder. Consistency of least-squares estimates used in linear systems identification. *SIAM Journal on Control*, 13(6):1183–1193, 1975.

[35] M. H. Protter, C. B. Morrey, et al. *A first course in real analysis*. Springer Science & Business Media, 2012.

[36] D. Puangdownreong, K. Areerak, A. Srikaew, S. Sujitjorn, and P. Totarong. System identification via adaptive tabu search. In *Proc. IEEE Int. Conf. on Industrial Technology (ICIT'02)*, volume 2, pages 915–920, 2002.

[37] P. Regalia. An improved lattice-based adaptive iir notch filter. *IEEE Trans. Signal Process.*, 39(9):2124–2128, 1991.

[38] P. Regalia. *Adaptive IIR filtering in signal processing and control*. CRC Press, 1994.

[39] P. Regalia et al. An unbiased equation error identifier and reduced-order approximations. *Signal Processing, IEEE Transactions on*, 42(6):1397–1412, 1994.

[40] S. R. Shaw, M. Keppler, and S. B. Leeb. Pre-estimation for better initial guesses. *Instrumentation and Measurement, IEEE Transactions on*, 53(3):762–769, 2004.

[41] S. R. Shaw and C. R. Laughman. A method for nonlinear least squares with structured residuals. *IEEE transactions on automatic control*, 51(10):1704, 2006.

[42] T. Söderström and P. Stoica. Comparison of some instrumental variable methods—consistency and accuracy aspects. *Automatica*, 17(1):101–115, 1981.

[43] T. Söderström and P. Stoica. Some properties of the output error method. *Automatica*, 18(1):93–99, 1982.

[44] T. Söderström and P. Stoica. *System identification*. Prentice-Hall, Inc., 1988.

[45] J. Stewart. *Calculus: early transcendentals*. Cengage Learning, 2015.

[46] P. Stoica, R. Moses, B. Friedlander, and T. Soderstrom. Maximum likelihood estimation of the parameters of multiple sinusoids from noisy measurements. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(3):378–392, Mar 1989.

[47] P. Stoica and A. Nehorai. Statistical analysis of two nonlinear least-squares estimators of sine-wave parameters in the colored-noise case. *Circuits, Systems and Signal Processing*, 8(1):3–15, 1989.

[48] P. Stoica and T. Soderstrom. The steiglitz-mcbride identification algorithm revisited–convergence analysis and accuracy aspects. *Automatic Control, IEEE Transactions on*, 26(3):712–717, 1981.

[49] P. Stoica and T. Söderström. Optimal instrumental-variable methods for identification of multivariable linear systems. *Automatica*, 19(4):425–429, 1983.

[50] K. Tan, Y. Li, D. Murray-Smith, and K. Sharman. System identification and linearisation using genetic algorithms with simulated annealing. 1995.

[51] H. Toliyat, E. Levi, M. Raina, et al. A review of rfo induction motor parameter estimation techniques. *Energy conversion, IEEE Transactions on*, 18(2):271–283, 2003.

[52] D. Trudnowski, D. Kosterev, and J. Undrill. Pdci damping control analysis for the western north american power system. In *Power and Energy Society General Meeting (PES), 2013 IEEE*, pages 1–5. IEEE, 2013.

[53] D. Trudnowski, D. Kosterev, and J. Wold. Open-loop pdci probing tests for the western north american power system. In *PES General Meeting— Conference & Exposition, 2014 IEEE*, pages 1–5. IEEE, 2014.

[54] D. Tufts and R. Kumaresan. Estimation of frequencies of multiple sinusoids: Making linear prediction perform like maximum likelihood. *Proceedings of the IEEE*, 70(9):975–989, Sept 1982.

[55] P. J. Van Laarhoven and E. H. Aarts. *Simulated annealing*. Springer, 1987.

[56] P. Van Overschee and B. De Moor. N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994.

[57] M. Verhaegen. Subspace model identification part 3. analysis of the ordinary output-error state-space model identification algorithm. *International Journal of control*, 58(3):555–586, 1993.

[58] M. Verhaegen and P. Dewilde. Subspace model identification part 1. the output-error state-space model identification class of algorithms. *International journal of control*, 56(5):1187–1210, 1992.

[59] M. Verhaegen and P. Dewilde. Subspace model identification part 2. analysis of the elementary output-error state-space model identification algorithm. *International journal of control*, 56(5):1211–1241, 1992.

[60] M. Verhaegen and V. Verdult. *Filtering and system identification: a least squares approach*. Cambridge university press, 2007.

[61] J. Zhang and et al. Sinewave fit algorithm based on total least-squares method with application to adc effective bits measurement. *IEEE Trans. Instrum. Meas.*, 46(4):1026 –1030, aug 1997.

[62] Z. Zhong, C. Xu, B. Billian, L. Zhang, S. Tsai, R. Conners, V. Centeno, A. Phadke, and Y. Liu. Power system frequency monitoring network (fnet) implementation. *Power Systems, IEEE Transactions on*, 20(4):1914–1921, Nov 2005.