Power System Identification Toolbox

Phase Two Progress

D. J. Trudnowski

August 1994

Prepared for
the Bonneville Power Administration
under a Related Services Agreement
with the U.S. Department of Energy
Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Richland, Washington 99352

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Executive Summary

Pacific Northwest Laboratory (PNL), under contract to the Bonneville Power Administration (BPA), is developing production-grade software tools that identify dynamic models from measured and simulated power system response. Technical foundation and collaboration for the project includes many years of ongoing research and development at the University of Colorado and Montana State University in theory, and at the BPA in applications, as well as PNL research over the last three years.

The project is being conducted as a three-phase effort. The first phase has been completed and involved scoping current techniques. Phase Two includes extending existing tools, developing new tools, and demonstrating and developing applications. The final phase will focus on organizing software into a modular package and developing user manuals with instruction and application guidelines. This report describes progress to date on Phase Two, which is approximately 50% complete.

The needs of the power industry for specialized tools of this type are particularly acute. The power industry is currently pushing to operate electrical transmission systems much closer to theoretical limits by using real-time control systems. Maintaining reliable electrical service under these stressed conditions requires extensive engineering analysis capability. As has been demonstrated in the engineering research literature, the tools being developed under this project provide fundamental enabling technology in developing this analysis capability.

Continued work on this project will focus on completing Phases Two and Three as milestones. Assuming that current support for the project is maintained, it is expected that Phase Two will be completed at the end of 1994 and Phase Three at the end of 1995. At the end of Phase Three, available products will include a formalized version of the codes and a set of manuals describing proper use, general application guidelines, and specific power system application guidelines.

Technical Summary

This report describes current progress on a project funded by the Bonneville Power Administration (BPA) to develop a set of state-of-the-art analysis software (termed the Power System Identification [PSI] Toolbox) for fitting dynamic models to measured data. The project is being conducted as a three-phase effort. The first phase, completed in late 1992, involved investigating the characteristics of the analysis techniques by evaluating existing software and developing guidelines for best use. Phase Two includes extending current software, developing new analysis algorithms and software, and demonstrating and developing applications. The final phase will focus on reorganizing the software into a modular collection of documented computer programs and developing user manuals with instruction and application guidelines. Phase Two is approximately 50% complete; progress to date and a vision for the final product of the PSI Toolbox are described.

The needs of the power industry for specialized system identification methods are particularly acute. The industry is currently pushing to operate transmission systems much closer to theoretical limits by using real-time, large-scale control systems to dictate power flows and maintain dynamic stability. Reliably maintaining stability requires extensive system-dynamic modeling and analysis capability, including measurement-based methods. To serve this need, the BPA has developed specialized system-identification computer codes through in-house efforts and university contract research over the last several years. To make full integrated use of the codes, as well as other techniques, the BPA has commissioned Pacific Northwest Laboratory (PNL) to further develop the codes and techniques into the PSI Toolbox. These tools are being developed to solve the following fundamental problems: 1) identify linear models of systems with multiple modes of oscillation; and 2) identify specific parameters of a dynamic system. As has been demonstrated in the engineering research literature, solving these fundamental problems allows one to analyze and understand several power-system-dynamic problems. Examples include control system tuning and commissioning tests; power system stabilizer tuning; fault-response studies to determine system dynamic characteristics; torsional oscillation and subsynchronous investigations; modal analysis of transient simulation data; and identification of high-frequency models to analyze problems such as ferroresonance in transformers. The PSI Toolbox will provide enabling technology for solving many of these power-system-dynamics problems.

The mathematical and engineering literature is rich with techniques for system identification, each with its own specific application and advantage. Methods included in the PSI Toolbox are those that meet specific challenges encountered with power system dynamics, such as noisy data, very high-order systems, oscillatory dynamics, and nonlinearities. Three basic analysis techniques are contained in the Toolbox: Prony analysis of time-domain data; linear transfer-function identification through frequency-domain fitting; and structured-model identification through frequency-domain fitting. By combining these methods with standard signal analysis techniques (such as the discrete Fourier transform [DFT]), one can perform many types of experiments and analyses.

Phase Two results described in this report are focused on developing a vision for the final product of the PSI Toolbox; developing analysis code; and developing application guidelines. The final form of the Toolbox will be very flexible so that it can be easily used as the basis for future application-specific packages. It will consist of four primary parts: 1) advanced standard-FORTRAN analysis routines; 2) codes to interface the analysis routines with application-specific industry and commercial software; 3) custom user interface codes; and 4) documentation such as

user manuals, general application guidelines, and specific guidelines for solving power system problems. By far, the primary focus of the project consists of developing the FORTRAN analysis routines and developing guidelines for their proper use.

Code development described in this report has involved both extending existing software and writing new routines. Existing Prony analysis software has been expanded to include applications to transfer-function identification, multi-signal analysis, and error-bound estimates of identified parameters. A new code was written to directly identify a linear transfer function from a system's frequency response. When comparing this new code with conducting a Prony analysis of the inverse discrete Fourier transform (IDFT) of the frequency response, the Prony analysis approach provided more accurate results for all cases except when only a few data points are available.

Application guidelines were developed for using pseudo-random and swept-sine probing signals for identifying modal parameters for electromechanical oscillation and subsynchronous resonance problems. This was done by conducting several simulation and analysis experiments with varying parameters. System responses to these probing signals were analyzed using DFT methods to estimate the system's frequency response. Model parameters were then estimated using IDFT and Prony analysis or direct linear frequency-domain fitting. In general, it was found that accurate model identification can be obtained at much lower signal-to-noise (SNR) ratios than with pulse probing. The tradeoff is that considerably more data is required to identify an accurate model (several minutes versus tens of seconds for electromechanical dynamics). The SNR advantage could prove to be very valuable as this allows one to identify system parameters with very weak probing signals that disturb the system only slightly.

Continued development of the PSI Toolbox will focus on completing Phases Two and Three as milestones. Assuming that current support for the project is maintained, it is projected that Phase Two will be completed at the end of 1994 and Phase Three at the end of 1995. At the end of Phase Three, available products will include a formalized version of the codes and a set of manuals describing proper use, general application guidelines, and specific power system application guidelines.

Over the next year, Phase Two tasks will focus on

- continued code development and refinement to include new algorithms and to make existing codes more robust, efficient, and flexible
- testing and evaluating new algorithms and codes
- continued application guideline development - Major effort will be placed on guidelines for analysis of transient-simulation and measured data. Other efforts include coordinated use between Prony analysis and structured frequency-domain fitting.

Specific objectives and tasks will be formed for Phase Three as Phase Two is concluded. These tasks will focus on reorganizing the software into a modular collection of documented computer programs and developing user manuals with instruction and application guidelines.

Acknowledgments

The author wishes to thank Dr. John Hauer of the Bonneville Power Administration for guiding and supporting this work. Professor Louis Scharf of the University of Colorado has provided insightful technical guidance related to Prony analysis and signal analysis theory. Also, the author thanks Dr. Matthew Donnelly of Pacific Northwest Laboratory for many fruitful technical discussions. And finally, graduate student Mr. Jeff Johnson of Washington State University is thanked for conducting several simulations, analysis, and code writing; and graduate student Mr. Brian Mauer of the University of Colorado is thanked for computer-coding analysis algorithms.

Contents

Executive Summary	iii
Technical Summary	v
Acknowledgments	vii
1.0 Introduction	1.1
2.0 System Identification.....	2.1
2.1 Pulse Inputs	2.2
2.2 Pseudo-Random Inputs	2.2
2.2.1 Estimating $G(j\omega)$	2.3
2.2.2 Parameter Identification	2.5
2.3 Swept-Sine Inputs.....	2.6
2.3.1 Chirp Sweep.....	2.6
2.3.2 Discrete Sweep	2.7
2.4 Identifying Specific Parameters	2.7
3.0 The PSI Toolbox Vision	3.1
4.0 Phase Two Progress	4.1
4.1 Phase Two Tasks	4.1
4.1.1 Coding Guidelines	4.1
4.1.2 PRSPAK Updates	4.3
4.1.3 Develop FREQID	4.3
4.1.4 Code Interfaces to MATLAB and LabVIEW.....	4.3
4.1.5 Develop "Local Intelligence" Routines	4.4
4.2 User Support	4.5
5.0 Experience with Pseudo-Random and Swept-Sine Probing.....	5.1
5.1 Experiment Descriptions and Test Systems.....	5.1
5.2 Pseudo-Random Probing	5.5
5.2.1 Periodogram Averaging	5.6

5.2.2 Pseudo-Random Probing Versus Pulse Probing	5.6
5.2.3 Band-Limited Identification	5.11
5.2.4 Incorporating More Data	5.13
5.3 Chirp Probing	5.15
5.4 Discrete Sweep	5.15
5.4.1 Discrete Sweep Versus Pseudo-Random and Chirp	5.18
5.4.2 FREQID Versus Prony	5.19
5.5 Comparing Approaches	5.20
6.0 Conclusions and Future Tasks	6.1
6.1 Conclusions	6.1
6.2 Future Tasks.....	6.2
7.0 References.....	7.1
Appendix - PSI Toolbox Copyholders	A.1

Figures

2.1	General System Model	2.1
3.1	PSI Toolbox Architecture.....	3.1
5.1	General Test System Diagram.....	5.1
5.2	Frequency Response of Test System 1	5.3
5.3	Frequency Response of Test System 2	5.4
5.4	Frequency Response of Test System 3	5.4
5.5	Estimating Test System 3 Response with 13 dB SNR Due to n_p and 33 dB SNR Due to n_m , $N = 4096$, and $M = 512$	5.7
5.6	Coherence for Test System 3 Response with 13 dB SNR Due to n_p and 33 dB SNR Due to n_m , $N = 4096$, and $M = 512$	5.7
5.7	Estimating Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 512$	5.8
5.8	Coherence for Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 512$	5.8
5.9	Estimating Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 2048$	5.9
5.10	Coherence for Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 2048$	5.9
5.11	Comparison of Pulse Probing with Random Probing for Test System 1.....	5.10
5.12	Test System 1 Response to Pulse Probing and Random Probing	5.10
5.13	Test System 2 Output for Probing with at 0 dB SNR Versus Noise-Only Output	5.12
5.14	Frequency Response of Test System 2 for $N = 32000$ and $N = 1312023$ Versus the Actual Response	5.12
5.15	Coherence for Test System 2 for $N = 32000$ and $N = 1312023$ Versus the Actual Response	5.13

5.16	Model and Periodogram Frequency Response Versus Actual System Response for Test System 3, $N = 32000$	5.14
5.17	Coherence Function for Test System 1, Chirp Probing Versus Pseudo-Random Probing	5.16
5.18	Frequency Response Estimates for Test System 1, Chirp Probing Versus Pseudo-Random Probing	5.16
5.19	Discrete-Sweep-Input Frequency Response for Steps of 0.1 Hz.....	5.18

Tables

5.1	Test System 1	5.2
5.2	Test System 2	5.2
5.3	Test System 3	5.3
5.4	Identified Poles for Test System 1, Pulse Probing Versus Pseudo-Random Probing.....	5.11
5.5	Identified Residues for Test System 1, Pulse Probing Versus Pseudo-Random Probing	5.11
5.6	Identified Poles for Test System 2, $N = 32000$ Versus $N = 131023$	5.13
5.7	Identified Poles for Test System 3, $N = 5950$ Versus $N = 32000$	5.14
5.8	Identified Poles for Test System 1, Chirp Probing Versus Pseudo-Random Probing	5.17
5.9	Identified Sources for Test System 1, Chirp Probing Versus Pseudo-Random Probing	5.17
5.10	Identified Model for Discrete-Sweep Input for Steps of 0.1 Hz.....	5.19
5.11	Identified Model for Discrete-Sweep Input for Steps of 0.05 Hz.....	5.19
5.12	Identified Poles for Test System 3, FREQID Versus Prony for a Few Data Points	5.19

1.0 Introduction

Many dynamic systems encountered in engineering and science exhibit dynamic responses. Examples are endless but include electromechanical oscillations in large power systems, flexible mechanical structures, long-reach robotic manipulators, and many aircraft control problems. Although the underlying dynamics of such systems are often nonlinear, it is often possible and very useful to represent the system operating at a point with a linear model. The purposes of such models include modal analysis, control system design, development of more-detailed computer models, and monitoring. Such linear models can be extracted using two basic approaches: model the system using theoretical derivations and linearize about a given operating condition; or use mathematical fitting techniques to optimally fit a linear model to specified system response data. This second approach is often termed system identification. Both approaches have unique advantages that make them very complementary. Described in this report is a set of state-of-the-art analysis software currently being developed for fitting dynamic models to measured data.

The needs of the power industry for specialized system-identification methods are particularly acute. The power industry is currently pushing to operate transmission systems much closer to theoretical limits by using real-time, large-scale control systems to dictate power flows and maintain dynamic stability. Reliably maintaining stability requires extensive system-dynamic modeling and analysis capability, including measurement-based methods. To serve this need, the Bonneville Power Administration (BPA) has developed specialized system-identification computer codes through in-house efforts and university contract research over the last several years. To make full integrated use of the codes, as well as other techniques, the BPA has commissioned Pacific Northwest Laboratory (PNL) to further develop the codes and techniques into a software package termed the Power System Identification (PSI) Toolbox. These tools are being developed to solve the following fundamental problems: 1) identify linear models of systems with multiple modes of oscillation; and 2) identify specific parameters of a dynamic system. Algorithms focus on conditions often encountered in analyzing field or laboratory data (such as noise and nonlinearities). The PSI Toolbox will provide enabling technology for solving many power-system-dynamics problems.

Several applications of system identification techniques in solving power system dynamic problems are documented in the literature. Examples include control system tuning and commissioning tests; power system stabilizer tuning (Trudnowski et al. 1991); studying fault-response data to determine interarea electromechanical dynamic characteristics (Hauer 1989b); investigating torsional oscillation and subsynchronous effects (Donnelly et al. 1993); modal analysis of transient simulation data (Kamwa et al. 1993); and identifying high-frequency models to analyze problems such as ferroresonance in transformers (Soysal and Semlyen 1992). The PSI Toolbox will provide fundamental analysis software for analyzing and solving problems in these classes.

Because many of the tools being developed under this project can be applied to dynamic systems in general, applications can extend outside power system problems. One technology that could benefit is the design of long-reach robotic manipulators. Such technology is currently being developed for remediation of hazardous waste sites at U.S. Department of Energy facilities. In designing long-reach manipulators, the system's position and movement must be controlled for successful implementation. The development of adequate control systems requires extensive modeling and analysis both in the laboratory and mathematically. The software tools being developed in this project hold considerable potential for solving many of the problems encountered

in designing advanced control systems. An example of how these techniques have already been applied to a prototype robotic system is described in Trudnowski, Baker, and Evans (1993).

PSI Toolbox development is being conducted in a three-phase process. The first phase has involved investigating the characteristics of the analysis techniques by evaluating existing software and developing guidelines for best use. This effort has been completed and is described in Trudnowski (1992a, b). Phase Two includes extending current software, developing new analysis algorithms and software, and demonstrating and developing applications. The final phase will focus on reorganizing the software into a modular collection of documented computer programs and developing user manuals with instruction and application guidelines.

This report describes progress made on the first half of Phase Two during 1993. Results described in this report are focused on developing a vision for the final product of the PSI Toolbox; analysis code development; and developing application guidelines. The final form of the Toolbox will be very flexible so that it can be easily used as the basis for future application-specific packages. It will consist of four main parts: 1) advanced standard-FORTRAN analysis routines; 2) codes to interface the analysis routines with application-specific industry and commercial software; 3) custom user interface codes; and 4) documentation such as user manuals, general application guidelines, and specific guidelines for solving power system problems. By far, the primary focus of the project is on developing the FORTRAN analysis routines and developing guidelines for their proper use.

Code development described in this report has involved both extending existing software and writing new routines. Existing Prony analysis software has been expanded to include applications to transfer-function identification, multi-signal analysis, and error-bound estimates of identified parameters. A new code was written to directly identify a linear transfer function from a system's frequency response. When comparing this new code with conducting a Prony analysis of the inverse discrete Fourier transform (IDFT) of the frequency response, the Prony analysis approach provided more accurate results for all cases except when only a few data points are available.

Application guidelines were developed for using pseudo-random and swept-sine probing signals for identifying modal parameters for electromechanical oscillation and subsynchronous resonance problems. This was done by conducting several simulation and analysis experiments with varying parameters. System responses to these probing signals were analyzed using DFT methods to estimate the system's frequency response. Model parameters were then estimated using IDFT and Prony analysis or direct linear frequency domain fitting. In general, it was found that accurate model identification can be obtained at much lower single-to-noise ratios (SNR) than with pulse probing. The tradeoff is that considerably more data is required to identify an accurate model (several minutes versus tens of seconds for electromechanical dynamics). The SNR advantage could prove to be very valuable as this allows one to identify system parameters with very weak probing signals that disturb the system only slightly.

Continued development of the PSI Toolbox will focus on completing Phases Two and Three as milestones. Assuming that support for the project is maintained, it is projected that Phase Two will be completed at the end of 1994, and Phase Three will be completed at the end of 1995. At the end of Phase Three, available products will include a formalized version of the codes and a set of manuals describing proper use, general application guidelines, and specific power system application guidelines.

In Section 2 of this report, the problem of system identification is introduced. Section 3 provides an overview of the final PSI Toolbox product. Section 4 describes the current status of tasks being conducted under Phase Two. Experiments and results related to identifying models through pseudo-random and swept-sine probing are provided in Section 5, and final conclusions and descriptions of future work are summarized in Section 6.

2.0 System Identification

The mathematical and engineering literature is rich with techniques for system identification, each with its specific application and advantage. Methods included in the PSI Toolbox are those that meet specific challenges encountered with power system dynamics, such as noisy data, very high-order systems, oscillatory dynamics, and nonlinearities. This section provides an introduction to the general system identification approaches that best meet the challenges of power system problems.

Consider the general dynamic system in Figure 2.1. Input $u(t)$ is a known probing signal at time t used to excite the system; $n_p(t)$ is unknown process noise that also excites the system; $n_m(t)$ is unknown measurement noise; and $y(t)$ is the measured signal. The process noise represents the many random factors that constantly excite the system under ambient conditions. In a power system, these include varying load and generation effects, system switching effects, and small disturbances. n_m represents the noise associated with measurement devices. The system is assumed to be time invariant (or at least slowly varying) and of finite dimension, although the dimension can be very large. In most cases, the system is nonlinear, but it is assumed that it behaves in a relatively linear manner about its current operating point. These assumptions fit well for a wide class of engineering systems including power systems. In the most general case, u , n_p , n_m , and y are vectors allowing for multiple inputs and outputs.

Figure 2.1. General System Model

In most cases, measurement noise can usually be mitigated through filtering and instrumentation refinements. Process noise, on the other hand, is more difficult to handle because it produces energy in the same bandwidth that is to be identified. For most analysis techniques, n_p and n_m are assumed to be stationary processes.

The objective is to identify parameters that characterize the system by analyzing finite records of y and possibly u . Depending on the particular application, different situations can occur. In some cases, u is a pulse or step function that creates a free response at y . At other times, one may apply a function to u such as a pseudo-random signal or a swept-sine wave. Also, the objective of the identification can vary. One may be interested in identifying a linear transfer function model for the system, determining the modal content of the system, or identifying specific parameters within the system (such as time constants or inertia coefficients). In the following sections, several of these cases are considered.

2.1 Pulse Inputs

In many cases, it is easiest to simply apply a pulse function at u , for example, a step or series of step functions. In power systems, this type input often naturally occurs as the result of breakers disconnecting system elements such as a transmission line. If u is known, then one can analyze y to fit a linear transfer function to the system. If u is unknown, the modal content of the system can be estimated (i.e., an estimate to the characteristic equation of the system).

This problem has been thoroughly studied, with Prony analysis (Scharf 1991) proving to be a very valuable tool. The properties of Prony analysis for identifying linear impulse-response models are thoroughly investigated in Trudnowski (1992a), and comparisons are made with a standard least-squares approach in Trudnowski, Donnelly, and Hauer (1993). These and other studies have shown that Prony analysis results in accurate model identification for high-order oscillatory systems under noisy conditions. Applications for general transfer function identification is demonstrated by Pierre, Trudnowski, and Hauer (1992). Algorithms for estimating model order are discussed by Trudnowski (1993b). Applications to power systems have been demonstrated throughout the literature; for example, see Hauer (1989b).

2.2 Pseudo-Random Inputs

The objective here is to apply a stationary low-level pseudo-random probing signal at u , measure the response y , and estimate the linearized system transfer function. This problem can be approached from two perspectives. The first involves analyzing relatively short data segments of u and y and directly fitting a model to this data. This approach has been studied extensively in the literature (e.g., see Soderstrom and Stocica 1989). A few available commercial software packages follow this approach. One that seems to perform very well is ADAPTX (by Adaptics, Inc.). The other approach is to analyze considerably larger segments of u and y and use spectral estimation methods to approximate the frequency response of the system. A transfer function is then estimated by fitting to this frequency response. Because spectral estimation can significantly reduce the detrimental effects of the noise, this second approach results in more accurate transfer function estimates for cases where the effects of n_p and n_m are a problem (such as in power systems). Because the literature is rich with information concerning the first approach, only the second is discussed here.

The identification procedure is designed to reduce the effects of process and measurement noise by using a pseudo-random excitation at u and spectral estimation. This allows for accurate model identification at low signal-to-noise ratios (SNRs). The identification procedure is conducted using the following steps:

1. Apply u to excite the frequency range of interest and measure y . Using u and y , estimate the system's frequency-response transfer function $G(j\omega)$ at frequencies ω and call this estimate $G_e(j\omega)$.
2. Fit a parametric model to $G_e(j\omega)$. The parametric model will be the transfer function.

2.2.1 Estimating $G(j\omega)$

In estimating $G(j\omega)$, a few definitions are required: $U(j\omega)$ is the Fourier transform of u ; $S_{uy}(\omega)$ is the cross spectral density between u and y ; and $S_{uu}(\omega)$ is the auto spectral density function for u . Using standard probability theory (Bendat and Piersol 1980), we know that

$$S_{uy}(\omega) = E\{U^*(j\omega)Y(j\omega)\} \quad (2.1)$$

$$S_{uu}(\omega) = E\{U^*(j\omega)U(j\omega)\} \quad (2.2)$$

where $E\{\cdot\}$ represents the expectation operator.

Let $G(s)$ represent the linearized Laplace-domain transfer function of the system in Figure 2.1. Also, assume that u is uncorrelated with n_p and n_m , and assume that u , n_p , and n_m are stationary (this is a safe assumption because the user has complete control over u). Then the frequency response of $G(s)$ can be obtained from Bendat and Piersol (1980)

$$G(j\omega) = \frac{S_{uy}(\omega)}{S_{uu}(\omega)} \quad (2.3)$$

Various noise effects generally preclude exact calculation of S_{uy} and S_{uu} from finite-length records. However, practical estimates can often be formed by using the well established *periodogram averaging* or *Welch* method (Oppenheim and Schaffer 1975). Let u_k be a sampled version of $u(t)$ with sample period T and $k = 0, 1, \dots, N-1$. It is processed as K subsets (or *processing blocks*) $u^{(i)}$, each of length M . These are permitted to overlap, often by as much as 90% or more. Then the estimate S_{uu}^e for S_{uu} is formed as

$$S_{uu}^e(\omega) = \frac{T}{WK} \sum_{i=1}^K \left\{ \left| \text{DFT}(w_M u^{(i)}) \right|^2 \right\} \quad (2.4a)$$

$$W = \frac{1}{M} \sum_{k=0}^{M-1} w_M^2(k) \quad (2.4b)$$

where w_M is some windowing function (such as a Hanning window (Oppenheim and Schaffer 1975)), and DFT indicates the discrete Fourier transformation. Calculation of the cross spectral estimate S_{uy}^e proceeds similarly. The estimated frequency response of the transfer function is now available as

$$G_e(j\omega) = \frac{S_{uy}^e(\omega)}{S_{uu}^e(\omega)} \quad (2.5)$$

The quality and efficiency of the processing itself can be directly adjusted through the parameters N and M , the amount of block overlap, and the choice of window w_M . In many circumstances, the signal must be preconditioned before processing starts. This may involve filtering, trend removal, record padding, and, in special cases, the processing of outliers or discontinuities. Such operations are important, but their influence is indirect and not considered here.

The main objective here is to adjust the processing, together with the characteristics of the probing signal u , so that $G_e(j\omega)$ accurately represents $G(j\omega)$ across the frequency range of interest. This requires some indicator for the quality of $G_e(j\omega)$. One such indicator is available through the (squared) coherency function, defined in Bendat and Piersol (1980) as

$$\gamma_{uy}^2(\omega) = \frac{|S_{uy}(\omega)|^2}{S_{uu}(\omega)S_{yy}(\omega)} \quad (2.6)$$

Using Figure 2.1, the following identities result:

$$S_{uy}(\omega) = G(j\omega)S_{uu}(\omega) \quad (2.7)$$

$$S_{yy}(\omega) = |G(j\omega)|^2 S_{uu}(\omega) + S_{zz}(\omega) \quad (2.8a)$$

$$S_{zz}(\omega) = |G(j\omega)|^2 S_{n_p n_p}(\omega) + 2\text{Re}\{G(j\omega)S_{n_p n_m}(\omega)\} + S_{n_m n_m}(\omega) \quad (2.8b)$$

where $\text{Re}\{\cdot\}$ is the real part of $\{\cdot\}$. S_{zz} represents activity in the output signal that is statistically orthogonal to the probing signal. Using Equations (2.7) through (2.8b), the coherency function is

$$\gamma_{uy}^2(\omega) = \frac{|G(j\omega)|^2 S_{uu}^2(\omega)}{|G(j\omega)|^2 S_{uu}^2(\omega) + S_{zz}(\omega)S_{uu}(\omega)} \quad (2.9)$$

The function ranges from 0 to 1 in value and can be thought of as a normalized measure of the degree to which u and y "vary together" at frequency ω . This leads to the interpretation of γ_{uy} as a correlation coefficient associated with the projection of y onto a basis defined by the response

function G to u . In most cases, one should expect the coherency function to be peaked at peaks in $G(j\omega)$ and to fall when $G(j\omega)$ is small.

Implications of the (squared) coherency *estimate*

$$\hat{\gamma}_{uy}^2(\omega) = \frac{|S_{uy}^e(\omega)|^2}{S_{uu}^e(\omega)S_{yy}^e(\omega)} \quad (2.10)$$

are significantly different from those of the true coherency function. This estimate is biased, with unity value for averaging of just one periodogram. Its behavior as more periodograms are accumulated into the average contains important information that can be used to adjust the test procedure and signal processing, assess the estimation quality, and perhaps obtain some initial indications of dominant modal frequencies. Two important properties include the following:

1. If the coherency estimate falls over a range where the transfer function estimate is peaked or notched, then the estimated transfer function will likely have a biased error. This indicates that the frequency resolution may be poor and the window size M should be increased. This may also be an indication of nonlinearities in the system.
2. If the coherency estimate takes on a random form and falls broadly over a given frequency range, then the estimated transfer function will likely have random errors associated with it because of process or measurement noise effects. This problem can be mitigated if more data is used and/or the magnitude of u is increased.

Suppose, for example, that probing is being done with a rather weak broadband signal and that the periodogram parameters have been adjusted to account for the above properties so that the coherency estimate has settled into a pattern of deep nulls alternating with pronounced peaks. To the extent that this represents the true coherency function, it is apparent that the SNR is very poor near the nulls and that protracted averaging (or stronger probing) may be needed for accurate results there. However, because the processing seems to have successfully distinguished the orthogonal noise from that linearly related to u , it may well be that enough averages have been accumulated and an accurate transfer-function estimate obtained. An abrupt change of the coherency estimate from its settled configuration would probably indicate a disturbance of the system under test and that processing should be restarted.

2.2.2 Parameter Identification

Given $G_e(j\omega)$, a parametric model is identified using one of two approaches. With the first, an inverse discrete Fourier transformation (IDFT) operation is used to estimate the impulse response of $G(s)$ in the frequency range of interest, and Prony analysis is used to fit a model. With the second approach, the fitting is applied directly to $G_e(j\omega)$.

The inverse Fourier transform of $G_e(j\omega)$ is the impulse response of $G_e(s)$ and is estimated by

$$g_e(mT) = \text{IDFT} \begin{bmatrix} G_{e,0} & G_{e,1} & \cdots & G_{e,N-1} & \text{Re}(G_{e,N-1}) & G_{e,1}^* & \cdots & G_{e,N-2}^* \end{bmatrix}, m = 0, 1, \dots, 2N-1 \quad (2.11)$$

where T is the sample period for u and y , $G_{e,k} = G_e(j2\pi k/(NT))$, and $\text{Re}(\bullet)$ is the real part of $(\bullet)$. Conducting a Prony fit to g_e results in an estimate of the transfer function (Trudnowski 1992a). In applying pseudo-random probing and swept-sine inputs, it is possible to excite a specified frequency range of interest. Samples of $G_e(j\omega)$ outside this range are not accurate; therefore, they are ignored in Equation (2.7) by assuming they are zero.

Several methods have been proposed for fitting a parametric model to the frequency response of a system. An extensive literature review is contained in Trudnowski (1992b). One method used in the PSI Toolbox involves fitting a transfer-function model to the frequency-response data in a least-squares sense (Johnson and Trudnowski 1993). With the approach, the estimated frequency response is determined by the function

$$G_e(j\omega) = \frac{j^0 b_0 T_0(\hat{\omega}) + j^1 b_1 T_1(\hat{\omega}) + \dots + j^m b_m T_m(\hat{\omega})}{j^0 a_0 T_0(\hat{\omega}) + j^1 a_1 T_1(\hat{\omega}) + \dots + j^n a_n T_n(\hat{\omega})} \quad (2.12)$$

where T_i is the i th Chebyshev polynomial, $\hat{\omega}$ is a normalized frequency vector, and a_i and b_i are the unknown coefficients. $G(j\omega)$ is substituted for $G_e(j\omega)$, and Equation (2.12) is solved for the unknown coefficients using a linear least-squares approach. The formulation of Equation (2.12) is used because it results in a numerically well-conditioned problem (see Johnson and Trudnowski 1993). After the fitting, a standard transfer function form is obtained through a known linear transformation matrix involving the Chebyshev polynomial coefficients.

2.3 Swept-Sine Inputs

One can also apply a swept-sine function at u (a sinusoidal function with a changing frequency). This approach is similar to pseudo-random probing, except the probing function is no longer stationary. As with other types of inputs, one can directly fit a model to u and y , but this approach requires a large SNR. As with pseudo-random probing, if larger segments of u and y can be analyzed, it is better to use spectral estimation methods to approximate the frequency response of the system and estimate the transfer function by fitting to the frequency response.

Because a swept-sine function is not stationary, periodogram averaging cannot be applied as with pseudo-random probing to estimate the frequency response. Alternatively, two approaches are possible. One involves applying a sine function at u with a linearly changing frequency (this is termed a *chirp sweep*). The other requires that a set of sine functions be separately applied, each with its own frequency (this is termed a *discrete sweep*).

2.3.1 Chirp Sweep

With a chirp sweep, the input is a sinusoidal function but with a frequency that linearly changes from an initial value to a final value so as to cover the desired range (Franklin, Powell, and Workman 1990). The system's frequency response is estimated by dividing the DFT of u by the DFT of y . The effects of noise can be reduced by conducting the experiment several times and averaging together spectral estimates before dividing.

If more than one experiment is conducted, one uses calculations similar to those used for random probing. In this case, the spectral density is estimated by

$$S_{uy}^e(\omega) = \frac{T}{WK} \sum_{i=1}^K \left\{ \text{DFT}(w_M u)^* \text{DFT}(w_M y^{(i)}) \right\} \quad (2.13)$$

Here, u is the chirp signal, $y^{(i)}$ is the output of the system for the i th experiment, K is the number of experiments, and w_M is the windowing function. No overlap is allowed between experiments, and the same input u must be applied. For $K=1$, the coherence function will be unity, but will have no meaning. If more than one case is used, then the coherence function as defined in Equation (2.6) can be used to judge the quality of the frequency response estimate.

2.3.2 Discrete Sweep

With this approach, the frequency response of the system is estimated one frequency at a time. A known fixed-frequency sinusoidal function is applied at u for several cycles. The least-squared-error estimate of the frequency response for that fixed frequency is calculated using (Franklin, Powell, and Workman 1990)

$$G_e(j\omega) = \frac{2}{AN} \sum_{k=0}^{N-1} y(kT) [\sin(\omega kT) + j \cos(\omega kT)] \quad (2.14)$$

where N is the number of time samples, T is the sample period, $j = \sqrt{-1}$, ω is the frequency, and the input is

$$u(kT) = A \sin(\omega kT) \quad (2.15)$$

N is chosen so that an integer number of cycles of ω is used. The process is repeated for each frequency of interest.

2.4 Identifying Specific Parameters

All the previously described methods focused on identifying a transfer function. Another problem often encountered is that of identifying specific parameters of a system. In this case, the identified model must preserve the structure of the model. Power system examples include identifying the reactances of a generator or the excitation system time constants. One approach to solving this problem is to estimate the frequency response of the system using the methods described in Sections 2.2 and 2.3, and then fit a structured model to the frequency response. In most cases, the structure of the system model is such that the fitting problem is not linear; the problem is therefore very difficult to solve. One solution approach is to use a gradient-based search method to fit the parameters. This approach is discussed in detail in Trudnowski (1992b).

3.0 The PSI Toolbox Vision

The PSI Toolbox consists of a variety of routines. When used together, they allow one to fit a linear model to a system's response using one of three approaches:

1. Analyze the system's response to a step, square-wave pulse, or a combination of steps and pulses to obtain a transfer function estimate or the modal content of the system.
2. Analyze the system's response to a pseudo-random noise input to obtain a transfer-function estimate of the system or to identify specific model parameters of the system.
3. Analyze the system's response to a swept-frequency input to obtain a transfer-function estimate of the system or to identify specific model parameters of the system.

The final form of the PSI Toolbox will be very flexible so that it can be easily used as the basis for future application-specific packages. Figure 3.1 depicts its currently envisioned form. It will consist of four primary parts: 1) advanced standard-FORTRAN analysis routines; 2) codes to interface the analysis routines with application-specific industry and commercial software; 3) custom user interface codes; and 4) documentation such as user manuals, general application guidelines, and specific guidelines for solving power system problems. By far, the primary focus of the project consists of developing the FORTRAN analysis routines and developing guidelines for their proper use. These routines are highly specialized codes that, when used with standard DFT analysis, encompass the three analysis approaches described above.

Figure 3.1. PSI Toolbox Architecture

The development model for the FORTRAN routines is the well known LAPACK (formerly LINPACK) routines developed through Argonne National Laboratory (Anderson et al. 1992). LAPACK is basically an extensive library of computer routines for solving mathematical matrix problems. Similarly, this project will develop a collection of routines for identifying dynamic models from measured dynamic system response data.

Interfaces to industry and commercial codes are being developed to simplify the application of these techniques to simulation programs (such as transient simulation), and because many secondary analysis and data manipulation functions are available in high-quality commercial software packages. The second reason allows one to use the FORTRAN routines in conjunction with commercial packages. As an example, recent work at PNL and BPA involved implementing Prony analysis FORTRAN codes into the data acquisition program LabVIEW (by National Instruments, Inc.) for use in a new BPA Macintosh-based field monitoring system. Using LabVIEW, custom graphics and data interfacing specific for Prony analysis and the monitoring problem were made possible.

The analysis codes are being written in FORTRAN for several reasons. FORTRAN is a well established language that has industry standards so that porting codes to different computer platforms is relatively easy. It is very useful in coding analysis and "number crunching" algorithms. Also, the original codes developed at BPA and the University of Colorado were written in FORTRAN. The public domain mathematical package LAPACK was developed in FORTRAN (many of the PSI routines use LAPACK).

In 1993, considerable work was performed to improve existing PSI routines and develop new routines. Altogether, the PSI Toolbox includes approximately 100 subroutines; because of the large volume of code, a complete listing of these routines is not contained in this report. Copies of current versions can be obtained from the author. The following list provides a description of each major FORTRAN routine now included in the PSI Toolbox and its current status.

1. PRSPAK - This subroutine conducts a Prony analysis. An older version of the routine is described in Trudnowski (1992a). The latest version of the program conducts fits to multiple signals. When used with other routines, this allows one to identify a single-input, multi-output (SIMO) transfer functions. Development of PRSPAK has been conducted as a collaborative effort with the University of Colorado.
2. FREQID - This subroutine identifies a transfer function by fitting to a system's frequency response. Algorithms used include those described in Johnson and Trudnowski (1993).
3. ORDER - A subroutine for determining the order of an identified model, it is based on the algorithm described in Trudnowski (1993). Order is used in combination with PRSPAK or FREQID.
4. TRANFUN - This subroutine allows one to employ Prony analysis for transfer-function identification based on the algorithm described in Pierre, Trudnowski, and Hauer (1992). It is used in combination with PRSPAK.
5. IPRONYID and SIGPAKZ - These are main programs that call the subroutines described above. IPRONYID is an interactive program, and SIGPAKZ is a batch program.
6. BOUND - This subroutine estimates error bounds for identified parameters from Prony

analysis. It is based on the algorithm in McWhorter and Scharf (1993). The current version was written at the University of Colorado through a student internship funded from this project.

7. interfaces to commercial codes - Several routines have been written to interface the analysis routines to commercial packages, including MATLAB (by the Mathworks, Inc.) and LabVIEW. Each major subroutine has an interfacing program.
8. SYSFIT - An extensive program that fits a structured model to the frequency response of a system, SYSFIT is described in Hauer (1974, 1977, 1982, 1989a), Hauer and Taylor (1990), and Trudnowski (1992b). Considerable flexibility is allowed in building the model, including feedback paths and multipath models. SYSFIT uses a general-purpose gradient projection program called MORPAK II (Hauer 1974) to find the optimal model.

4.0 Phase Two Progress

The PSI Toolbox is being developed in phases. The first phase involved assessment of current tools and identification of needed improvements. This phase was primarily completed in 1992, with results described in Trudnowski (1992a, 1992b). Phase Two focuses on improving existing tools, developing new tools, and developing best-use guidelines. We are currently in the middle of this phase. Phase Three will involve organizing the software and guidelines into a formal documented package.

From the results of Phase One, the objectives of Phase Two were formed. These needs include the following:

- Overhaul and update current analysis codes to include new theoretical developments that improve their performance and make them more flexible.
- Develop new analysis codes that complement existing codes and fulfill objectives of the PSI Toolbox.
- Implement a system to maintain the code versions and ensure that standards are followed in code development.
- Develop "local intelligence" in analysis codes to expedite use for non-experts.
- Develop general best-use guidelines through theoretical analysis, experimentation, and demonstration.

Other activities also covered under Phase Two relate to providing support to parties interested in using PSI software.

4.1 Phase Two Tasks

Based on these objectives, the following Phase Two main tasks were developed: 1) implement coding development guidelines and version controls; 2) update and refine the PRSPAK subroutine; 3) develop the FREQID routine; 4) update and refine SYSFIT and supporting routines; 5) develop initial interfaces of Toolbox codes to MATLAB and LabVIEW; 6) develop "local intelligence" in identification codes through modifications and support routines; 7) test, evaluate, and demonstrate application of Toolbox routines for transfer-function identification from pseudo-random probing and swept-sinusoidal probing; 8) demonstrate application of PSI Toolbox routines in analysis of transient simulation data; and 9) demonstrate application of PSI Toolbox routines in analysis of field measured data. Tasks 1, 2, 3, 5, and 7 have been completed or partially completed. Progress on tasks 1, 2, 3, and 5 is described here. Task 7, which involved considerable effort, is described in Section 5.

4.1.1 Coding Guidelines

To maintain quality control over the developed computer codes, guidelines for code development are required. Because scientific coding encompasses such a wide variety of issues, few formal guidelines have been developed for analysis code development (it is difficult to cover

all possible situations). One objective of the FORTRAN routines being developed for the Toolbox is that they be very modular (e.g., portable between many different computer platforms). Another objective is that the codes can be easily incorporated into other analysis codes. Based on these objectives, and several discussions with PNL and BPA staff, the following FORTRAN guidelines were developed:

- Use standard FORTRAN 77.
- Place all array dimensions in PARAMETER statements and INCLUDE files. This allows one to easily change array dimensions.
- Always pass variables to subroutines through arguments, not COMMON blocks. This avoids confusion that can occur with codes that consist of many subroutines.
- Use no EQUIVALENCING. The array sizes required in routines for the Toolbox are not large enough to justify the added confusion that EQUIVALENCING can add.
- Perform all input/output (including reading/writing data files, graphics calls, etc.) at the highest level possible in subroutine chains. This is because many of these functions are platform-dependent, which makes porting difficult.
- If input/output is required in an analysis subroutine, do it through standardized input/output subroutine calls. This makes porting easier because only the input/output subroutine needs modification.
- Do not use tabs. Each compiler may recognize tabs differently.
- Descriptively name all variables. Use IMPLICIT NONE in all routines, and clearly define all variables.
- Minimize memory and CPU requirements through efficient coding.
- Document each subroutine with both headers and comments within the code.

Because the Toolbox consists of many subroutines and several version of each, a system must be employed to track the modifications made to each routine. The system implemented this past year is the Revision Control System (RCS), a widely used public domain system available under UNIX operating systems (Tichy 1985). RCS manages revisions of text documents, particularly source programs, documentation, and data. It automates the storing, retrieval, logging, and identification of revisions, and it provides selection mechanisms for composing configurations. A version of a source file is assigned a hierarchical version number, and RCS tracks all changes according to the version numbers.

A four-digit number scheme is used for the Toolbox. Each source file is given a number *a.b.c.d*. Numbers *a.b.c* track modifications made to the file for updating analysis code; *d* tracks changes required for porting to a new computer platform. The default for *d* is *d* = 1; this implies it can be compiled on a UNIX platform. If a version with *d* = 2 exists, then this version is required to compile the file on a DOS platform; if a *d* = 2 version does not exist, then the *d* = 1 version is used (no platform-dependent modifications are required). Similarly, *d* = 3 is used for a VMS platform, and *d* = 4 is used for a MACINTOSH platform.

4.1.2 PRSPAK Updates

Considerable work has been devoted this past year to updating PRSPAK. This work has been a coordinated effort with the University of Colorado. The following major modifications have been implemented:

- General code cleanup using the guidelines in Section 4.1.1 removed all direct input/output in PRSPAK and made it totally platform-independent.
- Modifications to improve code efficiency and robustness include extended checking on algorithm controls and removing the fast linear-prediction routines (because they added little speed-up [polynomial rooting is much more computationally expensive] and the codes required considerable modifications to make them portable).
- LAPACK has replaced LINPACK calls and some custom subroutines developed at the University of Colorado. LAPACK "expert drivers" are used to solve singular value decomposition (SVD) and QR-based problems. These drivers provide the latest options in improved numerical techniques, such as matrix scaling.
- An option was implemented to recalculate residues from existing poles.
- PRSPAK can now identify common modes for multiple signals, permitting single-input, multi-output identification. This was implemented by extending the data vector for the linear prediction calculation.

4.1.3 Develop FREQID

As described earlier, one approach to identifying a system's transfer function is to directly fit a model to its frequency response. Using the algorithm in Johnson and Trudnowski (1993), the subroutine FREQID was developed this past year. The structure for FREQID is very similar to that of PRSPAK and the same guidelines were used.

Several tests were conducted with FREQID. This subroutine has proved to be less numerically robust than conducting Prony analysis on the IDFT of the frequency response although, in cases where very few data points are available, FREQID provides better results. Also, FREQID does not require equally spaced data (such as the IDFT approach).

4.1.4 Code Interfaces to MATLAB and LabVIEW

As mentioned in Section 3, several of the codes being developed here will be implemented into other commercial codes through interfaces. Early development, testing, and field use of the PSI Toolbox has required that interfaces be developed for MATLAB and LabVIEW. Development of these interfaces also falls under the scope of other PNL and BPA projects. Consequently, much of this work has been conducted through the funding of these projects.

The MATLAB routines were implemented through FORTRAN-MEX, routines which are FORTRAN routines that can be called by MATLAB as a function. To date, PRSPAK, FREQID, TRANFUN, and ORDER have been implemented.

PSI FORTRAN codes have been incorporated into the LabVIEW programming environment in two ways -- as a code interface node (CIN) and as an Apple Event. A CIN is a LabVIEW structure that allows line code in the traditional programming languages to be used in a LabVIEW virtual instrument (VI). The advantages of this approach are a consistent user interface and a compact application. The user interface is consistent because the FORTRAN algorithm becomes another self-contained VI within LabVIEW to be used in much the same manner as any of the other signal processing VIs. In fact, most of the LabVIEW signal processing VIs are CINs. The CIN approach is compact in that the FORTRAN analysis is performed entirely in LabVIEW. No other applications need to be used. The primary disadvantages of the CIN approach are software maintenance and "single-tasking." Major modifications often must be made to the FORTRAN codes to create a CIN, and the CIN must be rebuilt for each new version of LabVIEW. This makes it difficult to maintain the CIN through new releases of LabVIEW. Single-tasking refers to the mode in which LabVIEW executes CINs. Normally, LabVIEW VIs execute in parallel, but CINs must execute to completion before LabVIEW allows any other VI to time-share the Macintosh CPU. This becomes a problem with PSI algorithms because a single analysis may take tens of minutes to execute.

Apple Events are Macintosh operating system features that allow Mac applications to communicate with one another. In this situation, the FORTRAN program is a separate application. LabVIEW and the FORTRAN code both execute on the Macintosh at the same time, and data is passed between the two using Apple Events. Advantages for CINs become disadvantages for Apple Events and vice versa. The user experience can be made to be the same for both approaches except that, with Apple Events, LabVIEW continues to perform concurrent tasks while the FORTRAN analysis is performed.

To date, PRSPAK, FREQID, TRANFUN, and ORDER have been implemented into LabVIEW. Experience indicates that the Apple-Events approach is the best because of its ease of maintenance and multitasking ability. In either case, LabVIEW provides an easy-to-understand graphical presentation of PSI results that is a great benefit to the user.

4.1.5 Develop "Local Intelligence" Routines

Successful use of the system identification tools requires pre- and post-analysis. Examples include determining the model order, calculating transfer function residues from the input function and the output residues, estimating error bounds on the identified parameters, and determining the best options for algorithm controls. Each of these areas has been investigated this past year, and routines have been developed.

TRANFUN is employed to calculate the transfer function terms after a Prony analysis. The equations employed in this routine are described in Pierre, Trudnowski, and Hauer (1992) and Trudnowski, Donnelly, and Hauer (1993).

Another recently developed subroutine is ORDER. It is used to determine which terms from a Prony analysis represent actual modes and which represent extraneous modes. The algorithm, its applications, and characteristics are described in Trudnowski (1993).

Estimating the error of identified parameters is currently an active area of research at the University of Colorado (UC). Through collaboration with UC, the subroutine BOUND has recently been developed. This routine uses the Cramer-Rao bound algorithm in McWhorter and Scharf (1993) to estimate a parameter bound.

4.2 User Support

Several individuals from the electric utility industry have requested a copy of software in the PSI Toolbox. Persons to whom copies have been distributed are listed in the Appendix. Support to these individuals has included mailing source code and documentation notes, general liaison support, and minor code modifications. Little attempt has been made to develop an extensive user group at this time. This is primarily because the codes are being modified and tested in Phase Two. As the project moves into Phase Three, more active user participation will be pursued.

5.0 Experience with Pseudo-Random and Swept-Sine Probing

As described in Section 2, several approaches are possible for identifying a transfer function model for a system. Two of these are probing the system with a pseudo-random signal and with a swept-sine wave. The theory behind applying these approaches is described in Sections 3.3 and 3.4. In this section, several results on test systems that demonstrate the properties of these approaches are described. The approaches are compared with each other and with a pulse input approach evaluated in Trudnowski (1992a). In the processes of investigating the properties, several experiments were conducted with varying parameters. The results presented here show only the experiments that demonstrate conclusive points.

5.1 Experiment Descriptions and Test Systems

The experiments can be described by considering the diagram in Figure 5.1, which is similar to Figure 2.1. Noise is added at n_p to represent process noise and at n_m to represent measurement noise. A pseudo-random or swept-sine function is added at u . Signals u and y are analyzed as described in Section 2 to estimate a transfer function or the modal content of the test system. Three test systems are used, each with unique characteristics.

Figure 5.1. General Test System Diagram

The linearized system is represented in parallel Laplace-domain form as

$$G(s) = \sum_{i=1}^n \frac{R_i}{s - p_i} \quad (5.1)$$

where R_i and p_i are the i th residue and pole, and R_0 is the feedforward term. Using linear system theory, (5.1) can be rewritten in many other forms such as pole/zero or state-space. The model is identified as (5.1) because it is a numerically robust formulation.

The first test system (described in Table 5.1 in a parallel form) is an 11th-order linearized model of a synchronous generator in a 17-machine computer-simulated power system. The model represents the transfer function from the reference voltage to the accelerating power of the generator. The model contains five damped oscillatory modes and one highly-damped pole. As

indicated by the residues and the system's frequency response in Figure 5.2, the two modes near 1.8 Hz and 1.5 Hz dominate the response and are the result of the local mode of the generator and a neighboring generator. The modes near 0.7 Hz and 0.36 Hz are interarea modes that are not nearly as strong and, therefore, are more difficult to identify.

Test System 2 represents the linearized transfer function of a computer-simulated three-generator system used to study turbine-generator shaft oscillations. This model is taken from Donnelly, et al. (1993), and represents the linearized response between a dynamic brake connected to the terminals of one of the generators and the generator speed deviation signal. Table 5.2 shows the parameters for this model, and Figure 5.3 shows the frequency response. A central challenge of this system is the two closely-spaced modes near 21 Hz and 19 Hz.

The last test system is shown in Table 5.3 and Figure 5.4. It is taken from Hauer (1989b) and was developed by analyzing electromechanical response of the western North American power system to a disturbance. The frequency content of this 23rd-order system is similar to that of Test System 1. Its response is dominated by the 0.43-Hz and 0.33-Hz modes. The other modes are considerably weaker; therefore, they are much more difficult to identify.

The noise signals n_p and n_m are simulated using the computer-generated Gaussian-based signals described in Trudnowski (1992a, b). The term *white noise* is used to describe signals with a flat frequency content; *fractal noise* is used to describe signals with a frequency content that rolls off at 10 dB/decade. These types of signals often occur in physical systems, including power systems.

Table 5.1. Test System 1

Poles / (2π)	Residues (absolute < angle°)
-0.0762 +/-j 1.814	286.6 < -/+170.9°
-0.0716 +/-j 1.560	470.2 < -/+134.9°
-0.0048 +/-j 0.6875	5.50 < -/+110.1°
-0.0261 +/-j 0.3613	11.8 < -/+98.7°
-3.104 +/-j 4.998	786.7 < -/+92.4°
-1.8303	-1371.0

Table 5.2. Test System 2

Poles / (2π)	Residues (absolute < angle°)
-0.0412 +/-j 21.02	1.05 < -/+169.0°
-0.0621 +/-j 18.67	2.24 < -/+172.8°
-0.0544 +/-j 1.040	1.92 < -/+175.2°
-0.2040 +/-j 2.037	0.260 < -/+1.46°
-0.0383 +/-j 10.31	2.65 < +/-1.73°
-0.1187 +/-j 45.85	0.514 < -/+1.50°

Table 5.3. Test System 3

Poles / (2π)	Residues (absolute < angle°)
-0.0233 +/-j 1.830	0.0232 < +/-74.8°
-0.0462 +/-j 1.018	0.0096 < -/+60.3°
-0.0088 +/-j 0.7880	0.0129 < -/+140.3°
-0.2081 +/-j 0.7380	0.175 < +/-91.7°
-0.0247 +/-j 0.6914	0.0333 < +/-117.6°
-0.0221 +/-j 0.4776	0.0123 < -/+42.3°
-0.0241 +/-j 0.4335	0.219 < +/-102.9°
-0.0195 +/-j 0.3261	0.150 < +/-68.5°
-0.0025 +/-j 0.2208	0.0029 < +/-104.3°
-0.0697 +/-j 0.1526	0.0684 < +/-87.8°
-0.0528	0.0889
-0.0031 +/-j 0.0336	0.0233 < -/+122.8°

Figure 5.2. Frequency Response of Test System 1

Figure 5.3. Frequency Response of Test System 2

Figure 5.4. Frequency Response of Test System 3

5.2 Pseudo-Random Probing

Two steps are described in Section 2 for identifying a transfer-function model with pseudo-random probing. The first involves estimating $G(j\omega)$ with $G_e(j\omega)$ using periodogram averaging; the second involves estimating the transfer-function terms from $G_e(j\omega)$. In applying these steps, several parameters must be selected. With the objective of obtaining the most accurate transfer function estimate possible, several issues must be considered in selecting parameters. Recommendations on how to select the parameters are given in this section, along with several demonstrative examples.

Critical parameters in the periodogram averaging include number of data points (N), window sizes (M), window function type, amount of window overlap, and the magnitude and frequency content of u . Several tradeoffs exist between the parameters; some general conclusions include those listed here:

- There is a tradeoff between the magnitude of u and N . As u becomes weaker, more data must be used to accurately estimate the frequency response.
- The sample period T is selected to obey the Nyquist criterion. For electromechanical dynamics, a sample period of 0.05 s is often satisfactory.
- Overlap provides smoothing to the higher frequency noise. For the parameters used with Test Systems 1 and 3, the overlap tends to considerably improve the accuracy of $G_e(j\omega)$ above 2 Hz, while improvement below 2 Hz is less dramatic. In general, an overlap of at least 75% should be used (90% is most often used). Increasing above 75% improves the accuracy of $G_e(j\omega)$ only marginally. As the overlap is increased, the required computational time for the periodogram estimate increases.
- Use of a standard window (such as a Hanning window) prevents leakage in the periodogram estimate but degrades the frequency resolution. Experiments with several windows indicate that all have approximately the same effect. Because of their wide acceptance in the signal processing industry, it is recommended that one use a Hanning or Hanning-squared window.
- The window size should be selected to be large enough to obtain quality frequency resolution but small enough to allow enough terms in the periodogram average. Through several experiments, it seems that M should be between 512 and 2048 for electromechanical problems.
- As M becomes larger, smoothing is lost if N or the magnitude of u is not also increased.
- The coherence function provides an indication of the quality of the frequency response estimate $G_e(j\omega)$. One should vary N and M until the coherence function is consistently close to unity over the frequency range of interest. Details on how to vary the parameters are summarized in Section 2 and demonstrated in the following examples.

It is impossible to provide one set of parameters for all problems, and the proper selection depends on several factors. But, based on the experiments conducted, several recommendations can be made for power system applications. For example, in applying this method to electromechanical problems where the dynamics of the system are multimodal between 0.2 Hz and 5 Hz, good starting values might be $N = 4096$ (at a sample rate of 0.05 s), $M = 1024$ with a Hanning

window, overlap = 90%, u is band-limited between 0.1 Hz and 5 Hz, and its amplitude is increased until the coherence function approaches unity. With these starting values, the periodogram parameters are varied as the coherence function dictates until an accurate $G_e(j\omega)$ is obtained. Prony analysis (via PRSPAK and ORDER) is used to estimate the poles and residues. The following subsections provide further examples and discussion related to different applications and issues.

5.2.1 Periodogram Averaging

Obtaining a quality $G_e(j\omega)$ estimate is critical to identifying an accurate transfer function. The coherence function is the indicator on judging the quality of $G_e(j\omega)$ and provides information on how to adjust periodogram parameters.

As an example, consider Test System 3. Input u is band-limited between 0 and 5 Hz, and its amplitude is set so that the SNR at y is 13 dB for the effects of u and n_p , and 33 dB for the effects of u and n_m . Process noise effects are larger, as they tend to be larger in power systems. The key periodogram parameters are $T = 0.05$ s, $N = 4096$, $M = 512$. Figure 5.5 shows the periodogram estimate $G_e(j\omega)$ versus the actual system response $G(j\omega)$, and Figure 5.6 shows the corresponding coherence function. Three characteristics should be noted in the coherence function: 1) it reaches a maximum near 0.9; 2) it appears to have a random component; and 3) it decreases considerably near peaks in $G(j\omega)$. The first two characteristics indicate that the amplitude of u should be increased and/or N should be increased to provide more information and smoothing.

Figures 5.7 and 5.8 show the responses for increasing the amplitude of u by a factor of 4. The coherence function is now smoother and closer to unity except near the system poles. Comparing Figures 5.5 and 5.7 shows that the $G_e(j\omega)$ estimate has been smoothed near the higher frequency end but appears no more accurate. It is no more accurate because more resolution is needed (i.e., M should be increased). M is increased until the coherence function no longer falls over the range of the system poles; the new $G_e(j\omega)$ is shown in Figures 5.9 and 5.10 for $M = 2048$. Note that the coherence function is now close to unity over the entire frequency range and that $G_e(j\omega)$ accurately estimates $G(j\omega)$.

5.2.2 Pseudo-Random Probing Versus Pulse Probing

In this section, two identification approaches are compared. With the first, a pulse input is used as the probing signal; with the second, a pseudo-random signal band-limited between 0 and 5 Hz is used. Figure 5.11 shows the two probing signals. The pseudo-random probing signal is applied for 3.4 min, and the analysis parameters are $N = 4096$, $M = 1024$, 90% overlap, and $T = 0.05$ s. Noise signals n_p and n_m are adjusted so that the SNR is 12 dB for n_p and 33 dB for n_m at the output. The pulse-probing signal has an amplitude of 0.05 (an effective SNR of 25 dB at the output). Figure 5.12 shows that the output for pseudo-random probing is much less than for pulse probing. In an application perspective, this means the system (i.e., the generator) is much less stressed by the pseudo-random probing signal.

Prony analysis is used to identify models from the pulse-input response and from the IDFT of the pseudo-random response. Tables 5.4 and 5.5 show the identified poles and residues, and the errors when compared with the actual system. For the most part, pseudo-random probing provided better estimates of the poles, while the residues were more accurately identified from the pulse response. If the amplitude of the pseudo-random input is increased to achieve the same SNR as the pulse input, then both the poles and residues identified from the pseudo-random response

Figure 5.5. Estimating Test System 3 Response with 13 dB SNR Due to n_p and 33 dB SNR Due to n_m , $N = 4096$, and $M = 512$

Figure 5.6. Coherence for Test System 3 Response with 13 dB SNR Due to n_p and 33 dB SNR Due to n_m , $N = 4096$, and $M = 512$

Figure 5.7. Estimating Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 512$

Figure 5.8. Coherence for Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 512$

Figure 5.9. Estimating Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 2048$

Figure 5.10. Coherence for Test System 3 Response with 25 dB SNR Due to n_p and 45 dB SNR Due to n_m , $N = 4096$, and $M = 2048$

Figure 5.11. Comparison of Pulse Probing with Random Probing for Test System 1

Figure 5.12. Test System 1 Response to Pulse Probing and Random Probing

Table 5.4. Identified Poles for Test System 1, Pulse Probing Versus Pseudo-Random Probing

Identified Poles / (2π) (pulse probing)	Error (pulse probing)	Identified Poles / (2π) (random probing)	Error (random probing)
-0.0431 +/-j 1.5613	0.0285	-0.0718 +/-j 1.5640	0.0043
-0.0882 +/-j 1.8148	0.0120	-0.0769 +/-j 1.8226	0.0083
-0.0296 +/-j 0.3597	0.0038	-0.0337 +/-j 0.3610	0.0076
-0.0068 +/-j 0.6899	0.0032	-0.0078 +/-j 0.6856	0.0036

Table 5.5. Identified Residues for Test System 1, Pulse Probing Versus Pseudo-Random Probing

Residues (absolute < angle°) (pulse probing)	Error (pulse probing)	Residues (absolute < angle°) (random probing)	Error (random probing)
439.6 < -/+ 129.0°	56.24	461.5 < -/+ 150.8°	129.0
366.0 < -/+ 174.3°	81.69	278.6 < +/- 166.2°	112.3
12.81 < -/+ 102.0°	1.196	13.40 < -/+ 106.6°	2.324
7.370 < -/+ 115.5°	1.946	5.194 < -/+ 131.2°	1.984

are more accurate. To obtain similar accuracy with pulse probing, the amplitude of the pulse would need to be increased, causing even more stress on the system.

Although a more accurate model can be identified with pseudo-random probing with less stress on the system, considerably more data is required. For this example, 10 s of the pulse response was used for identification, compared with 205 s of pseudo-random data.

5.2.3 Band-Limited Identification

By limiting the bandwidth of u , one can focus on identifying the characteristics of a system in a specified bandwidth without exciting terms outside u 's bandwidth. This is demonstrated using Test System 2. The objective is to identify the 21-Hz and 18-Hz modes under noisy conditions. Figure 5.13 shows the system response with and without probing. In the no-probing case, the output is caused by n_p and n_m . With probing, the output is caused by u , n_p , and n_m . This probing signal represents a SNR of 0 dB for n_p and 20 dB for n_m . To excite only the 21-Hz and 18-Hz modes, the frequency response of the probing signal is sharply band-limited between 15 Hz and 25 Hz. Two parameter cases are considered: 1) $N = 32000$, $M = 2048$, 75% window overlap; and 2) $N = 131023$, $M = 8192$, 75% overlap. For both cases, $T = 0.005$ s.

Figure 5.14 shows the actual frequency response versus that estimated for the two cases. Both cases appear to accurately estimate the frequency response. Figure 5.15 shows the coherency functions for both cases. Note that with $N = 131023$ and $M = 2048$, the function is closer to 1, indicating a more accurate estimate of $G(j\omega)$.

Figure 5.13. Test System 2 Output for Probing at 0 dB SNR Versus Noise-Only Output

Figure 5.14. Frequency Response of Test System 2 for $N = 32000$ and $N = 131023$ Versus the Actual Response

Figure 5.15. Coherence for Test System 2 for $N = 32000$ and $N = 131023$ Versus the Actual Response

To use Prony analysis to estimate the system modes, the impulse response is calculated using an IDFT of the estimated frequency response. In calculating the IDFT, the frequency response outside the 15-Hz and 25-Hz range is ignored by assuming it to be zero because the probing signal does not excite this range. Table 5.6 shows the estimated modes. Because the $N = 131023$ case uses more data to estimate the system's frequency response, it identifies the modes more accurately. But, the modes are still identified relatively accurately in the $N = 32000$ case.

Table 5.6. Identified Poles for Test System 2, $N = 32000$ Versus $N = 131023$

Identified Poles / (2π) ($N = 32000$)	Error ($N = 32000$)	Identified Poles / (2π) ($N = 131023$)	Error ($N = 131023$)
-0.0719 +/-j 18.67	0.0098	-0.0655 +/-j 18.67	0.034
-0.0658 +/-j 21.00	0.0317	-0.0407 +/-j 21.02	0.0005

5.2.4 Incorporating More Data

The previous example demonstrated the advantage of using more data to increase the accuracy of transfer-function terms. The example shown here further expands this point. Two cases for Test System 3 are compared; with the first, $N = 5950$, and with the second, $N = 32000$. Probing signal u is adjusted to be band-limited between 0 and 5 Hz and to provide enough excitation so the SNR for n_p is near 20 dB and the SNR for n_m is near 45 dB. Table 5.7 shows the

identified poles for the two cases. As can be seen, using more data tends to result in a more accurate model. As expected, the errors tend to be larger for the poles with smaller residues. The frequency response for the identified $N = 131023$ model is shown in Figure 5.16.

Table 5.7. Identified Poles for Test System 3, $N = 5950$ Versus $N = 32000$

Identified Poles / (2π) ($N = 32000$)	Error ($N = 32000$)	Identified Poles / (2π) ($N = 131023$)	Error ($N = 131023$)
-0.0204 +/-j 1.834	0.0044	-0.0239 +/-j 1.831	0.0008
-0.0464 +/-j 1.025	0.0072	-0.0444 +/-j 1.015	0.0032
-0.0107 +/-j 0.789	0.0020	-0.0093 +/-j 0.7875	0.0007
Not Identified	Not Identified	-0.1764 +/-j 0.7780	0.0510
-0.0273 +/-j 0.6870	0.0050	-0.0256 +/-j 0.6924	0.0014
-0.0292 +/-j 0.4170	0.0610	-0.0625 +/-j 0.4872	0.0416
-0.0321 +/-j 0.4370	0.0088	-0.0255 +/-j 0.4342	0.0016
-0.0215 +/-j 0.3281	0.0029	-0.0205 +/-j 0.3260	0.0010
-0.0218 +/-j 0.2224	0.0194	-0.0115 +/-j 0.2239	0.0095
-0.0397 +/-j 0.1577	0.0304	-0.0570 +/-j 0.1452	0.0147
Not Identified	Not Identified	-0.0745	0.0217
-0.0087 +/-j 0.0275	0.0083	-0.0036 +/-j 0.0309	0.0028

Figure 5.16. Model and Periodogram Frequency Response Versus Actual System Response for Test System 3, $N = 32000$

5.3 Chirp Probing

As outlined in Section 2, the process for identifying a model using a chirp input is very similar to that for a random input. In both cases, the system's frequency response must first be estimated. The quality of this estimate is critical to obtaining a quality transfer-function fit. In general, probing with a pseudo-random signal will provide a better quality frequency response estimate than with a chirp. This is due primarily to the stationarity of the random signal, which allows one to overlap the time-domain response in periodogram averaging. This overlapping provides extra windows to the average, which, in turn, results in a lower estimation variance (Bendat and Piersol 1980).

To demonstrate this, pseudo-random probing is compared with chirp probing for Test System 1. The pseudo-random input is band-limited between 0 and 5 Hz, and its amplitude is set so that the SNR at y is 8 dB for n_p and 42 dB for n_m . Periodogram averaging parameters are $T = 0.05$ s, $N = 21599$, $M = 2048$, and 75% overlap. This test requires approximately 18 min of data from the system. For comparison, the chirp input uses similar parameters. The starting frequency is 0.05 Hz and increases by 0.01 Hz/s to 2.5 Hz; the amplitude is adjusted so that the same SNRs are achieved as with the pseudo-random probing. The chirp input is applied to the system five consecutive times, and spectral estimates are averaged to obtain $G_e(j\omega)$. The five chirp experiments require approximately a total of 20 min of data.

Figure 5.17 shows the coherency functions, and Figure 5.18 shows the frequency response of $G_e(j\omega)$ for the two inputs. When compared with the system's actual response as shown in Figure 5.2, the pseudo-random result is more accurate. Prony fits are conducted to the IDFT of the two responses, with the results shown in Tables 5.8 and 5.9. The damping estimates have larger errors for the chirp fit, especially for the 0.36-Hz and 0.69-Hz modes. This is despite the fact that more data was used for the chirp-input approach.

5.4 Discrete Sweep

The last approach for identifying a transfer function model is the use of a discrete-sweep input. In general, this approach can result in similar or better solutions than pseudo-random or chirp probing. This is especially true for systems that are time varying, which, for some power system applications, is a significant issue. The tradeoffs are that the experiment can take considerably more time and that the coherency function cannot be used to judge the quality of the fit.

In comparing random probing with discrete-step probing, one should consider the following hypothesis: to identify a model of a system with a specified accuracy, a given amount of energy must be applied at the input in the frequency range of interest. With random probing, the power of the input contains the entire frequency range but at a very low level for any given frequency. To achieve the energy required, the signal must be applied for a relatively long period. With a discrete sweep, the power is focused at a given frequency with the same overall power as the random input; therefore, it takes much less time to obtain the needed energy at that frequency.

Figure 5.17. Coherency Estimate for Test System 1, Chirp Probing Versus Pseudo-Random Probing

Figure 5.18. Frequency Response Estimates for Test System 1, Chirp Probing Versus Pseudo-Random Probing

Table 5.8. Identified Poles for Test System 1, Chirp Probing Versus Pseudo-Random Probing

Identified Poles / (2π) (chirp probing)	Error (chirp probing)	Identified Poles / (2π) (random probing)	Error (random probing)
-0.0829 +/-j 1.811	0.0066 +j 0.0036	-0.0806 +/-j 1.825	0.0044 +j 0.0108
-0.0845 +/-j 1.551	0.0128 +j 0.0086	-0.0696 +/-j 1.559	0.0020 +j 0.0011
0.0088 +/-j 0.6866	0.0136 +j 0.0010	-0.0053 +/-j 0.6772	0.0005 +j 0.0104
-0.0127 +/-j 0.3698	0.0134 +j 0.0085	-0.0266 +/-j 0.3580	0.0005 +j 0.0033

Table 5.9. Identified Residues for Test System 1, Chirp Probing Versus Pseudo-Random Probing

Residues (absolute < angle°) (chirp probing)	Error (chirp probing)	Residues (absolute < angle°) (random probing)	Error (random probing)
311.5 < +/-178.8°	48.06	303.5 < +/- 163.5°	131.4
617.3 < +/-154.3°	233.3	435.6 < +/- 146.9°	100.9
4.579 < +/- 117.6°	1.145	5.632 < +/- 100.4°	0.9492
11.99 < +/- 123.8°	5.165	11.415 < +/- 98.35°	0.4301

Two primary parameters must be chosen when applying a discrete-step input: the amplitude of the sine wave and the number of cycles to be applied. The larger the amplitude, the fewer cycles required. Some general guideline in applying discrete-sweep inputs include the following:

- Each sinusoid should be applied for several seconds before calculations are initiated allowing the output response to reach steady-state. The delay depends on the stability of the system. A safe value is five times the dominant pole's time constant (including complex poles).
- The coherence function is difficult to apply because, in most cases, the sinusoid is not applied to the system long enough to provide enough data for periodogram averaging.
- Two primary parameters must be chosen when applying a discrete-step input: the amplitude of the sine wave and the number of cycles to be applied. The larger the amplitude, the fewer cycles required.
- Discrete-step probing allows one to use non-equally spaced frequency domain points, which cannot be done with random and chirp inputs. This allows one to concentrate on a given frequency range of interest or to identify a very specific frequency range while not exciting modes outside that range.

The following examples demonstrate the accuracy that can be obtained with discrete probing.

5.4.1 Discrete-Sweep Versus Pseudo-Random and Chirp

Once again, consider Test System 1. The input is adjusted to be the same amplitude as used in the experiment described in Section 5.3 for pseudo-random and chirp inputs. The input frequency is varied from 0.1 Hz to 5.0 Hz in 0.1-Hz steps, resulting in 50 frequency domain points. Each frequency is applied for 12 s to allow the system to reach its steady-state condition, and then five cycles are used to calculate the frequency response. This experiment requires approximately 15 min of data. Figure 5.19 shows the estimated frequency response, and the identified model from Prony analysis is in Table 5.10. Comparing the data in Tables 5.8 and 5.9 with that in Table 5.10 shows that similar accuracy can be achieved in identifying the system poles, but that the system residues are more accurately identified with a discrete-sweep input.

With this experiment, only 50 data points are used to estimate $G_e(j\omega)$ but, with the pseudo-random and chirp probing used in Section 5.3, over 500 data points were available in the frequency range of interest. To improve the frequency resolution of the discrete-step probing, 100 data points are taken between 0 and 5 Hz. The identified model is shown in Table 5.11. The errors on the identified parameters are not significantly less than 50 data point case. This is because the 50 data points provide 100 time-domain data points after the IDFT operation, which is more than enough data to obtain a quality fit using Prony analysis. Including more data improves the quality of the fit only slightly.

Figure 5.19. Discrete-Sweep-Input Frequency Response for Steps of 0.1 Hz

Table 5.10. Identified Model for Discrete-Sweep Input for Steps of 0.1 Hz

Identified Poles / (2π)	Error (poles)	Residues (absolute < angle°)	Error (residues)
-0.0703 +/-j 1.823	0.0059 +/-j 0.0089	272.5 < -/+177.2°	33.65
-0.0734 +/-j 1.562	0.0018 +/-j 0.0017	464.1 < -/+ 136.0°	10.93
0.0056 +/-j 0.6752	0.0103 +/-j 0.0123	7.782 < -/+ 118.2°	2.444
-0.0463 +/-j 0.3514	0.0202 +/-j 0.0099	13.34 < -/+ 98.07°	1.506

Table 5.11. Identified Model for Discrete-Sweep Input for Steps of 0.05 Hz

Identified Poles / (2π)	Error (poles)	Residues (absolute < angle°)	Error (residues)
-0.0722 +/-j 1.813	0.0040 +/-j 0.0017	279.3 < -/+ 169.1°	11.43
-0.0711 +/-j 1.563	0.0005 +/-j 0.0029	464.8 < -/+ 136.8°	16.03
-0.0019 +/-j 0.6623	0.0029 +/-j 0.0252	5.691 < -/+ 89.67°	2.000
-0.0482 +/-j 0.3583	0.0221 +/-j 0.0030	16.74 < -/+ 94.18°	5.032

5.4.2 FREQID Versus Prony

Two approaches are possible for identifying a model from $G_e(j\omega)$: 1) Prony analysis of the IDFT of $G_e(j\omega)$; or 2) directly fit to $G_e(j\omega)$. In most cases, the first approach provides better results. One exception is when only a relatively few data points are available in a small frequency band. In this case, the second approach appears to provide better results.

As an example, consider Test System 3 with the objective of identifying the two dominant modes near 0.33 Hz and 0.43 Hz without exciting modes outside this range. A discrete sweep is used to estimate $G_e(j\omega)$ from 0.25 Hz to 0.5 Hz in 0.01-Hz steps, resulting in 26 data points. FREQID is used to identify the modes directly from the frequency-domain data, and Prony analysis is used by assuming the frequency response is zero from 0 Hz to 0.25 Hz and from 0.5 Hz to 5.0 Hz. The results are compared in Table 5.12. In this case, FREQID performs considerably better.

Table 5.12. Identified Poles for Test System 3, FREQID Versus Prony for a Few Data Points

Identified Poles / (2π) (FREQID)	Error (FREQID)	Identified Poles / (2π) (Prony)	Error (Prony)
-0.0325 +/-j 0.4408	0.0084 +j 0.0073	-0.0044 +/-j 0.4480	0.0197 +j 0.0145
-0.0219 +/-j 0.3317	0.0024 +j 0.0056	-0.0056 +/-j 0.2860	0.0138 +j 0.0402

5.5 Comparing Approaches

In conducting the experiments reported in this section, several conclusions became apparent. Some of the more relevant ones include the following:

- Considerably higher noise levels can be tolerated with pseudo-random and swept-sine probing than with pulse probing. The level of tolerable SNR depends on the amount of data used to estimate the system's frequency response. One example here demonstrates accurate identification even at 0 dB SNR.
- The tradeoff for tolerating a smaller SNR is the requirement for more data, which means longer experiments. For power system electromechanical dynamics problems, this can mean several minutes of data.
- In general, pseudo-random probing provides better results than a chirp input. Discrete-sweep probing is a very complementary approach to pseudo-random probing. Overall, discrete-sweep provides the most accurate results (especially for system residues). The advantage of random probing is that the coherence function can be used to evaluate the accuracy of the frequency response estimate.
- Only the portions of $G_e(j\omega)$ that are in the bandwidth of u should be used for the transfer-function identification step. Terms outside this range are assumed zero when conducting IDFT operations for the Prony analysis [see Equation (3.7)] or are not included in the frequency-domain fitting.
- In general, Prony analysis provides more accurate model fits than direct frequency-domain fitting via FREQID except when only a few data points are used (approximately 50 or less). An example of when direct frequency-domain fitting is more appropriate is determining the dominant poles from a single peak of $G_e(j\omega)$.
- Once g_e is obtained, the recommendations provided in Trudnowski (1992a) should be used to conduct the Prony analysis.
- Although not investigated here, SYSFIT can be used to improve the model fit by using the Prony or FREQID model as the starting point for the SYSFIT model.

6.0 Conclusions and Future Tasks

Major milestones completed during 1993 include scoping application of pseudo-random and swept-sine probing, updating and writing major analysis routines, and forming a vision for the final product for the PSI Toolbox. If the current level of effort is maintained, it is expected that Phase Two will be completed by the end of 1994, and Phase Three will be completed by the end of 1995.

6.1 Conclusions

Several tasks were completed in 1993. These tasks center around forming a vision for the final product of the PSI Toolbox, developing new analysis routines, updating and modifying existing routines, and forming application guidelines for analyzing a system's response to pseudo-random and swept-sine probing.

The final product for the PSI Toolbox will be a collection of flexible FORTRAN analysis routines that can easily be incorporated into other analysis programs or used in a stand-alone form. Significant documentation and guidelines being developed will demonstrate the fundamental applications of the codes for solving power system dynamics problems.

Modifications made to PRSPAK (the Prony analysis routine) include updating the codes to include the latest numerical algorithms and developing a multi-output version. Effort has also been placed on developing expert routines such as error-bound analysis, model ordering, and transfer-function identification. Much of this work has been coordinated and conducted with the support of the University of Colorado.

A new analysis routine developed this past year is FREQID, a frequency-domain transfer-function identification routine. FREQID fits a single-input single-output model to the frequency response of a system using a linear least-squares approach. In general, it is less numerically robust compared to conducting a Prony analysis of the IDFT of the frequency-domain data. But, it does not require equally spaced data and does outperform Prony analysis when only a few (near 50) frequency-domain data points are available.

When investigating the use of pseudo-random and swept-sine probing for model identifications, several conclusions became apparent. Some of the more relevant include the following:

- Considerably higher noise levels can be tolerated with pseudo-random and swept-sine probing than with pulse probing. For power system applications, this means the probing signal can be of considerably small magnitude.
- The tradeoff for tolerating more noise is the requirement for more data, which means longer experiments. For power system electromechanical dynamics problems, this can mean several minutes of data.
- Pseudo-random and discrete-sweep probing provide better results than a chirp-sweep input.
- In general, Prony analysis provides more accurate model fits than direct frequency-domain

fitting except when only a few data points are used (approximately 50 or less). An example of when direct frequency-domain fitting is more appropriate is determining the dominant poles from a single peak of $G_e(j\omega)$.

6.2 Future Tasks

Continued development of the PSI Toolbox will focus on completing Phases Two and Three as milestones. Assuming that the current support for the project is maintained, it is projected that Phase Two will be completed at the end of 1994; Phase Three will be completed at the end of 1995. At the end of Phase Three, available products will include a formalized version of the codes and a set of manuals describing proper use, general application guidelines, and specific power system application guidelines.

During 1994, Phase Two tasks will focus on

- continued code development and refinement to include new algorithms and to make existing codes more robust, efficient, and flexible - For example, a specific task includes maximizing the amount of variable dimensioning within the codes. This will make incorporating the codes into other packages easier.
- testing and evaluating new algorithms - This includes evaluation of multi-output Prony analysis, error-bound algorithms developed this year, and identifying systems that have a feedback path.
- continued application guideline development - Major effort will be placed on guidelines for analysis of transient-simulation and measured data. Other efforts include coordinated use between Prony analysis, FREQID, and SYSFIT.

Specific objectives and tasks will be formed for Phase Three as Phase Two is concluded. These tasks will focus on reorganizing the software into a modular collection of documented computer programs and developing user manuals with instruction and application guidelines.

7.0 References

- Agee, J. 1991. *Excitation System Commissioning Procedures*. REC-ERC-91-2, U.S. Bureau of Reclamation, Denver, Colorado.
- Anderson E., Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. 1992. *LAPACK User's Guide*. Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania.
- Bendat, J., and A. Piersol. 1980. *Engineering Applications of Correlation and Spectral Analysis*. John Wiley & Sons, New York.
- Donnelly, M., J. Smith, R. Johnson, J. Hauer, R. Brush, and R. Adapa. 1993. "Control of a Dynamic Brake to Reduce Turbine-Generator Shaft Transient Torques." *IEEE Transactions on Power Systems* 8(1):67-73.
- Franklin, G., J. Powell, and M. Workman. 1990. *Digital Control of Dynamic Systems*. 2nd ed. Addison-Wesley, New York.
- Hauer, J. F. 1974. *MORPAK-II: An Accelerated Projection Program for Constrained Nonlinear Minimization*. Technical Report TR74-13, Department of Computer Science, University of Alberta, Edmonton, Alberta, Canada.
- Hauer, J. F. 1977. *Computer Routines for Fitting Transfer Functions to System Response Measurements*. Bonneville Power Administration, Portland, Oregon.
- Hauer, J. F. 1982. "Power System Identification by Fitting Structured Models to Measured Frequency Response." *IEEE Transactions on Power Systems* PAS-101(4):915-922.
- Hauer, J. F. 1989a. "Identification of Power System Models for Large-Scale Damping Control." In *Proceedings of the 28th IEEE Conference on Decision and Control*, pp. 1841-1846. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.
- Hauer, J. 1989b. "The Use of Prony Analysis to Determine Modal Content and Equivalent Models for Measured Power System Response." In *IEEE Symposium on Eigenanalysis and Frequency Domain Methods for System Dynamic Performance*, pp. 105-115. Publication 90TH0292-3-PWR, Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.
- Hauer, J. F., and C. W. Taylor. 1990. *Feedback Control and Frequency Domain Analysis of Large Electric Power Systems, Volume 1: Seminar Workbook*. Carson Taylor Seminars, Portland, Oregon.
- Institute of Electrical and Electronics Engineers. 1992. *Current Activity in Flexible AC Transmission Systems (FACTS)*. 92 TH 0465-5 PWR, New York.
- Johnson, J., and D. Trudowski. 1993. "Frequency-Domain Transfer-Function Identification using Chebychev Polynomials." *Proceedings of the 32nd IEEE Conference on Decision and Control*, Vol. 4, pp. 3510-3511. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.
- Kamwa, I., R. Grondin, J. Dickinson, and S. Fortin. 1993. "A Minimal Realization Approach to Reduced-Order Modelling and Modal Analysis for Power System Response Signals." *IEEE Transactions on Power Systems* 8(3):1020-1029.

McWhorter, T., and L. L. Scharf. 1993. "Cramer-Rao Bounds for Deterministic Modal Analysis." *IEEE Transactions on Signal Processing* 41(5):1847-1866.

Oppenheim, A., and R. Schaffer. 1975. *Digital Signal Processing*. Prentice-Hall, Inc., New York.

Pierre, D., D. Trudnowski, and J. Hauer. 1992. "Identifying Linear Reduced-Order Models for Systems with Arbitrary Initial Conditions Using Prony Signal Analysis." *IEEE Transactions on Automatic Control* 37(6):831-835.

Scharf, L. 1991. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Addison-Wesley Publishing Company, New York.

Soderstrom, T., and P. Stocica. 1989. *System Identification*. Prentice Hall, Inc., New York.

Soysal, A., and A. Semlyen. 1992. "Practical Transfer Function Estimation and Its Application to Wide Frequency Range Representation of Transformers." Paper no. 92SM358-2-PWRD, 1992 IEEE Power Engineering Society Summer Meeting. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Tichy, W. F. 1985. "RCS -- A System for Version Control." *Software -- Practice & Experience* 15(7):637-654.

Trudnowski, D., J. Smith, T. Short, and D. Pierre. 1991. "An Application of Prony Methods in PSS Design for Multimachine Systems." *IEEE Transactions on Power Systems* 6(1):118-126.

Trudnowski, D. 1992a. *Characteristics of Identifying Linear Dynamic Models from Impulse Response Data Using Prony Analysis*. PNL-8450, Pacific Northwest Laboratory, Richland, Washington.

Trudnowski, D. 1992b. *Frequency Domain Transfer Function Identification Using the Computer Program SYSFIT*. PNL-8455, Pacific Northwest Laboratory, Richland, Washington.

Trudnowski, D., C. Baker, and M. Evans. 1993. "Damping Control of a Large Flexible Manipulator Through Inertial Forces of a Small Manipulator." In *Proceedings of the 1993 American Control Conference*, Vol. 3, pp. 2878-2879. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Trudnowski, D. 1993. "Order Reduction of Large-Scale Linear Oscillatory System Models." Paper no. 93WM209-7PWRS, 1993 IEEE Power Engineering Society Winter Meeting. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Trudnowski, D., M. Donnelly, and J. Hauer. 1993. "Advances in the Identification of Transfer Function Models Using Prony Analysis." In *Proceedings of the 1993 American Control Conference*, Vol. 2, pp. 1561-1562. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Appendix

PSI Toolbox Copyholders

Appendix

PSI Toolbox Copyholders

As of January 31, 1994, the people listed below have requested and received a copy of the PSI Toolbox:

Mr. Jay Agee
U.S. Bureau of Reclamation
P.O. Box 25007, D-3772
Denver, CO 80225

Dr. Jim McCalley
111 Coover Hall
Iowa State University
Ames, IA 50011

Mr. C. E. Grund
Power Systems Engineering Department
General Electric Company
One River Road
Schenectady, NY 12345

Mr. Henry Mosteller
Power Systems Engineering Department
General Electric Company
One River Road
Schenectady, NY 12345

Dr. Serge Lefebvre
IREQ
1800, montee Ste-Julie
Varennas, Quebec
CANADA J3X 1S1

Mr. Yakout Mansour
BC Hydro and Power Authority
970 Burrard Street
Vancouver, British Columbia
CANADA V6Z 1Y3

Mr. A Roman Messina
Col. Lindavista
07300, Mexico, D.F.
MEXICO

Dr. Atef S. Moreched
Ontario Hydro
Toronto, Ontario
CANADA M56 1X6

Mr. Rob O'Keef
American Electric Power
1 River Side Plaza
Columbus, OH 43215

Professor Louis Scharf
Department of Electrical Engineering
University of Colorado
Boulder, CO 80309-0425

Mr. Don Sutphin
Transmission Planning, Z6A
Pacific Gas and Electric Company
P.O. Box 770000
San Francisco, CA 94177

Distribution

No. of
Copies

Offsite

12 DOE/Office of Scientific and
Technical Information

5 J. F. Hauer
EOH
Bonneville Power Administration
P.O. Box 3621
Portland, OR 97208

J. J. Ray
EO
Bonneville Power Administration
P.O. Box 3621
Portland, OR 97208

C. W. Taylor
EOH
Bonneville Power Administration
P.O. Box 3621
Portland, OR 97208

J. Agee
M.S. D-3772
U.S. Bureau of Reclamation
P.O. Box 25007
Denver, CO 80225

L. L. Scharf
Department of Electrical and Comp-
uter Engineering
Engineering Center, OT 3-12
University of Colorado at Boulder
Boulder, CO 80309-0425

D. A. Pierre
Electrical Engineering Department
Montana State University
Bozeman, MT 59717

J. L. Luini
Pacific Gas and Electric Company
Transmission Planning, Z6A
P.O. Box 770000
San Francisco, CA 94177

No. of
Copies

Foreign

G. J. Rogers
Cherry Tree Scientific Software
RR #5
Colborne, Ontario K0K150
CANADA

Onsite

DOE Richland Operations Office

D. D. Green

18 Pacific Northwest Laboratory

J. E. Dagle K5-20
M. K. Donnelly K5-20
D. J. Trudnowski (10) K5-20
Publishing Coordination
Technical Report Files (5)