An Equivalent Model Generator for Large Scale Linear Networks Based on A System Identification Approach.

by

WIESLAW T. KWASNICKI

A thesis
submitted for
the Degree of Master of Science
in the Faculty of Graduate Studies
University of Manitoba

Electrical Engineering Department
University of Manitoba
© **August 15, 1986**

Permission has been granted to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film.

The author (copyright owner) has reserved other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without his/her written permission.

L'autorisation a été accordée à la Bibliothèque nationale du Canada de microfilmer cette thèse et de prêter ou de vendre des exemplaires du film.

L'auteur (titulaire du droit d'auteur) se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation écrite.

ISBN 0-315-34022-3

AN EQUIVALENT MODEL GENERATOR FOR LARGE SCALE LINEAR NETWORKS BASED ON A SYSTEM IDENTIFICATION APPROACH

BY

WIESLAW T. KWASNICKI

A thesis submitted to the Faculty of Graduate Studies of
the University of Manitoba in partial fulfillment of the requirements
of the degree of

MASTER OF SCIENCE

© 1986

Permission has been granted to the LIBRARY OF THE UNIVERSITY OF MANITOBA to lend or sell copies of this thesis, to the NATIONAL LIBRARY OF CANADA to microfilm this thesis and to lend or sell copies of the film, and UNIVERSITY MICROFILMS to publish an abstract of this thesis.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

Abstract

For simulation and analysis purposes, a large linear network or its parts can be substituted by the Thevenin equivalent model. This model contains two parts : a voltage source and a series impedance. Each of these parts is modelled separately in the form of a difference equation.

The unknown coefficients of the model difference equations are estimated using the least-squares identification technique. The input-output observations required for the identification procedure are taken from the network when a pseudo-random binary test signal (PRBS) is applied. This type of test signal provides maximum information about the network dynamics preserved in the measurement sequence.

When measurements are contaminated by measurement noise, the least-squares estimates of the unknown parameters suffer from bias. This bias can be eliminated using linear regression and filtering techniques.

To improve estimation efficiency, the number of observations or the order of the model equations can be increased. For this purpose, recursive least-squares procedures are developed. These procedures update the previous estimates of the unknown parameters when new observations are supplied or a higher order model is desired. The updating process does not involve matrix inversion.

A technique has been developed which reduces large linear systems modelled on a power system transient modelling program (EMTDC) to ones of smaller order and/or larger timestep. The simplified models can then be used in future simulations with time and memory economy.

The complete package of computer programs for the Thevenin equivalent model generation and its use in the simulation process is prepared. The results and comparision with other simulation techniques are provided.

Some suggestions for further development and applications of the model building technique are also proposed.

Ackowledgement

The author wishes to express sincere appreciation to Professor A. Gole, for his guidance, continuous interest and encouragement through all phases of this work.

Special thanks are due to Professor M. Pawlak and Mr. F. Franczyk for their suggestions and comments to the problems throughout the course of this study.

Financial assistance from the Manitoba HVDC Research Centre and the National Research Council are gratefully acknowledged.

Finally, my sincere thanks to my wife and children for their support, understanding and unexpected patience during the time of this work.

Table of Contents.

CHAPTER 1.

Introduction.

Electrical networks with a large number of elements are always difficult to analyze. Using the state or node method, we obtain a large number of dynamic differential equations describing the behavior of the network. The solution of such a set of equations involves many steps and transformations and thus is very time consuming.

When we are interested only in the behavior of the network at the terminals where it is conected to some external network, we can replace this network by another equivalent network which provides the same current-voltage relationship at the terminals.

A very convenient method of obtaining such an equivalent network is given by Thevenin's Theorem [5]. This theorem states that an active network can be represented by a voltage source and a series impedance. The voltage source produces a voltage that is equal to the one at the open terminals of the active network and the series impedance is the network impedance seen from the terminals when all the network sources are reduced to zero.

Therefore, the Thevenin equivalent network model we would like to construct, is composed of two parts : the voltage source model and the series impedance model.

The models we are concerned with are the discrete-time models in the form of linear difference equations of unknown orders. For the construction of an equivalent network model we need two such equations. One for the voltage source model that relates the network source voltage and the open terminal voltage and another one for the series impedance model that relates voltage and current at the output terminals when all the interior sources are reduced to zero.

The unknowns to be determined are the coefficients and orders of these two difference equations. Thus, the problem of network modelling reduces to the problem of parameter identification [1,2].

Applying system identification techniques, we can provide the statistical estimates of the unknown parameters. These estimates are obtained from the sequence of observations taken from the network and they optimally fit the model equation to the observed data. When the least-square identification technique is used, the optimal fit means that the error function, which is the sum of squared errors, is minimized.

As the order of the model equation increases, the error function decreases significantly until the true value of model order is reached. Thus the appropriate order of the model equation can be chosen as the one for which the error function ceases to decrease significantly.

The efficiency of the estimates depends on the number of observations taken for the identification procedure, the measurement error, and the type of signals present in the network when it is observed [7].

To assure the most accurate estimates, the white random process is used as a test signal. Such a signal has a wide power spectrum, so that the network is excited by a wide spectrum of frequencies and hence the measurements contain maximum information about the dynamic modes of the network.

When the measurements are contaminated with error, the least-square estimates may be biased. This bias can be eliminated by filtering the error in such a way that the filtered error converts to a white noise [10].

A very important problem is the choice of a sampling time in which the measurements are taken. It should be such that allows us to observe the network dynamics.

Clearly, the values of the model parameters depend on the sampling time. Therefore, its value must always go together with the set of estimated parameters.

A complete Thevenin equivalent network model thus contains two sets of equation parameter estimates, their orders, and the sampling time. This information allow us to simulate the network behavior at the output terminals at any time and for various external networks connected to the terminals.

To produce the desired voltage at the output terminals, we use the Thevenin equivalent model in the following manner. First we compute the Thevenin source voltage from the available knowledge of the network source voltage and the source model. Then, from the measurements of the current at the output terminals and from the impedance model, we compute the "voltage drop" due to the series impedance. The output voltage is the difference between the Thevenin source voltage and the "voltage drop".

The proposed method for the modelling of large networks needs some effort to gather the observations and then build the model. However, when the model is created it is then very handy in use and storage.

Furthermore, when using the recursive algorithm for the least-square solution, we can update parameter estimates step by step as the new experimental data are continuously in supply [3]. This on-line estimation procedure does not involve matrix inversion and therefore the computational effort is much reduced. Our knowledge about the modeled network can be improved any time the new observations are avaiable.

When the observations are appropriately weighted according to their age, we may be able to track parameter changes due to the time or environment effect on the network. The stage of the network can then be monitored by the set of model parameters that are continuously updated.

A parameter vector can be used as a feature vector for the pattern recognition procedure. This procedure can classify vectors of parameter estimates into a chosen number of classes [6]. If the classes correspond to certain types of faults, then we will be able to detect and classify the faults of the network. This is reserved for a future study.

The fault detection problem is not discussed in this work but it is indicated how the computer program package, created for the network modelling purpose, can be adapted to achieve this task.

CHAPTER 2.

System Identification.

In this chapter we describe the general problem of system identification and derive algorithms for the estimation of the unknown parameters of a system model equation. These algorithms will be used in the next chapter to model linear networks.

The solution of the basic identification problem shows how the input-output observations on the system may be used to estimate the unknown parameters of its model. This solution involves matrix inversion, which is very inconvenient when the system model is of a higher order. In order to avoid matrix inversion, some extensive estimation schemes are proposed. These are the recursive procedures for updating parameter estimates when the number of observations or the model order increases.

We also discuss some related problems such as : a system with output noise and associated with this a bias of the estimate, system order determination and the design of an optimum test signal.

1. General Description of the Problem.

The problem of system identification is generally referred to as the determination of a mathematical model for a system or a process by observing its input-output relationship.

Figure 2.1 shows a system with its input and output. The system model we are seeking is the mathematical equation that relates the input to the output at all times. In order to obtain such a model, we are permited to probe the system with a variety of inputs and observe its responses. The input-output data are then

processed to yield the model. On the basis of the degree of *a priori knowledge* about the system, we can classify the system identification problem into two categories :

1. The complete identification problem :

   where we don't know anything about the basic properties of the system.

2. The partial identification problem :

   where some basic characteristics of the system, such as linearity and bandwidth are assumed to be known but we may not know the specific order of the dynamic equation or the value of the associated coefficients.

Fortunately, the majority of the engineering systems we encounter in practice are of the later type. In many cases, we know a good deal about the structure of the system, so that it is possible to derive a specific mathematical model of the system dynamics. Consequently, only a set of parameters in the model equation are left to be determined. Thus the modelling problem is reduced to that of *parameter identification*. From the viewpoint of system theory, we can precisely determine the unknown parameters in an exact system model equation where the exact measurements of the input-output data are given. In reality however, the input-output data are corrupted by measurement noise. Furthermore, there are inaccuracies in the model equation as well. Therefore, the determination of the system parameters is essentially a statistical estimation problem : we seek to specify a mathematical model that fits the noisy observation data.

We can choose a number of possible representations for a given system, including models characterized in the frequency domain or the time domain, in continuous-time or discrete-time. In the time domain representation, we have the choice between a weighting function (or sequence), a differential (or difference) equation, or a state variable equation. The choice depends on the identification objective and its associated input-output data.

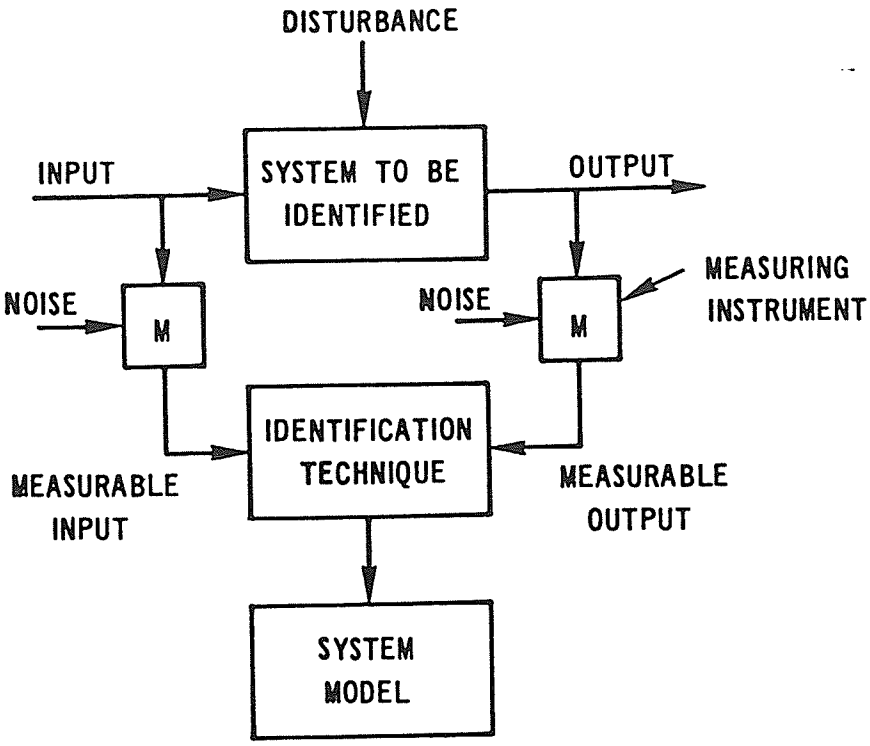


Fig.2.1 Block Diagram Representation of the System Identification Problem.[1]

Since information from continuous signals can be preserved in samples taken in appropriate sampling frequencies, continuous systems can be closely approximated by discrete models. The great majority of system identification techniques are digitally oriented as a result of the employment of digital computers. Therefore, discrete system models are more convenient to deal with.

We can choose various forms of representation for the linear dynamic systems but each of the forms is uniquely related to the other. Thus, a linear system can be regarded as completely identified once a particular form of representation is obtained.

For our purpose, it is very convenient to use the difference equation representation since it has a simple relationship with the transfer function of a system.

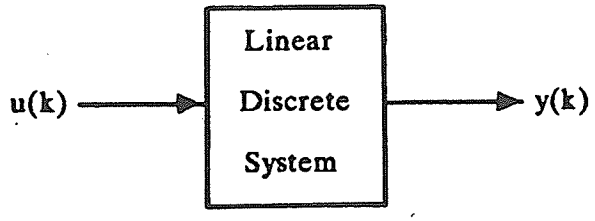


Fig.2.2 A Single-input and Single-output Discrete System.

The general n-th order difference equation relating the input u(k) and output y(k) for a single-variable, linear discrete system is :

$$y(k)+a_1y(k-1)+\cdots+a_ny(k-n) = b_0u(k)+b_1u(k-1)+\cdots+b_nu(k-n)$$

or

$$y(k) + \sum_{j=1}^{j=n} a_j y(k-j) = \sum_{j=0}^{j=n} b_j u(k-j) \tag{2.1}$$

where k is the integer time index, and a_j and b_j are the constant coefficients.

This equation can be rewritten in the following shifting-operator form

$$A(q^{-1})\cdot y(k) = B(q^{-1})\cdot u(k) \tag{2.2}$$

where

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_nq^{-n}$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \cdots + b_nq^{-n}$$

and q^{-1} is a shifting operator : $q^{-j}\cdot f(k) = f(k-j)$.

Applying the Z-transformation on equation (2.1), assuming zero initial conditions

$$y(k) = u(k) = 0 \quad \text{for } k \leq 0$$

we can obtain the following

$$\left[1 + a_1 z^{-1} + \cdots + a_n z^{-n} \right] \cdot Y(z) = \left[b_0 + b_1 z^{-1} + \cdots + b_n z^{-n} \right] \cdot U(z)$$

where z is the complex Z-transform variable.

The transfer function can be defined as

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} = \frac{B(z^{-1})}{A(z^{-1})} \qquad (2.3)$$

We can see that the transfer function of a system is directly related to its difference equation.

The representation of equation (2.1) can be extended to systems that have multiple inputs and outputs .

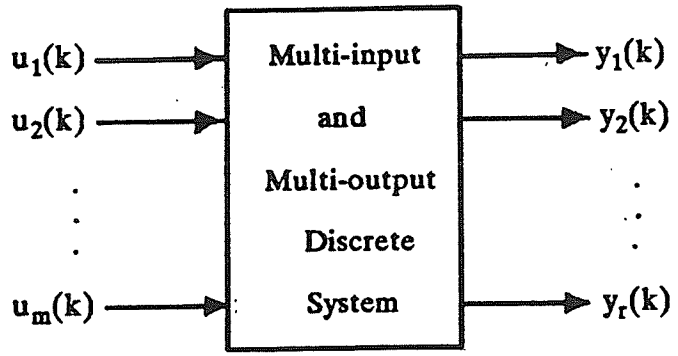


Fig.2.3 Multi-input and Multi-output Discrete System.

Consider that there are m inputs and r outputs, as in Fig.2.3, and that we define the vectors $\bar{u}(k)$ and $\bar{y}(k)$ as :

$$\bar{u}(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_m(k) \end{bmatrix} \quad ; \quad \bar{y}(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_r(k) \end{bmatrix}$$

The system can then be represented by the vector difference equation

$$\bar{y}(k) + \sum_{j=1}^{j=n} A_j \bar{y}(k-j) = \sum_{j=0}^{j=n} B_j \bar{u}(k-j) \qquad (2.4)$$

in which A_j and B_j are constant coefficient matrices of dimension rxr and rxm respectively.

2. Linear Parametric Model Identification by Least Squares.

The least-squares identification technique is the most popular among engineers and scientists. It has been successfully applied to many engineering problems, and hence our approach to the network modelling problem is also based on this technique [1,2,3,4].

In this section we describe the principles of this technique and examine some important properties of the least-square estimate.

As in the case for all estimation techniques, we are seeking to minimize certain appropriately defined error criterion as a means to optimally fit the model to the system data.

There are two modes, in which identification can be accomplished :

1) *Off-line identification,* in which a record of input-output data is first observed and the model parameters are then estimated based on the entire data record.

2) *On-line identification,* in which the parameter estimates are recursively calculated for every data set so that the new data is used to correct and update the existing estimates.

If the updating process can be made very fast, it becomes possible to obtain parameter estimates of time varying systems with reasonable accuracy.

This capability is called *on-line real-time identification.*

2.1. The Basic Identification Problem.

We consider a single-variable linear discrete system as shown in Fig.2.4 where u(nT) and y(nT) are the samples of the system's input and output signals and T is the constant sampling period.

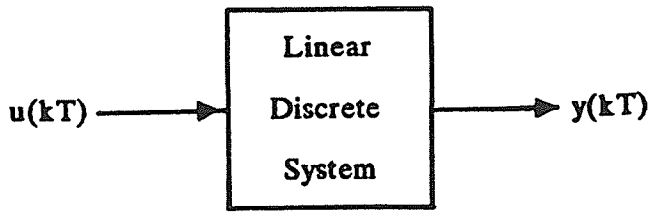


Fig.2.4 A Stable Single-variable Discrete System.

Let the system be described by an n-th order difference equation of constant coefficients :

$$y(k) + \sum_{j=1}^{j=n} a_j y(k-j) = \sum_{j=0}^{j=n} b_j u(k-j) \qquad (2.1)$$

where T is dropped from u(kT) and y(kT) for the sake of simplicity.

Assuming that the system is stable and that it is in steady state operation we state *the basic identification problem* as follows.

Given n and the input-output measurements :

$$\left\{ u(k) , y(k) ; k=1,2, ... , N+n \right\}$$

estimate the constant system parameters :

$$\left\{ b_0 , (a_i , b_i) ; i=1, ... , n \right\}$$

We will solve the above problem by fitting the system equation (2.1) to the input-output data so as to chose the best parameter values in the sense of minimum-square-error.

To accomplish this, we first write the system equation as :

$$y(k) = -\sum_{j=1}^{j=n} a_j y(k-j) + \sum_{j=0}^{j=n} b_j u(k-j) + e(k) \qquad (2.5)$$

The term e(k) is introduced to account for the *fitting error*.

This error is schematically depicted in Fig.2.5.

Define the 2n+1-dimensional input-output vector $\bar{x}$ (k) as :

$$\bar{x}(k) = \left[u(k), -y(k-1), u(k-1), \cdots , -y(k-n), u(k-n) \right]$$

and the 2n+1-dimensional parameter vector :

$$\theta = \left[b_0, a_1, b_1, \cdots , a_n, b_n \right]^T$$

Then we can write (2.5) in the following vector notation :

$$y(k) = \bar{x}(k) \cdot \theta + e(k) \qquad (2.6)$$

Because we have available a string of data

$$\left\{ u(k) , y(k) ; k=1, \cdots , N+n \right\}$$

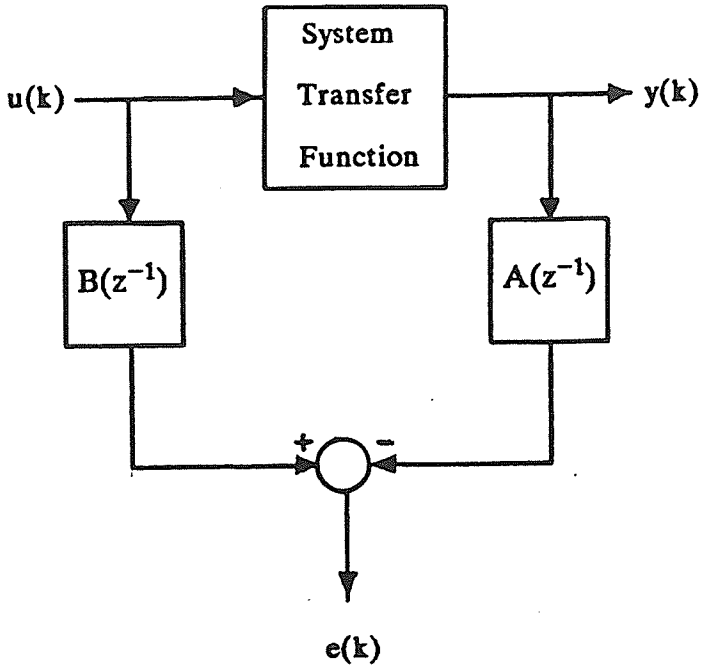


Fig.2.5 Block Diagram of the Error Signal e(k).

we can set up a system of N equations (assume N>>2n) as :

$$\bar{y} = \mathbf{X}\cdot\theta + \bar{e} \tag{2.7}$$

where

$$\bar{y} = \left[y(n+1), y(n+2), \cdots, y(n+N)\right]^T$$

$$\bar{e} = \left[e(n+1), e(k+2), \cdots, e(n+N)\right]^T$$

$$\mathbf{X} = \begin{bmatrix} \bar{x}(n+1) \\ \bar{x}(n+2) \\ \vdots \\ \bar{x}(n+N) \end{bmatrix} = \begin{bmatrix} u(n+1), & -y(n), & u(n), & \cdots & -y(1), u(1) \\ u(n+2), & -y(n+1), & u(n+1), & \cdots & -y(2), u(2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ u(n+N), & -y(n+N-1), & u(n+N-1), & \cdots & -y(N), u(N) \end{bmatrix}$$

With the vector equation (2.7), in which $\bar{y}$ and $\mathbf{X}$ are given, we can conveniently estimate the parameter vector θ by means of a least squares technique.

This identification approach was first suggested by Kalman in 1958 [7]. The complete solution is presented below.

The least-squares procedure says that we choose the estimate $\hat{\theta}$ as that value of θ that minimizes the error function J :

$$J = \sum_{k=n+1}^{N+n} e^2(k) = \bar{e}^T \cdot \bar{e}$$

or

$$J = \left[\bar{y} - \mathbf{X}\cdot\theta\right]^T \cdot \left[\bar{y} - \mathbf{X}\cdot\theta\right] \tag{2.8}$$

Upon setting

$$\left[\frac{\partial J}{\partial \theta}\right]_{\theta=\hat{\theta}} = -2\mathbf{X}^T\bar{y} + 2\mathbf{X}^T\mathbf{X}\hat{\theta} = 0$$

we immediately obtain the least-squares estimate $\hat{\theta}$ by

$$\hat{\theta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1} \cdot \mathbf{X}^T\bar{y} \tag{2.9}$$

This solution exists if $\mathbf{X}^T\mathbf{X}$ is nonsingular. We assume that this condition is met by the input-output data.

The type of input sequence {u(k)} driving the system that yields a nonsingular matrix $\left(\mathbf{X}^T\mathbf{X}\right)_{mxm}$ is called *persistently exciting* of order m.

In order to examine the statistical properties of $\hat{\theta}$, let us assume that e(k) are independent random variables with zero mean and variance σ^2, that is

$$E[e(k)] = 0 \quad \text{for } k = 1,2, \cdots, N+n$$

$$E[e(i)\cdot e(j)] = \sigma^2\delta_{ij} \text{ where } \delta_{ij} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases} \text{ for } i,j=1,2,\cdots, N+n.$$

Furthermore, if we assume that e(k) is uncorrelated with u(k) and y(k), we observe the following properties of $\hat{\theta}$.

$$\hat{\theta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{y}$$

$$= \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\left(\mathbf{X}\cdot\theta + \bar{e}\right)$$

$$= \theta + \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{e}$$

or

$$\hat{\theta}-\theta = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{e}$$

Thus the estimator bias can be expressed as

$$E\left[\hat{\theta}-\theta\right] = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{e}\right]$$

and since $\mathbf{X}$ and $\bar{e}$ are uncorrelated

$$E\left[\hat{\theta}-\theta\right] = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\right]\cdot E\left[\bar{e}\right] = 0$$

Therefore, when the error is a random process with zero mean, $\hat{\theta}$ is unbiased. That is $E[\hat{\theta}] = \theta$.

Next, let us evaluate the covariance of $\hat{\theta}$.

$$\Psi = E\left[\left(\hat{\theta}-\theta\right)\cdot\left(\hat{\theta}-\theta\right)^T\right]$$

$$= E\left\{\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{e}\cdot\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{e}\right]^T\right\}$$

$$= E\left[\left(\mathbb{X}^T\mathbb{X}\right)^{-1}\cdot\mathbb{X}^T\right]\cdot E\left[\bar{e}\cdot\bar{e}^T\right]\cdot E\left[\mathbb{X}\cdot\left(\mathbb{X}^T\mathbb{X}\right)^{-1}\right]$$

but we have

$$E\left[\bar{e}\cdot\bar{e}^T\right] = \sigma^2\cdot I$$

thus for deterministic $\mathbb{X}$ (u and y measured without error)

$$\Psi = \sigma^2\cdot\left(\mathbb{X}^T\mathbb{X}\right)^{-1}$$

$$= \frac{\sigma^2}{N}\cdot\left[\frac{1}{N}\cdot\left(\mathbb{X}^T\mathbb{X}\right)\right]^{-1}$$

Since we have assumed that the system is stable and that the input signal is such that $\left(\mathbb{X}^T\mathbb{X}\right)$ is nonsingular for any length of the string data N, the limit

$$\lim_{N\to\infty}\frac{1}{N}\cdot\left(\mathbb{X}^T\mathbb{X}\right) = \Phi$$

exists and Φ is positive definite.

Thus

$$\lim_{N\to\infty}\Psi = \lim_{N\to\infty}\frac{\sigma^2}{N}\cdot\Phi^{-1} = 0$$

which implies that $\hat{\theta}$ converges to θ as $N\to\infty$.

Zero error covariance means that $\hat{\theta}=\theta$ at $N\to\infty$. Hence, $\hat{\theta}$ is a consistent estimator of θ.

2.2. System With Output Noise.

When the output measurements are contaminated with noise then the least-squares estimation applied to the generalized model (2.5) suffers from a bias in the estimates. Consequently the estimates are wrong and we usually have a great deal of confidence in the wrong results.

Thus the properties of the estimates for the system with output noise need to be studied in some detail.

Shown in Fig.2.6 is a system with noise n(k) at the output

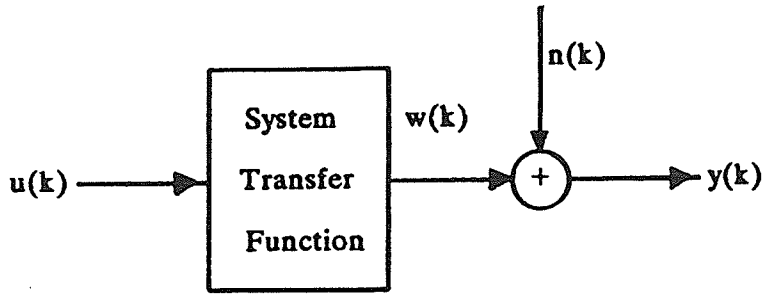


Fig.2.6 System with Output Noise.

We wish to interpret the random fitting error e(k) in terms of an output measurement noise.

From Fig.2.6, we have the following relationship

$$\begin{cases} A(q^{-1})\cdot w(k) = B(q^{-1})\cdot u(k) \\ y(k) = w(k) + n(k) \end{cases} \tag{2.10}$$

Simplifying yields

$$A(q^{-1})\cdot y(k) = B(q^{-1})\cdot u(k) + A(q^{-1})\cdot n(k) \tag{2.11}$$

Comparing equations (2.11) and (2.5), we immediately see that

$$e(k) = A(q^{-1})\cdot n(k) \tag{2.12}$$

This means that n(k) is the output of a filter $1/A(q^{-1})$ with e(k) as the input.

The significance of this analysis is that the identification of the noisy system in Fig.2.6 is no different from that of the system in Fig.2.5 so long as n(k) is filtered white noise (white or uncorrelated residual).

In general, in the noisy system model

$$\begin{cases} A(q^{-1})\cdot y(k) = B(q^{-1})\cdot u(k) + \epsilon(k) \\ \epsilon(k) = A(q^{-1})\cdot n(k) \end{cases} \tag{2.13}$$

$\epsilon(k)$ is an autocorrelated random process (correlated residual).

We can show that in this case, the bias arises with the application of the least-squares estimation. To show this, we recall that the LSE of θ is given by

$$\hat{\theta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\cdot\bar{y} \tag{2.14}$$

Since the output of the system is contaminated with noise

$$y(k) = w(k) + n(k)$$

we can partition the matrix $\mathbf{X}$ into two parts :

$$\mathbf{X} = \mathbf{X}_{yu} = \mathbf{X}_{wu} + \mathbf{X}_{n0} \tag{2.15}$$

where the martrices $\mathbf{X}_{wu}$ and $\mathbf{X}_{n0}$ correspond to the actual output of the system and to the noise added at its output respectively.

We have

$$\bar{y} = \mathbf{X}_{wu}\cdot\theta + \bar{n} \tag{2.16}$$

$$\mathbf{X}_{wu} = \mathbf{X}_{yu} - \mathbf{X}_{n0} \tag{2.17}$$

Substitution of these two relations into (2.14) leads to

$$\hat{\theta} = \left[\mathbb{X}_{yu}^T \mathbb{X}_{yu}\right]^{-1} \cdot \mathbb{X}_{yu}^T \cdot \bar{y}$$

$$= \left[\mathbb{X}_{yu}^T \mathbb{X}_{yu}\right]^{-1} \cdot \mathbb{X}_{yu}^T \cdot \left[\mathbb{X}_{wu} \cdot \theta + \bar{n}\right]$$

$$= \left[\mathbb{X}_{yu}^T \mathbb{X}_{yu}\right]^{-1} \cdot \mathbb{X}_{yu}^T \cdot \left[\left(\mathbb{X}_{yu} - \mathbb{X}_{n0}\right) \cdot \theta + \bar{n}\right]$$

$$= \theta - \left[\mathbb{X}_{yu}^T \mathbb{X}_{yu}\right]^{-1} \cdot \mathbb{X}_{yu}^T \cdot \left[\mathbb{X}_{n0} \cdot \theta - \bar{n}\right]$$

The second right-hand term indicates the bias of the estimate.

The bias does not occur if

$$E\left[\left(\mathbb{X}_{uy}^T \mathbb{X}_{uy}\right)^{-1} \cdot \mathbb{X}_{uy}^T \cdot \left(\mathbb{X}_{n0} \cdot \theta - \bar{n}\right)\right] = 0 \qquad (2.18)$$

This condition can be met only if n(k) is generated from white noise e(k) by

$$n(k) = e(k)/A(q^{-1}) \qquad (2.19)$$

which is the condition for white residual as in (2.12) [1,2].

In any other case, the LSE of θ will be biased.

To circumvent the deterimental effects of the noise, several techniques have been developed (Astron and Eykhoff 1970) viz. :

1) Repeated Least-squares

2) Instrumental Variables

3) Tally Principle

4) Levin's Method

5) Generalized Least-squares (GLS) and related schemes

6) Maximum Likelihood

In the category of GLS techniques, the Succesive Linear Regression and Filtering approach is discused in section 2.4.

2.3. Recursive Estimation for Increasing Parameter Number.

The exact number of parameters needed to fit the data is usually unknown ahead of time. Thus we woud have to try to make the fit with different parameter numbers in the model equation.

As the number of parameters increases, the computation of the parameter estimates (2.9) becomes more and more complex since it involves matrix inversion of increasing dimension.

Further in this section we show that there is no need to compute the inverse matrix $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ every time the number of parameters increases. We can compute the inverse matrix from the available inverse matrix of the lower dimension. Consequently, the parameter estimates can also be computed recursively from the lower order estimates.

For the model of order zero there is only one parameter to be estimated and the identification procedure involves only a scalar inversion. Thus, increasing the number of parameters by one and using recursive procedure we can completely eliminate the matrix inversion from the estimation procedure.

To derive the recursive algorithm for the, estimation let us consider the n+1-dimensional parameter vector θ

$$\theta_{n+1} = \left[\theta(1), \theta(2), \cdots, \theta(n), \theta(n+1)\right]^T$$

or

$$\theta_{n+1} = \begin{bmatrix} \theta_n^* \\ \theta(n+1) \end{bmatrix} \tag{2.20}$$

where θ_n^* is an n-dimensional vector of the first n coordinates of θ_{n+1}.

The corresponding $\mathbb{X}$ matrix is

$$\mathbf{X}_{n+1} = \begin{bmatrix} \mathbf{X}_n, \mathbf{C}_{n+1} \end{bmatrix} \tag{2.21}$$

where $\mathbf{C}_{n+1}$ is the n+1-th column of the matrix $\mathbf{X}$ involved in (2.7).

The n+1-dimensional estimate of θ can be obtained from (2.9) as

$$\hat{\theta}_{n+1} = \left(\mathbf{X}_{n+1}^T \mathbf{X}_{n+1}\right)^{-1} \cdot \mathbf{X}_{n+1}^T \cdot \bar{y} \tag{2.22}$$

By denoting

$$\mathbf{P}_{n+1} = \mathbf{X}_{n+1}^T \cdot \mathbf{X}_{n+1}$$

$$\mathbf{S}_{n+1} = \mathbf{X}_{n+1}^T \cdot \bar{y}$$

we can write (2.22) as

$$\hat{\theta}_{n+1} = \mathbf{P}_{n+1}^{-1} \cdot \mathbf{S}_{n+1} \tag{2.23}$$

From (2.23) we have

$$\mathbf{P}_{n+1} \cdot \hat{\theta}_{n+1} = \mathbf{S}_{n+1} \tag{2.24}$$

but

$$\mathbf{P}_{n+1} = \mathbf{X}_{n+1}^T \mathbf{X}_{n+1}$$

$$= \begin{bmatrix} \mathbf{X}_n^T \\ \mathbf{C}_{n+1}^T \end{bmatrix} \cdot \begin{bmatrix} \mathbf{X}_n \, , \, \mathbf{C}_{n+1} \end{bmatrix}$$

$$= \left[\begin{array}{c|c} X_n^T X_n & X_n^T C_{n+1} \\ \hline C_{n+1}^T X_n & C_{n+1}^T C_{n+1} \end{array}\right]$$

$$= \left[\begin{array}{c|c} P_n & F_{n+1} \\ \hline F_{n+1}^T & H_{n+1} \end{array}\right] \qquad (2.25)$$

where

$$F_{n+1} = X_n^T \cdot C_{n+1}$$

$$H_{n+1} = C_{n+1}^T \cdot C_{n+1}$$

and

$$S_{n+1} = X_{n+1}^T \cdot \bar{y}$$

$$= \begin{bmatrix} X^T \\ C_{n+1}^T \end{bmatrix} \cdot \bar{y}$$

$$= \begin{bmatrix} S_n \\ C_{n+1}^T \bar{y} \end{bmatrix} \qquad (2.26)$$

Thus, substituting (2.20), (2.25) and (2.26) into (2.24) we obtain :

$$\left[\begin{array}{c|c} P_n & F_{n+1} \\ \hline F_{n+1}^T & H_{n+1} \end{array}\right] \cdot \begin{bmatrix} \hat{\theta}_n^* \\ \hat{\theta}(n+1) \end{bmatrix} = \begin{bmatrix} S_n \\ C_{n+1}^T \bar{y} \end{bmatrix}$$

or

$$P_n \cdot \hat{\theta}_n^* + F_{n+1} \cdot \hat{\theta}(n+1) = S_n \qquad (2.27)$$

$$F_{n+1}^T \cdot \hat{\theta}_n^* + H_{n+1} \cdot \hat{\theta}(n+1) = C_{n+1}^T \cdot \bar{y}$$

The solution of these two equations is

$$\hat{\theta}_n^* = \mathbf{P}_n^{-1}\cdot\mathbf{S}_n - \mathbf{P}_n^{-1}\cdot\mathbf{F}_{n+1}\cdot\hat{\theta}(n+1) \tag{2.28}$$

$$\hat{\theta}(n+1) = \left[\mathbf{C}_{n+1}^T\cdot\bar{y} - \mathbf{F}_{n+1}^T\cdot\mathbf{P}_n^{-1}\cdot\mathbf{S}_n\right]\cdot\left[\mathbf{H}_{n+1} - \mathbf{F}_{n+1}^T\cdot\mathbf{P}_n\cdot\mathbf{F}_{n+1}\right]^{-1}$$

Substituting

$$\mathbf{P}_n^{-1}\cdot\mathbf{S}_n = \hat{\theta}_n$$

$$\mathbf{C}_{n+1}^T\cdot\bar{y} - \mathbf{F}_{n+1}^T\cdot\mathbf{P}_n^{-1}\cdot\mathbf{S}_n = \mathbf{C}_{n+1}^T\cdot\left[\bar{y} - \mathbf{X}_n\cdot\hat{\theta}_n\right] = \mathbf{C}_{n+1}^T\cdot\bar{e}_n$$

and making notations

$$\mathbf{Q}_{n+1} = \mathbf{P}_n^{-1}\cdot\mathbf{F}_{n+1}$$

$$\mathbf{D}_{n+1} = \mathbf{H}_{n+1} - \mathbf{F}_{n+1}^T\cdot\mathbf{P}_n^{-1}\cdot\mathbf{F}_{n+1}$$

we can write (2.28) as

$$\hat{\theta}_n^* = \hat{\theta}_n - \mathbf{Q}_{n+1}\cdot\mathbf{C}_{n+1}^T\cdot\bar{e}_n/\mathbf{D}_{n+1} \tag{2.29}$$

$$\hat{\theta}(n+1) = \mathbf{C}_{n+1}^T\cdot\bar{e}_n/\mathbf{D}_{n+1}$$

To enable the algorithm in equation (2.29) to compute the parameter estimates of the next-higher-order model, we need to compute the inverse matrix $\mathbf{P}_{n+1}^{-1}$ as required in $\mathbf{Q}_{n+1}$ and $\mathbf{D}_{n+1}$.

Since $\mathbf{P}_{n+1}$ is a symmetric matrix thus $\mathbf{P}_{n+1}^{-1}$ must also be symmetric.

Let us partition $\mathbf{P}_{n+1}^{-1}$ as follows :

$$\mathbf{P}_{n+1}^{-1} = \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{B}^T & \mathbf{C} \end{array}\right] \tag{2.30}$$

where the dimensions of the matrices **A** , **B** and **C** are nxn,1xn and 1x1 respectively.

From the definition of an inverse matrix, we have

$$\mathbf{P}_{n+1} \cdot \mathbf{P}_{n+1}^{-1} = \mathbf{I}_{n+1}$$

thus

$$\left[\begin{array}{c|c} \mathbf{P}_{n_n} & \mathbf{F}_{n+1} \\ \hline \mathbf{F}_{n+1}^{T} & \mathbf{H}_{n+1} \end{array}\right] \cdot \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{B}^{T} & \mathbf{C} \end{array}\right] = \mathbf{I}_{n+1}$$

and

$$\mathbf{P}_n \cdot \mathbf{A} + \mathbf{F}_{n+1} \cdot \mathbf{B}^T = \mathbf{I}_n$$

$$\mathbf{P}_n \cdot \mathbf{B} + \mathbf{F}_{n+1} \cdot \mathbf{C} = \bar{0}$$

$$\mathbf{F}_{n+1}^T \cdot \mathbf{A} + \mathbf{H}_{n+1} \cdot \mathbf{B}^T = \bar{0}$$

$$\mathbf{F}_{n+1}^T \cdot \mathbf{A} + \mathbf{H}_{n+1} \cdot \mathbf{C} = 1$$

Solving this set of matrix equations for A,B,and C we obtain :

$$\mathbf{A} = \mathbf{P}_n^{-1} + \mathbf{P}_n^{-1} \cdot \mathbf{F}_{n+1} \cdot \mathbf{F}_{n+1}^T \cdot \mathbf{P}_n^{-1} \cdot \mathbf{C}$$

$$\mathbf{B} = -\mathbf{P}_n^{-1} \cdot \mathbf{F}_{n+1} \cdot \mathbf{C}$$

$$\mathbf{C} = \left[\mathbf{H}_{n+1} - \mathbf{F}_{n+1}^T \cdot \mathbf{P}_n^{-1} \cdot \mathbf{F}_{n+1}\right]^{-1}$$

or using previously introduced notations

$$\mathbf{A} = \mathbf{P}_n^{-1} + \mathbf{Q}_{n+1} \cdot \mathbf{Q}_{n+1}^T / \mathbf{D}_{n+1}$$

$$\mathbf{B} = -\mathbf{Q}_{n+1} / \mathbf{D}_{n+1} \qquad (2.31)$$

$$\mathbf{C} = 1 / \mathbf{D}_{n+1}$$

Equations (2.30) and (2.31) let us compute the inverse matrix $\mathbf{P}_{n+1}^{-1}$ recursively.

Since the computation only involves the previously computed inverse matrix $\mathbf{P}_n^{-1}$ thus (2.31) with the initial inverse matrix

$$\mathbf{P}_1^{-1} = 1 / \mathbf{C}_1^T \cdot \mathbf{C}_1$$

which is a scalar, completely eliminates matrix inversion from the estimation procedure.

We assumed the nonsigularity of the matrix $\mathbf{P}_n$. The existence of the inverse matrix $\mathbf{P}_n^{-1}$ depends on the input-output string data. This problem will be discussed in section 3.

Equations (2.29) and (2.31) show how we can compute the new n+1-th parameter estimates $\hat{\theta}_{n+1}$ from the available knowledge of lower order n-th parameter estimates $\hat{\theta}_n$.

Equation (2.29) indicates that the old parameter estimate is corrected by $\Delta\hat{\theta}_{n+1}$ which is proportional to the error made in the previous estimation

$$\hat{\theta}_{n+1} = \begin{bmatrix} \hat{\theta}_n \\ 0 \end{bmatrix} + \Delta\hat{\theta}_{n+1} \tag{2.32}$$

where $\hat{\theta}_{n+1}$ and $\Delta\hat{\theta}_{n+1}$ have the dimension n+1 and

$$\Delta\hat{\theta}_{n+1} = \begin{bmatrix} -\mathbf{Q}_{n+1} \cdot \mathbf{C}_{n+1}^T \cdot \bar{e}_n / D_{n+1} \\ \mathbf{C}_{n+1}^T \cdot \bar{e}_n / D_{n+1} \end{bmatrix}$$

$$= \begin{bmatrix} -\mathbf{Q}_{n+1} \cdot \mathbf{C}_{n+1}^T / D_{n+1} \\ \mathbf{C}_{n+1}^T / D_{n+1} \end{bmatrix} \cdot \bar{e}_n$$

It can be written in a general notation as

$$\Delta\hat{\theta}_{n+1} = \mathbf{G}_n \cdot \bar{e}_n \tag{2.33}$$

where $\mathbf{G}_n$ is a gain factor matrix

$$G_n = \begin{bmatrix} -Q_{n+1} \cdot C_{n+1}^T / D_{n+1} \\ C_{n+1}^T / D_{n+1} \end{bmatrix} \tag{2.34}$$

The scheme proposed here is called a *model adjustment scheme* which in general, has a gain G_n of varying complexity and can be also applied in procedures without the increase of parameter vector dimension.

When trying to apply this method to the generalized noisy model (2.13), one encounters the bias problem as discussed in the previous section.

To eliminate this bias we need an extensive estimation scheme such as Successive Linear Regression and Filtering which is described in the following section.

2.4. Successive Linear Regression and Filtering.

It was shown that for the noisy output system, the least-squares estimates were biased. Thus we need to modify the least-square solution in order to achieve better parameter identification.

A common technique used to eliminate estimate bias is the filtering of the generalized error. This usually requires estimation of the filter parameters and consequently increases the procedure complexity.

In this section we present a simple technique in which the output function $A(q^{-1})$ is used as an error filter. We thus do not have any other parameters to be estimated but the model parameters.

The generalized error according to Fig.2.5 is

$$\epsilon(k) = A_n(q^{-1}) \cdot y(k) - B_n(q^{-1}) \cdot u(k) \tag{2.35}$$

where the symbol n refers to the parameter values after n iterations.

Instead of choosing the generlized error, a proposal by Steiglitz and Mc Bride [8] (1965) implies the use of the error expression

$$e(k) = \frac{\epsilon(k)}{A_{n-1}(q^{-1})} = \frac{A_n(q^{-1})}{A_{n-1}(q^{-1})} \cdot y(k) - \frac{B_n(q^{-1})}{A_{n-1}(q^{-1})} \cdot u(k)$$

or

$$e(k) = \left\{ y(k) - \frac{B_n(q^{-1})}{A_n(q^{-1})} \cdot u(k) \right\} \cdot \frac{A_n(q^{-1})}{A_{n-1}(q^{-1})} \tag{2.36}$$

We still have a generalized model, linear with parameters but the term between brackets represents an ordinary model. This is illustrated in Fig.2.7

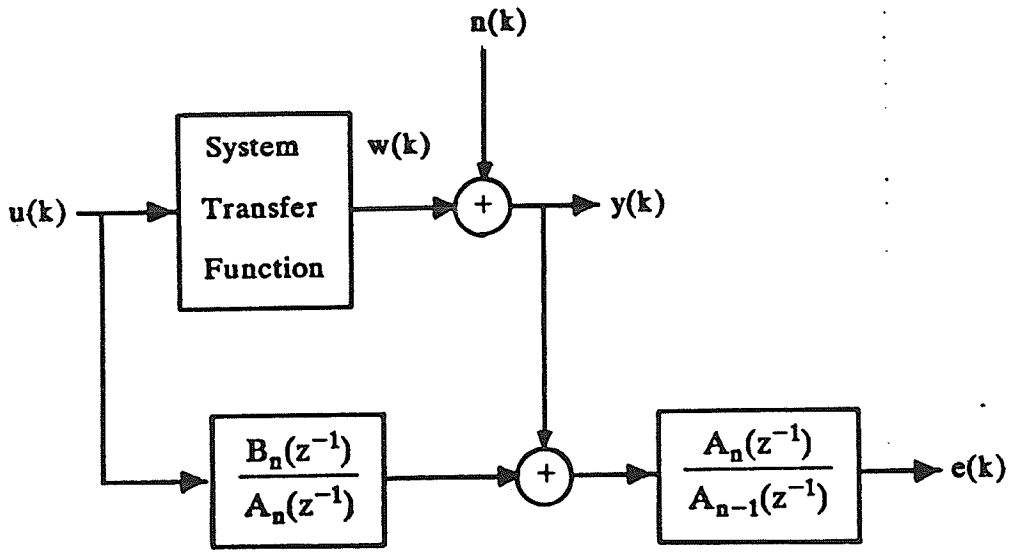


Fig.2.7 Block Diagram Interpretation of the Linear Regression and Filtering Algorithm.

This procedure has the advantage that the additive noise does not cause bias estimates.

The iterative procedure is as follows :

First : Compute LSE of the ordinary model

$$\hat{\theta}_1 = \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\cdot\mathbb{X}^T\bar{y}$$

Second : Filter system input and output signals with the filter

$$\frac{1}{A_1(q^{-1})}$$

Third : Base on the filtered signals estimate $\hat{\theta}_2$, filter input and output and so on.

No proof is known for the convergence of this scheme but it was successfully applied to identify model parameters of the noisy output system [8].

2.5. System Order Determination.

A simple but very effective method for order testing is to observe how the models of different orders fit to the observed data.

The "goodness" of the fit is measured by the error-square-sum function J(n) .

$$J(n) = \left(\bar{y} - \mathbb{X}_n\cdot\hat{\theta}_n\right)^T\cdot\left(\bar{y} - \mathbb{X}_n\cdot\hat{\theta}_n\right) \qquad (2.37)$$

where the index n idicates a given model order.

Let us consider a set of LSE for θ

$$\left\{\hat{\theta}_n \; ; \; n = 1,2, \ldots, K\right\}$$

and the corresponding set of error functions

$$\left\{J(n) \; ; \; n = 1,2, \ldots, K\right\}$$

In general, J(n) decreases as n increases. However, the reduction of J(n) ceases to be significant when n becomes greater than the true (or optimum) order n_0.

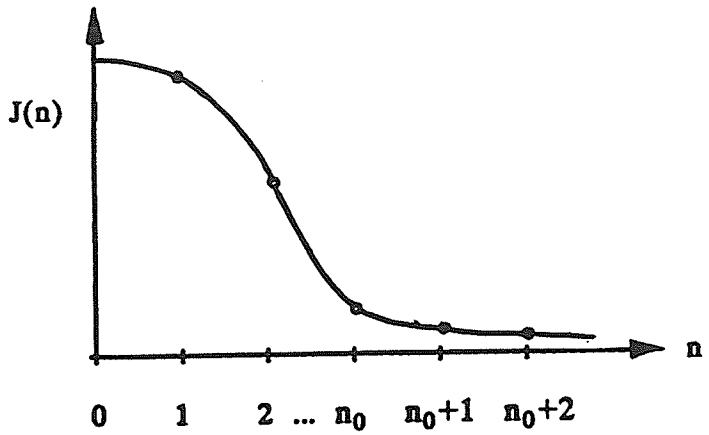


Fig.2.8 A typical shape of the error function.

This principle can be conveniently used to determine the desirable order of the model.

The procedure for order determination is simply to compute the least-squares estimates $\hat{\theta}_n$ and the corresponding error function J(n) for a sequence of model orders n=1,2, ... , K .

The appropriate model order can be chosen as the one that J ceases to decrease significantly.

3. Optimum Test Signal.

The previous sections have been primarily concerned with the problem of extracting information from a given set of data. However, in most situations there are a number of variables which can be adjusted, subject to certain constraints, so that the information provided by the experiments is maximized.

For dynamic systems, experiment design includes :

1) Choice of input and measurement ports.

2) Test signals.

3) Sampling instants.

4) Presampling filters.

Each of these variables has a significant bearing upon the information provided by an experiment. A detailed discussion of all of these aspects is provided by Goodwin in [7]. In this section we discuss only the test signal design problem.

To form a basis for a comparison of different experiments, a measure of the "goodness" of an experiment is required .

A logical approach is to choose a measure related to the expected accuracy of the parameter estimates to be obtained from the data collected. Clearly the parameter accuracy is a function of both the experimental conditions and the form of the estimator .

As mentioned before, LSE is a minimum variance estimator. This means that the covariance matrix corresponding to the error $(\hat{\theta} - \theta)$:

$$\Psi = E\left[\left(\hat{\theta} - \theta\right)\cdot\left(\hat{\theta} - \theta\right)^T\right] \tag{2.38}$$

reaches a minimum value for LSE

$$\hat{\theta} = \left(X^T X\right)^{-1}\cdot X^T\cdot\bar{y} \tag{2.39}$$

or in other words for any other estimates $\hat{\theta}_w$, the covariance matrix

$$\Psi_w = E\left[\left(\hat{\theta}_w - \theta\right)\cdot\left(\hat{\theta}_w - \theta\right)^T\right] \tag{2.40}$$

is such that $\Psi - \Psi_w$ is non-negative definite.

This leads us to choose the covariance matrix Ψ as a measure of the parameter estimation efficiency. The task is to minimize Ψ.

For the noise-disturbed system equation

$$\bar{y} = \mathbf{X}\cdot\theta + \bar{e} \tag{2.41}$$

when $\bar{e}$ is a stationary random vector with zero mean value $E[\bar{e}] = \bar{0}$ and $\bar{e}$ is uncorrelated with $\bar{y}$ and $\mathbf{X}$, the expectation and covariance matrices corresponding to the estimate error $\hat{\theta} - \theta$ are

$$E\left[\left(\hat{\theta} - \theta\right)\right] = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\right]\cdot E[\bar{e}] = \bar{0}$$

and

$$\Psi = COV(\hat{\theta}) = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\right]\cdot E\left[\bar{e}\cdot\bar{e}^T\right]\cdot E\left[\mathbf{X}\cdot\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\right]$$

as was shown in section 2.1.

Defining the covariance matrix of the vector $\bar{e}$ to be

$$\mathbb{R} = E[\bar{e}\cdot\bar{e}^T] \tag{2.42}$$

then Ψ is reduced to

$$\Psi = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\cdot\mathbf{X}^T\right]\cdot\mathbb{R}\cdot E\left[\mathbf{X}\cdot\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\right] \tag{2.43}$$

When the noise e(i) i=1,2, ... is independent and identically distributed with zero mean and variance σ^2 the covariance matrix $\mathbb{R}$ becomes

$$\mathbb{R} = \sigma^2\cdot\mathbf{I}$$

and

$$\Psi = \sigma^2\cdot E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\right]$$

$$= \frac{\sigma^2}{N} \cdot E\left\{\left(\frac{1}{N} \cdot \left(X^T X\right)\right)^{-1}\right\} \tag{2.44}$$

The type of input sequence {u(k)} driving the system should be such that yields a nonsingularity of matrices $\left(X^T X\right)$ (persistantly exciting) and an existence of the limit

$$\lim_{N \to \infty} \left(\frac{1}{N} \cdot \left(X^T X\right)\right)^{-1} = \Gamma \tag{2.45}$$

where Γ is a nonsingular constant matrix.

The structure of the matrix $\frac{1}{N} \cdot \left(X^T X\right)$ is such that we have only four types of its elements which are related to the correlation coefficients of the input-output measurements :

$$\hat{\Phi}_{uu}(j) = \frac{1}{N} \cdot \sum_{i=1}^{N} u(i) \cdot u(i+j)$$

$$\hat{\Phi}_{yy}(j) = \frac{1}{N} \cdot \sum_{i=1}^{N} y(i) \cdot y(i+j)$$

$$\hat{\Phi}_{yu}(j) = \frac{1}{N} \cdot \sum_{i=1}^{N} y(i) \cdot u(i+j)$$

$$\hat{\Phi}_{uy}(j) = \frac{1}{N} \cdot \sum_{i=1}^{N} u(i) \cdot y(i+j)$$

where j= 0,1,2, ... ,n

We observe that when white noise is used as an input signal, then the autocorrelation function $\Phi_{uu}(\tau)$ is a Dirac function and consequently from the relationship between the correlation functions given by the Wiener-Hopf equation

$$\Phi_{uy}(\tau) = \int_0^{\infty} h(t) \cdot \Phi_{uu}(\tau - t) dt \tag{2.46}$$

we obtain that the cross-correlation function $\Phi_{yu}(\tau)$ to simply be the system impulse-response function $h(\tau)$.

Thus the white-noise input signal is persistantly exciting and the limit in (2.45) exists.

Levin [9] has shown that the achievable minimum matrix Ψ is a diagonal matrix and that this is provided by a band-limited white noise sequence used as the test signal.

The practical white noise signal that we can easily generate by a digital circuit is the pseudo-random binary signal (PRBS). The PRBS is a periodic sequence that takes on only two values. The times at which transition can occur are multiples of a specified time interval Δt, and the state for any succeding interval is nearly independent of the state in any preceding interval.

An example of such a signal is shown in Fig.2.10. This signal has a periodic auto-correlation function closely approximating the delta function (Fig.2.11) which is the auto-correlation function of the white-noise test signal.

The approximation can be adjusted by changing N and Δt . The bias in the PRBS autocorrelation function can be minimized if N is large or compensate by adding a corrective d.c. signal to PRBS .

PRBS can be easily generated by a shift register with a modulo-two adder in feedback. Its binary output represents the two states of the PRBS. When an n-element shift register is used, then the output sequence has a period $N = (2^n - 1)$.

A PRBS digital computer simulation program is given in appendix A.

A disadvantage, due to the periodic character of these signals, is that the correlation functions will also be periodic. In order to maintain proper operation of the identification technique, it is necessary to choose a PRBS period $T = N \cdot \Delta t$ large enough such that $\Phi_{uy}(T) \approx 0$. In other words, T must be greater than the

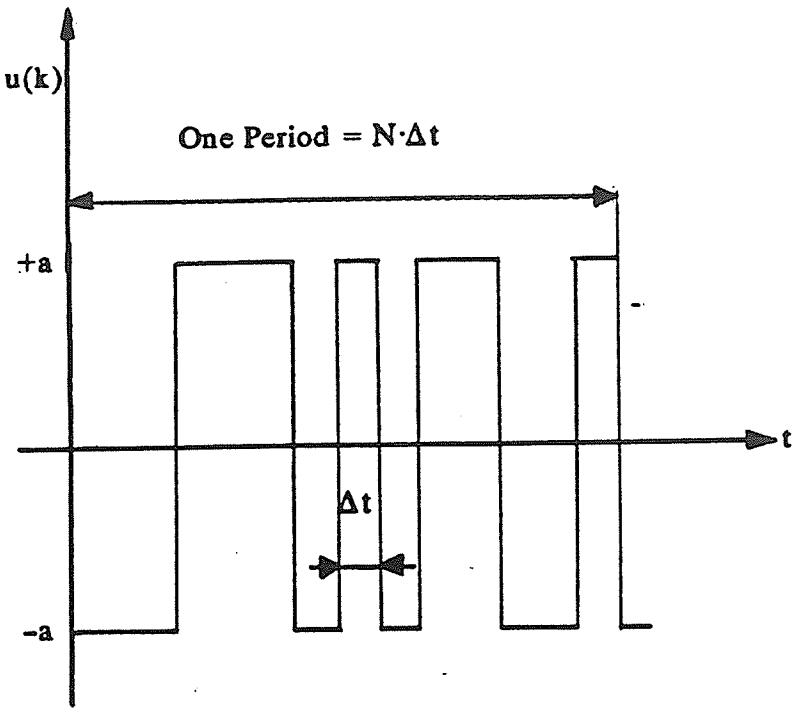


Fig.2.10 A Pseudo-random Binary Sequence (PRBS).

settling time of the weighting function of the system.

Another aspect which can be recognized as a disadvantage, is the lack of freedom to distribute the signal power over the frequency range. Due to the triangular shape of the autocorrelation function, the power spectrum has an envelope of the form (Fig.2.12) :

$$\Delta t \cdot \left[\frac{\sin(\omega \Delta t/2)}{\omega \Delta t/2} \right] \tag{2.48}$$

Due to the periodicity of the correlation function with a period T, the frequency spectrum consists of lines at distances $2\pi/T$.

The use of the PRBS signal can be compared with the use of a large number of sinusoidal components measured at all frequencies simultaneously.

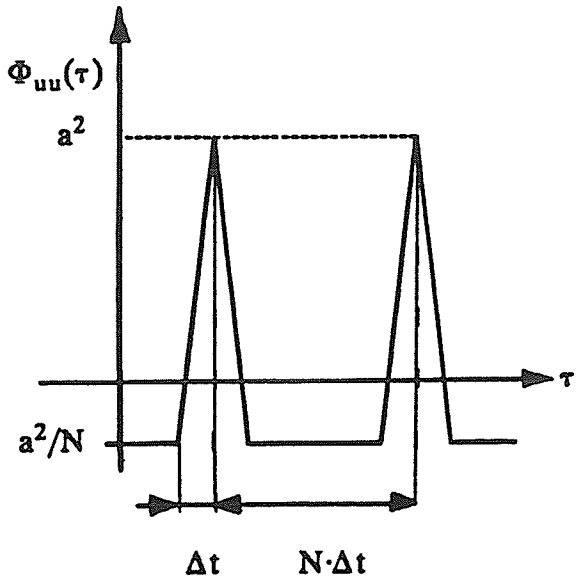


Fig.2.11 Autocorrelation Function of PRBS.

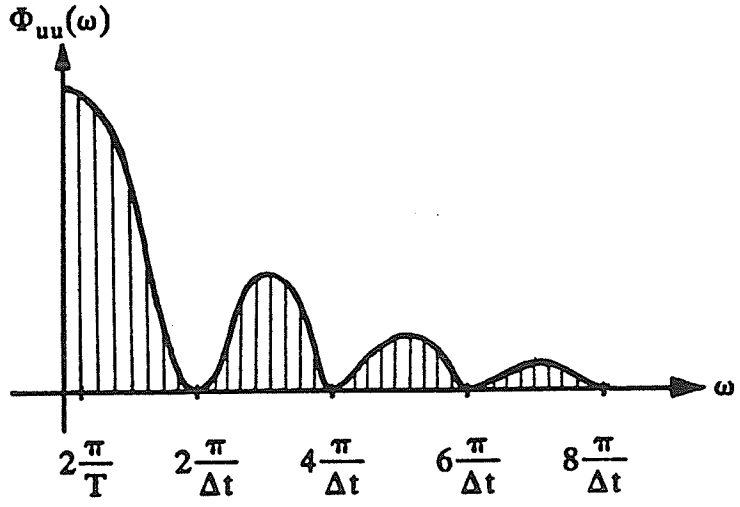


Fig.2.12 Power Spectrum of PRBS Autocorrelation Function.

An important reason for choosing the PRBS as a test signal is the shape of its autocorrelation function. It approximates the Dirac fuction which is the auto-correlation function of a white-noise test signal. However, when using the true triangular shape of the autocorrelation function in equation (2.46), the cross-correlation function may differ from the system impulse-response function.

The following types of error can be recognized :

1) Periodicity of the signal :

   Because the PRBS is periodic, its autocorrelation will have the same property. This implies that when determining the response h(t) of a system, one has to be sure that

$$h(t) \approx 0 \qquad \text{for } t \geq T = N \cdot \Delta t$$

   If this was not the case, then the effect of overlaping contributions due to different peaks of the $\Phi_{uu}(\tau)$ will occur. Consequently, this technique cannot be used on systems having a pure integration in their dynamics.

2) D.C. component of the signal.

   The d.c. component in the test signal leads to a constant term in the auto-correlation functon $\Phi_{uu}(\tau)$, viz. $-a^2/N$. This will influence the cross-correlation.

3) Use of "non white" (band limited) signal.

   The use of the PRBS signal is stimulated by its autocorrelation function , which may approximate a Dirac function. Whether an approximation is fair enough depends on the situation at hand. As fine details in a transfer function are caused by the smallest time constant τ_{min} of the system under test, it appears that $\Delta t \approx \tau_{min}/5$ is an acceptable choice.

The amount of error however, depends on the situation viz. on $h(\tau)$ of the system.

4) Additive noise.

Due to the deterministic nature of the PRBS signals, there is no inherent statistical uncertainty. The study of the uncertainty due to additive noise shows that the variance in the estimate due to white additive noise is,

$$\sigma^2 = \frac{a^2\sigma_n^2}{i \cdot N \cdot \Delta t}$$

where σ_n^2 is the variance of the additive noise and i is the (integer) number of sequence periods over which correlation has been performed.

4. Sequential Least-squares Estimation.

The need for a recursive solution arises when fresh experimental data are continuously in supply and we wish to improve our parameter estimates by making use of this new information.

In this section, a recursive algorithm for the basic least-squares solution (2.9) is presented.

With the recursive formula, the estimates can be updated step by step without repeatedly computing the matrix solution of equation (2.9), in which the matrix inversion is quite time-consuming. This recursive solution procedure is often referred to as *sequential* , or *on-line estimation*. Recall that the vector equation

$$\bar{y} = \mathbf{X} \cdot \theta \tag{2.50}$$

consists of a set of N equations.

Let us introduce N as a subscript to $\bar{y}$ and $\mathbf{X}$ in equation (2.50).

We have

$$\bar{y}_N = \mathbf{X}_N \cdot \theta$$

Furthermore, denote $\hat{\theta}$ in equation (2.9) as $\hat{\theta}(N)$

$$\hat{\theta}(N) = \left(\mathbf{X}_N^T \mathbf{X}_N\right)^{-1} \cdot \mathbf{X}_N^T \cdot \bar{y}_N \tag{2.51}$$

Suppose we have obtained a new N+1-th equation

$$y(N+1) = \theta_1 x_1(N+1) + \theta_2 x_2(N+1) + \cdots + \theta_m x_m(N+1)$$

Define

$$\bar{x}(N+1) = \left[x_1(N+1)\ ,\ x_2(N+1)\ ,\ \cdots\ ,\ x_m(N+1)\right]$$

we then have

$$y(N+1) = \bar{x}(N+1) \cdot \theta$$

Now, the system of N+1 equations can be written as

$$\bar{y}_{N+1} = \mathbf{X}_{N+1} \cdot \theta \tag{2.52}$$

in which

$$\bar{y}_{N+1} = \begin{bmatrix} \bar{y}_N \\ y(N+1) \end{bmatrix} \tag{2.53}$$

$$\mathbf{X}_{N+1} = \begin{bmatrix} \mathbf{X}_N \\ \bar{x}(N+1) \end{bmatrix} \tag{2.54}$$

The new least-squares estimator is

$$\hat{\theta}(N+1) = \left(\mathbf{X}_{N+1}^T \mathbf{X}_{N+1}\right)^{-1} \cdot \mathbf{X}_{N+1}^T \cdot \bar{y}(N+1) \tag{2.55}$$

It is apparent that to obtain $\hat{\theta}(N+1)$, we must invert an mxm matrix. The obvious question here is whether or not we can calculate $\hat{\theta}(N+1)$ by simply updating the previous estimate $\hat{\theta}(N)$ without matrix inversion. The answer is yes and we derive the updating algorithm below.

Define the matrices P(N) and S(N) as

$$\mathbf{P}(N) = \mathbf{X}_N^T \mathbf{X}_N \tag{2.56}$$

$$\mathbf{S}(N) = \mathbf{X}_N^T \cdot \bar{y}_N \tag{2.57}$$

we can write

$$\mathbf{P}(N+1) = \left[\mathbf{X}_N^T \, , \, \bar{x}^T(N+1)\right] \cdot \begin{bmatrix} \mathbf{X}_N \\ \bar{x}(N+1) \end{bmatrix}$$

$$= \mathbf{P}(N) + \bar{x}^T(N+1) \cdot \bar{x}(N+1)$$

$$\mathbf{S}(N+1) = \left[\mathbf{X}_N^T \, , \, \bar{x}^T(N+1)\right] \cdot \begin{bmatrix} \bar{y}_N \\ y(N+1) \end{bmatrix}$$

$$= \mathbf{S}(N) + \bar{x}^T(N+1) \cdot y(N+1)$$

Substituting equation (2.54) and applying the inversion lemma [1] :

Inversion Lemma :

Let A, C and A+BCD be nonsingular square matrices .

Then, the following matrix identity holds

$$\left(A+BCD\right)^{-1} = A^{-1} - A^{-1}B\left(C^{-1}+DA^{-1}B\right)^{-1}DA^{-1}$$

$\mathbb{P}(N+1)$ can be rewritten as follows

$$\mathbb{P}^{-1}(N+1) = \left[\mathbb{P}(N) + \bar{x}^T(N+1)\cdot\bar{x}(N+1)\right]^{-1} =$$

$$= \mathbb{P}^{-1}(N) - \mathbb{P}^{-1}\cdot\bar{x}^T(N+1)\cdot\left[1 + \bar{x}(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\right]^{-1}\cdot\bar{x}(N+1)\cdot\mathbb{P}^{-1}(N)$$

or

$$\mathbb{P}^{-1}(N+1) = \mathbb{P}^{-1}(N) - \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\cdot\bar{x}(N+1)\cdot\mathbb{P}^{-1}(N) \qquad (2.59)$$

where

$$\gamma(N+1) = \frac{1}{\left(1 + \bar{x}(N+1)\cdot\mathbb{P}^{-1}\cdot\bar{x}^T(N+1)\right)} \qquad (2.60)$$

In view of equation (2.55) we can see that

$$\hat{\theta}(N+1) = \mathbb{P}^{-1}(N+1)\cdot S(N+1) =$$

$$= \left[I_N - \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\cdot\bar{x}(N+1)\right]\cdot\mathbb{P}^{-1}(N)\cdot\left[S(N) + \bar{x}^T(N+1)\cdot y(N+1)\right]$$

$$= \left[I_N - \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\cdot\bar{x}(N+1)\right]\cdot\left[\hat{\theta}(N) + \mathbb{P}^{-1}\cdot\bar{x}^T(N+1)\cdot y(N+1)\right]$$

$$= \hat{\theta}(N) + \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\cdot\left[y(N+1) - \bar{x}(N+1)\cdot\hat{\theta}(N)\right]$$

Let

$$e(N+1) = y(N+1) - \bar{x}(N+1)\cdot\hat{\theta}(N)$$

then

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\cdot e(N+1) \qquad (2.61)$$

The result above simply shows that the new estimate is given by the old estimate plus a correction term.

The matrix $\mathbf{P}(N)$ in the correction term can be updated by the recursive formula in equation (2.59). It is clear that in both formulas we have completely eliminated the necessity of matrix inversion (we note that $\gamma(N)$ is a scalar) and therefore the computational efficiency is drastically improved for updating the estimate $\hat{\theta}$.

The recursive equation (2.61) has a very strong intuitive appeal. The correction term is proportional to the error of fitting the previous estimate $\hat{\theta}(N)$ to the new data $y(N+1)$ and $\bar{x}(N+1)$. The fitting error is weighted in the correction of $\hat{\theta}(N)$ with a direct measure of the error covariance represented by the matrix $\mathbf{P}^{-1}(N)$.

In summary, we have obtained the following recursive algorithm for on-line identification :

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \gamma(N+1)\cdot\mathbf{P}^{-1}(N)\cdot\bar{x}^{T}(N+1)\cdot e(N+1) \tag{2.62}$$

$$e(N+1) = y(N+1) - \bar{x}(N+1)\cdot\hat{\theta}(N)$$

$$\gamma(N+1) = \left[1 + \bar{x}(N+1)\cdot\mathbf{P}^{-1}(N)\cdot\bar{x}^{T}(N+1)\right]^{-1}$$

$$\mathbf{P}^{-1}(N+1) = \mathbf{P}^{-1}(N) - \gamma(N+1)\cdot\mathbf{P}^{-1}(N)\cdot\bar{x}^{T}(N+1)\cdot\bar{x}(N+1)\cdot\mathbf{P}^{-1}(N)$$

Therefore, by starting with an initial estimate $\hat{\theta}(0)$ and the corresponding $\mathbf{P}^{-1}(0)$, we can sequentially update $\hat{\theta}$ while new observations are continuously obtained.

The initial values of $\hat{\theta}$ and $\mathbf{P}$ for starting the algorithm can be selected in two ways :

1) Take the first N data points and solve $\hat{\theta}(N)$ and $\mathbb{P}(N)$ directly from

$$\begin{cases} \hat{\theta}(N) = \mathbb{P}^{-1}(N) \cdot \mathbb{X}_N^T \cdot \bar{y}_N \\ \mathbb{P}(N) = \mathbb{X}_N^T \cdot \mathbb{X}_N \end{cases} \tag{2.63}$$

as in the off-line procedure.

2) Set $\hat{\theta}(0)$ arbitrarily and $\mathbb{P}^{-1}(0) = \alpha \cdot I$, where α is a very large positive scalar and I is the identity matrix. In this case, after N iterations we have

$$\begin{cases} \mathbb{P}^{-1}(N) = \left[\mathbb{P}(0) + \mathbb{X}_N^T \cdot \mathbb{X}_N\right]^{-1} \\ \hat{\theta}(N) = \mathbb{P}^{-1}(N) \cdot \left[\mathbb{X}_N^T \bar{y}_N + \mathbb{P}(0) \cdot \hat{\theta}(0)\right] \end{cases} \tag{2.64}$$

By selecting large α, we simply force $\mathbb{P}(0)$ to zero, making equations (2.63) agree with equations (2.62).

5. Real-time Least-squares Algorithm.

The recursive algorithm presented in the previous section implies that all N samples processed for the estimate $\hat{\theta}(N)$ get an equal weight, irrespective of the "age" of those samples. For the estimation of slowly changing parameters, this is obviously undesirable.

A sequential algorithm that is able to closely track time-varying parameters (called a *real-time algorithm)* needs a method to gradually "forget" the past.

In order to achieve this task, consider the error function

$$J(N) = \sum_{i=1}^{N} \lambda^{N-i} \cdot e^2(i) \tag{2.65}$$

where for $0 < \lambda < 1$ the later squared errors are given more weight than earlier ones.

This weighted error function can be obtained by weighting the observations according to their age :

$$\mathbb{X}_{N+1} = \begin{bmatrix} c\cdot\mathbb{X}_N \\ \bar{x}(N+1) \end{bmatrix}$$

$$\bar{y}_{N+1} = \begin{bmatrix} c\cdot\bar{y}_N \\ y(N+1) \end{bmatrix}$$

where $0 < c < 1$.

This gives the residual vector

$$\bar{e}_{N+1} = \bar{y}_{N+1} - \mathbb{X}_{N+1}\cdot\hat{\theta}(N+1)$$

$$= \begin{bmatrix} c\cdot\left(\bar{y}_N - \mathbb{X}_N\cdot\theta(N+1)\right) \\ e(N+1) \end{bmatrix}$$

$$= \left[c^{N+1}\cdot e(1)\ ,\ c^N\cdot e(2)\ ,\ \cdots\ ,\ c\cdot e(N)\ ,\ e(N+1)\right]^T$$

and the desired weighting function (2.65) when $c^2 = \lambda$.

For $c^2 = \lambda$ we have

$$\mathbb{P}(N+1) = \lambda\cdot\mathbb{P}(N) + \bar{x}^T(N+1)\cdot\bar{x}(N+1) \qquad (2.66)$$

$$\mathbb{S}(N+1) = \lambda\cdot\mathbb{S}(N) + \bar{x}^T(N+1)\cdot y(N+1) \qquad (2.67)$$

By the inversion lemma, we obtain

$$\mathbb{P}^{-1}(N+1) = \frac{1}{\lambda}\left[\mathbb{P}^{-1}(N) - \gamma(N+1)\mathbb{P}^{-1}(N)\bar{x}^T(N+1)\bar{x}(N+1)\mathbb{P}^{-1}(N)\right] \qquad (2.68)$$

where

$$\gamma(N+1) = \left[\lambda + \bar{x}(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^T(N+1)\right]^{-1} \qquad (2.69)$$

Finally, the LSE for the error function given by (2.65) is

$$\hat{\theta}(N+1) = \mathbb{P}^{-1}(N+1)\cdot S(N+1)$$

which by (2.66) - (2.69) can be written as

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \gamma(N+1)\cdot\mathbb{P}^{-1}(N)\cdot\bar{x}^{T}(N+1)\cdot e^{*}(N+1) \tag{2.70}$$

where

$$e^{*}(N+1) = y(N+1) - \bar{x}(N+1)\cdot\hat{\theta}(N) \tag{2.71}$$

The equations (2.68) - (2.71) give a complete algorithm for real-time estimation.

We observe that the smaller the λ, the heavier are the weights assigned to the more recent data. This implies that the algorithm is more capable of tracking the parameter variations. However, the estimates may also fluctuate more because of noise disturbance. Therefore, λ is often chosen experimentally for a given task. In general, λ is chosen close to unity.

CHAPTER 3.

Linear Network Modelling.

The system identification techniques can be practically used to reduce electrical network complexity for computer simulation purposes. It will be shown in this chapter, how an active network containing a large number of elements can be represented by a single voltage source and a series impedance, that is by its Thevenin equivalent network.

A model of such an equivalent network will be derived based on the least-squares identification technique which was discussed in chapter 2.

A system identification approach to the network modelling problem allows us to avoid the detailed study of internal reletionships that for large networks may often be very complex.

When using a standard circuit approach to the modelling problem, network elements or branches are characterized by equations (called *constitutive relations*) relating branch voltages and currents.

Using either the state or node method, it is straightforward to derive a set of differential equations describing network dynamics. The number of equations, consequently, may be very large for some complex networks.

There are, however, a number of useful methods that can simplify network description. In many cases a part of a network can be removed, and something simpler can take its place. A list of simplifying techniques includes : equivalent series and parallel impedances, voltage- and current-divider formulas, superposition, Thevenin and Norton equivalents, ladder methods, etc.

If a part of a network is connected to the rest of the network at one pair of terminals, it can affect the rest of the network only by the voltage and current at

those terminals. Thus, such a subnetwork can be replaced by another *equivalent subnetwork* with the same relations between the voltage and current at the pair of terminals and this exchange does not affect conditions in any part of the network external to it.

In the next section, the derivation of the Thevenin equivalent network is presented.

1. Thevenin Equivalent Network.

A very convenient method of obtaining a simplified subnetwork which is equivalent to a section of a complete network is given by Thevenin's Theorem. This theorem might be formulated as follows.

The section of a network to be modelled or simplified will allways have at least one source and a pair of terminals at which it is connected to the remainder of the complete network. This is shown symbolically in Fig.3.1

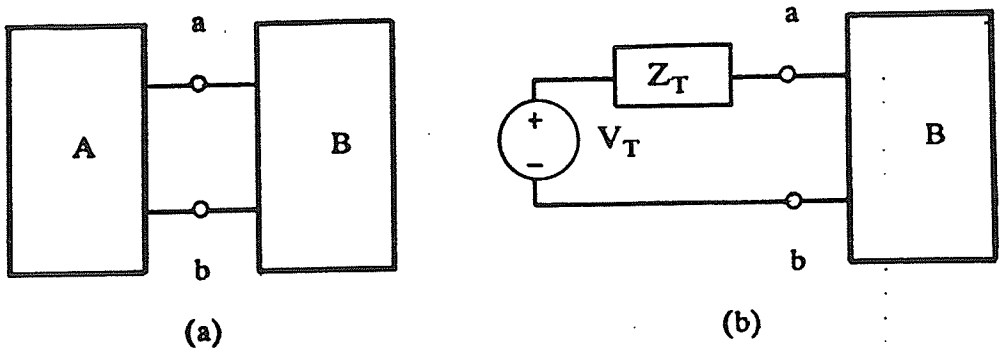


Fig.3.1 (a) A Network Divided into Two Parts A and B . (b) Part A of the Network Replaced by Its Thevenin Equivalent.

Thevenin's Theorem states simply that the network A shown in Fig.3.1(a) can be represented by voltage source V_T and a series impedance Z_T as shown in Fig.3.1(b).

The Thevenin equivalent network will produce exactly the same current- voltage relationship at the terminals a and b provided :

(1) V_T is the voltage between a and b when i=0 (i.e, the open circuit voltage).

(2) Z_T is the impedance between terminals a and b when all the sources within the network are reduced to zero by short-circuiting voltage sources and open-circuiting current sources.

1.1. Thevenin Equivalent of the Transformed Network.

For modelling purposes, network voltages and currents are considered as signals. Since our task is to construct a discrete model of the network, the signals are therefore indexed sequences of sampled values - discrete-time (DT) signals. The sample values are taken with a fixed sampling period T

$$x(n) \doteq x(n \cdot T)$$

Applying a unilateral Z-transform to the DT signals of the network

$$X(z) = \sum_{n=0}^{\infty} x(n) \cdot z^{-n}$$

we obtain the *transform network* in which :

1. All signals are represented by the corresponding transformed currents

$$I(z) = \sum_{n=0}^{\infty} i(n) \cdot z^{-n}$$

and transformed voltages

$$V(z) = \sum_{n=0}^{\infty} v(n) \cdot z^{-n}$$

2. For a single element of the network, the transformed impedance is defined as the ratio of the transformed voltage to the transformed current

$$Z(z) = \frac{V(z)}{I(z)}$$

for zero initial current in an inductor and zero initial voltage in a capacitor.

3. Initial conditions are represented by transformed voltage or current sources.

Applying Thevenin's Theorem to the transformed network we obtain an equivalent network.

The *Thevenin equivalent network* consists of a voltage source V_T and a series impedance Z_T.

Voltage source V_T represents the transformed voltage at the open terminals and impedance Z_T represents the transformed impedance at the network terminals when all independent sources are reduced to zero.

The terminal voltage for the Thevenin equivalent network can be expressed as a difference between the Thevenin source voltage V_T and the voltage drop at the series impedance Z_T .

$$V_o(z) = V_T(z) - I(z) \cdot Z_T(z) \qquad (3.1)$$

$$= V_T(z) - V_Z(z)$$

Thus, in order to generate this voltage for a simulation purpose, we need two models :

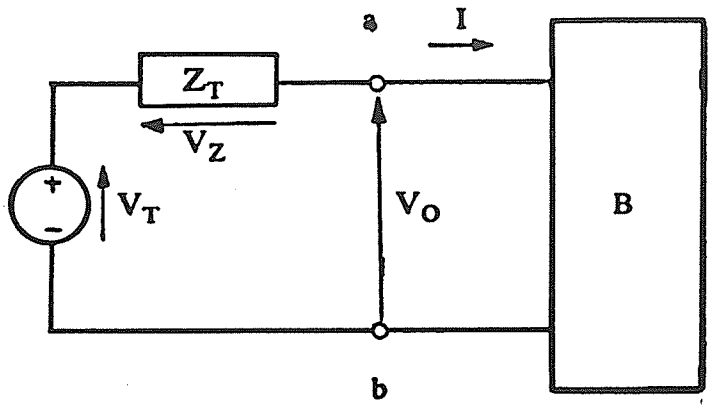


Fig.3.2 General Form of the Thevenin's Theorem.

1) A model of the Thevenin voltage source providing successive values of $v_T(k)$ k=1,2,3, ...

2) A model of the network impedance Z_T which together with the external network B will limit the current at the output terminals.

Both of these models can be determined using the system identification techniques discussed in chapter 2.

A. Thevenin Voltage Source Model.

Thevenin's Theorem says that the voltage V_T is an open-circuit voltage at the output terminals of the network containing at least one voltage source. We might look at the network as for the linear system having a source voltage (or voltages) V_S as its input and an open-circuit voltage V_T as its output.

The mathematical relationship between v_S and v_T can be expressed in terms of a transfer function of the system in Fig.3.3.

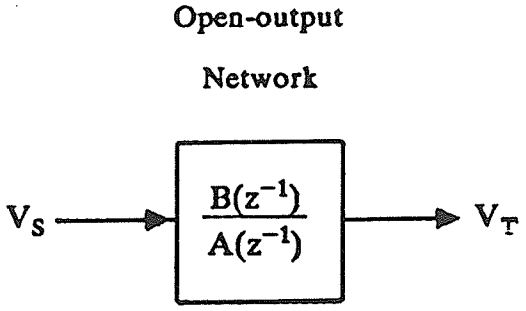


Fig.3.3 Thevenin Voltage Source Modelling Problem.

$$\frac{V_T(z)}{V_S(z)} = \frac{B(z^{-1})}{A(z^{-1})} \tag{3.2}$$

where A and B are polynomials of some order m

$$A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_m z^{-m} \tag{3.3}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}$$

or by the corresponding shifting-operator form

$$A(q^{-1}) v_T(k) = B(q^{-1}) v_S(k) \tag{3.4}$$

Thus, the Thevenin source voltage can be filtered out from $v_S(k)$ by

$$v_T(k) = \frac{B(q^{-1})}{A(q^{-1})} v_S(k) \tag{3.5}$$

To determine the parameters a_j and b_j, we can use least-square identification technique discussed in chapter 1.

B. **Series Impedance Model.**

To find a model of the series impedance, we consider a network with all its independent sources reduced to zero.

The model we are looking for is the mathematical relationship between the current I and the voltage V_Z at the output terminals of this passive network.

In this case, we look at the network as for the linear system with current I as its input and terminal voltage V_Z as its output.

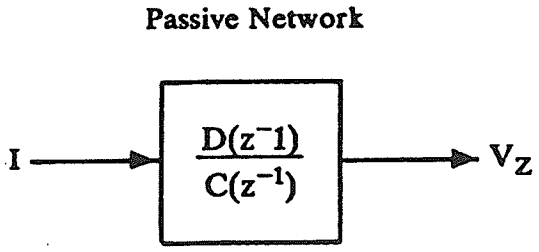


Fig.3.4 Network Impedance Modelling Problem.

The relationship between I and V_Z can be expressed in terms of the transfer function of the system in Fig.2.4

$$\frac{V_Z(z)}{I(z)} = \frac{D(z^{-1})}{C(z^{-1})} \tag{3.6}$$

where C and D are polynomials of some order n

$$C(z^{-1}) = 1 + c_1 z^{-1} + \cdots + c_n z^{-n} \tag{3.7}$$

$$D(z^{-1}) = d_0 + d_1 z^{-1} + \cdots + d_n z^{-n}$$

or in the corresponding shifting-operator form

$$C(q^{-1})\cdot v_Z(k) = D(q^{-1})\cdot i(k) \tag{3.8}$$

For a given current at the output terminals, the voltage drop at the network impedance Z_T can be filtered out by

$$v_Z(k) = \frac{D(q^{-1})}{C(q^{-1})}\cdot i(k) \tag{3.9}$$

As before, parameters c_j and d_j can be determined using least-squares identification techniques.

Equations (3.5) and (3.9) with estimated parameters of equations (3.3) and (3.7) give a complete model of the Thevenin equivalent network.

We observe that equations (3.5) and (3.9) can be written in a form of difference equations describing the dynamics of active and passive configurations of the network.

$$\sum_{j=0}^{m} a_j\cdot v_T(k-j) = \sum_{j=0}^{m} b_j\cdot v_S(k-j) \quad \text{with } a_0 = 1 \tag{3.10}$$

$$\sum_{j=0}^{n} c_j\cdot v_Z(k-j) = \sum_{j=0}^{n} d_j\cdot i(k-j) \quad \text{with } c_0 = 1 \tag{3.11}$$

From these two difference equations, we can evaluate successive values of the two voltages associated with the equivalent model :

1) Thevenin source voltage :

$$v_T(k) = \sum_{j=0}^{m} b_j\cdot v_S(k-j) - \sum_{j=1}^{m} a_j\cdot v_T(k-j) \tag{3.12}$$

2) Voltage drop at the series impedance :

$$v_Z(k) = \sum_{j=0}^{n} d_j\cdot i(k-j) - \sum_{j=1}^{n} c_j\cdot v_Z(k-j) \tag{3.13}$$

Having all unknown parameters estimated by the identification procedure, we can evaluate $v_T(k)$ and $v_Z(k)$ if :

1. Initial and successive values of the network source voltage $v_S(k)$, k=0,1,2,3, $\cdots$ are given.
2. The current i at the output terminal of the network is measurable (or computable).

The technical problem appears in evaluation of v_Z where for any instant of time k, we need to know the value of the current i(k). This, however, cannot be determined until we define $v_o(k)$.

By manipulating (3.13), we can extract from its left-hand side that term which depends on the present value of the current and we can then gather the remining terms into one, which only depends on the past values of i and v_Z.

$$v_Z(k) = d_0 \cdot i(k) + \left[\sum_{j=1}^{n} d_j \cdot i(k-j) - \sum_{j=1}^{n} c_j \cdot v_Z(k-j) \right] \tag{3.14}$$

$$= d_0 \cdot i(k) + v_Z(\text{past})$$

The first term on the left-hand side can be interpreted as a voltage drop at the series impedance due to the current value of i at the output terminals. That portion of $v_Z(k)$ can be modelled by a series impedance $Z_0 = d_0$ causing the desired voltage drop $d_0 \cdot i(k)$.

The remaining part of $v_Z(k)$ can be modelled as a series voltage source reducing the Thevenin voltage $v_T(k)$ by a factor of $v_Z(\text{past})$

A complete Thevenin equivalent model after modification is shown in Fig.3.5 where

$$Z_0 = d_0$$

$$v_d(k) = -v_Z(\text{past}) = \sum_{j=1}^{n} c_j v_Z(k-j) - \sum_{j=1}^{n} d_j i(k-j) \tag{3.15}$$

$$v_T(k) = \sum_{j=0}^{m} b_j v_Z(k-j) - \sum_{j=1}^{m} a_j v_T(k-j)$$

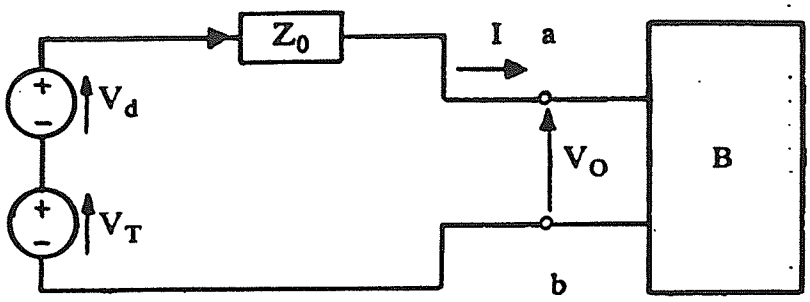


Fig.3.5 Thevenin Equivalent Network Model.

In the above analysis we considered an active network with one source. When the network contains more than one source, the series impedance model remains the same but the model of the Thevenin source must be modified.

This can be done either by use of the multiple-input identification technique discussed in section 1 of chapter 2 or by superposition of the Thevenin voltages V_T^i obtained by the application of the modelling procedure A to the network with active source V_S^i and all the other sources reduced to zero.

Both of these methods are very inconvenient when the number of sources in the network is large. In the first case we have to deal with coefficient matrices of a high dimension (as in 2.4) or in the second case the identification procedure must be repeated many times for each network source separately.

Very often, however, when the source is operating at one frequency it is reasonable to care only about the model that is suitable for this particular mode. In this case we can avoid a complicated procedure for modelling Thevenin source even if the number of sources in the network is very large.

In the next section we show how the Thevenin source model can be obtained when all the network sources are generating the same frequency signals.

2. Networks with Single Frequency Sinusoidal Sources.

When the network sources generate only single frequency sinusoidal signals, then by linearity, the Thevenin voltage source in the model will also generate a signal of the same frequency. The parameters to be determined are the amplitude and the phase shift.

To find these two parameters we assume that the series impedance had been found as

$$Z_T(z) = \frac{V_Z(z)}{I(z)} = \frac{D(z^{-1})}{C(z^{-1})}$$

where C and D are polynomials with the coeffitients estimated by the identification procedure B.

Since the following relationship between Z-transform and Laplace Transform complex variables holds

$$z = e^{sT} \quad \text{where T is a sampling period}$$

or for a given frequency f of the source signals

$$z = e^{j\omega T} \quad \text{where } \omega = 2\pi f \tag{3.16}$$

we can compute the series impedance Z_T as follows

$$Z_T(\omega) = \frac{D(e^{-j\omega T})}{C(e-j\omega T)} \tag{3.17}$$

$$= \frac{b_0 + \sum_{k=1}^{n} b_k \cdot e^{-j\omega kT}}{1 + \sum_{k=1}^{n} c_k \cdot e^{-j\omega kT}}$$

$$= \alpha + j\cdot\beta$$

If at the output terminals we connect the impedance,

$$Z_L(\omega) = \gamma + j\cdot\lambda$$

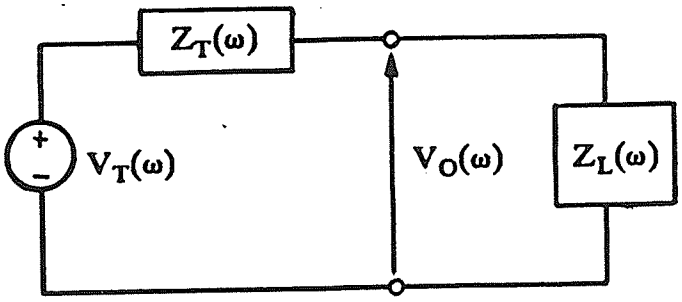


Fig.3.6 Thevenin Equivalent Network for Single Frequency Signals.

the Thevenin source voltage can be determined from the following relationship :

$$V_T(\omega) = \frac{Z_T(\omega) + Z_L(\omega)}{Z_L(\omega)} \cdot V_O(\omega) \tag{3.18}$$

$$= G(\omega)\cdot V_O(\omega)$$

Thus, by a simple measurement of the output voltage for a given load, we can determine the amplitude and the phase shift for the Thevenin source voltage. This can be done as follows.

Suppose that we observe the output voltage

$$V_O(t) = A\cdot\sin(\omega t) \tag{3.19}$$

Then, since $G(\omega)$ can be calculated as

$$G(\omega) = \frac{(\alpha + j\beta) + (\gamma + j\lambda)}{(\gamma + j\lambda)} \tag{3.20}$$

$$= \frac{(\alpha + \gamma) + j(\beta + \lambda)}{(\gamma + j\lambda)}$$

$$= \left[\frac{(\alpha + \gamma)\cdot\gamma + (\beta + \lambda)\cdot\lambda}{(\gamma^2+\lambda^2)}\right] + j\cdot\left[\frac{(\beta\cdot\gamma - \alpha\cdot\lambda)}{(\gamma^2+\lambda^2)}\right]$$

$$= a + j\cdot b$$

the parameters of the Thevenin voltage V_T

$$V_T(t) = B\cdot\sin(\omega t + \theta) \tag{3.21}$$

can be computed as follows

$$B = A\cdot\sqrt{a^2 + b^2} \tag{3.22}$$

$$\theta = \tan^{-1}\left(\frac{b}{a}\right) \tag{3.23}$$

The Thevenin source voltage determined in such a way represents the

cummulative effect of all sources present in the network.

Finally, the Thevenin equivalent model for a given frequency f is constructed with two fixed (or frequency dependent in general) elements :

1) A voltage source that is determined from the measurement of the output voltage V_O when the impedance Z_L is conected at the output terminals by the relations (3.21) - (3.23).

2) A series impedance that is calculated by (3.18) from the impedance model provided by the identification procedure applied to the passive network.

CHAPTER 4.

Equivalent Model Building Procedure. A Numerical Example.

A simple network will be modelled to illustrate how the procedure developed in the previous chapters is practically realized.

A complete procedure that leads us to the operation on the Thevenin equivalent model is carried out in the following steps :

1) Active and passive configurations of the network are modelled in the the EMTDC simulation program to produce the necessary observations for the identification procedure as required by procedures A and B in section 1 of chapter 3.

2) Using the STPRES program we produce step responses of the two subsystems and then determine a sampling period of the observed signals.

3) The PRBS test signals are supplied as voltage and current sources for both subsystems.

4) We gather the appropriate input-output observations and process them using the least-squares identification algorithms to obtain estimates of the unknown model parameters. This is done in two steps :

   a) First 200 observations are processed in a one-shot way using the recursive algorithm for off-line identification (as in section 2.3 of chapter 2) carried out by the PREL program.

   b) Then 800 observations are processed in a step-by-step way using the recursive algorithm for on-line identification (as in section 4 of chapter 2) carried out by the ONLINE program.

5) The last values of parameter estimates and the sampling period are stored in the data file.

6) For a simulation purpose, we read a model data file and produce the Thevenin voltage and the series impedance voltage drop as follows :

   a) we place the series impedance $Z_0=d_0$ in series with the voltage source (as in Fig.3.5).

   b) the source voltages are computed by (3.15).

   c) for each iteration we produce a current value of the network source voltage and measure the output current as requested by (3.15).

7) The simulation results can be compared with the results from the EMTDC program.

We consider the circuit of Fig.4.1. with the following numerical values :

$$R_1 = R_2 = 1\ k\Omega$$

$$L_1 = L_2 = 0.1\ H$$

$$C = 0.2\ \mu F$$

The Thevenin equivalent model contains two parts : a voltage source and a series impedance. In order to find the two components of the model, we observe the network of Fig.4.1 in two configurations :

1) Network with open output terminals for the Thevenin source model (Fig.4.2).

2) Network with short-circuited sources for the series impedance model (Fig.4.3).

As indicated in chapter 3, models are formed by linear difference equations and their coefficients can be found using the least-squares identification technique.

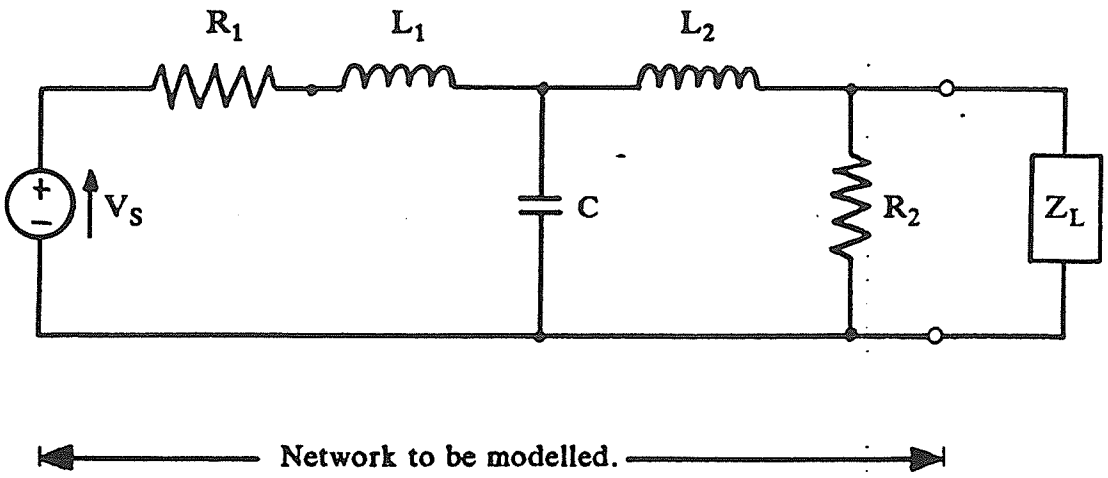


Fig.4.1. Circuit for Numerical Example.

The identification procedure requires a sequence of observations that is processed to obtain the estimates of the unknown model parameters.

For our study, instead of observing a real network, we take measurements from a high accuracy simulation program. This program is the Electromagnetic Transient Program EMTDC developed at Manitoba Hydro by D.A. Woodford [11]. In this program, all fixed elements of the network are specified as program data and varying components such as sources are called from the dynamics subroutine DSDYN. Any current or voltage within the network can be computed at any time which changes gradually with a specified time step DELT. All variables that are selected in the subroutine DSOUT can be outputted and plotted.

In order to obtain the required observations for the identification procedures, we enter the network data into an EMTDC data file called THVDA, as two seperate subsystems.

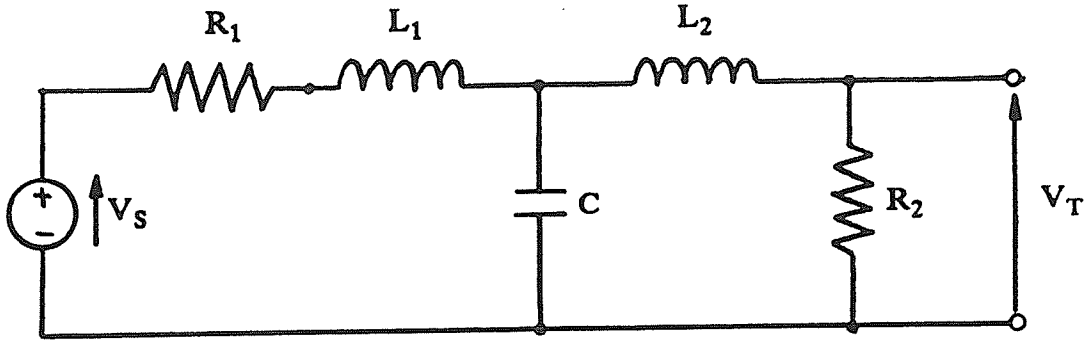


Fig.4.2.Open-output Network for the Thevenin Source Model.

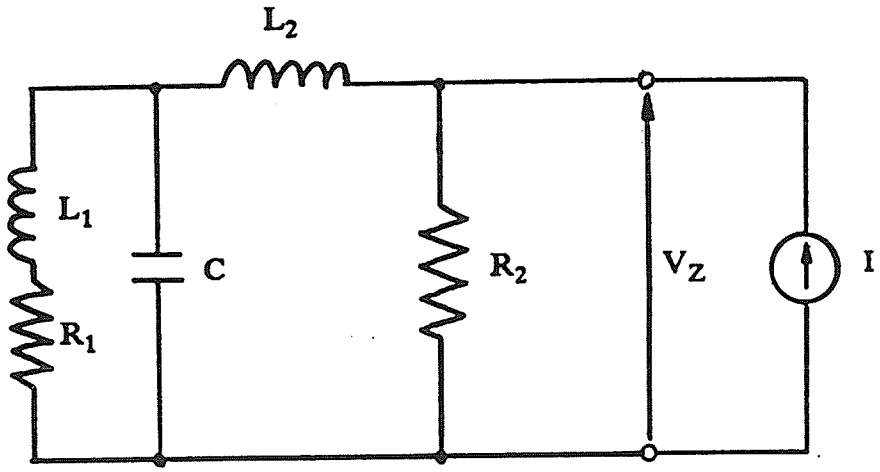


Fig.4.3.Network with Short-circuited Voltage Source for the Series Impedance Model.

One subsystem is the open-output network of Fig.4.2 that provides the measurements of the network source voltage V_S and the open-output voltage V_T used for the voltage source modelling.

The second subsystem is the network with a short-circuited voltage source as in Fig.4.3 which provides measurements of the injected current I and the voltage V_Z at the output terminals used for the impedance model.

The subsystem sources are generating test signals when they are observed.

It was shown in section 3 of chapter 2 that the optimum test signal is practically realized by the PRBS signal. Thus, the voltage source for the first subsystem and the current source for the second subsystem are generating PRBS signals. Parameters of these signals, such as amplitude or the minimum swiching time, can be adjusted from the EMTDC data file where the array VAR is used to assign the desired values for the program variables.

In our experiments, the minimum swiching time of the PRBS signal has the same value as the sampling period Δt and it is a multiple of the EMTDC time step DELT. The multiplier is entered by the user after he decides what time step is required for the equivalent model and whether it is reasonable to test the network with such a value.

The efficiency as well as the values of the estimates depend on this variable, thus it should be chosen very cautiously. Usually we try to maximize the time step in order to save the computation effort when using the model for a simulation purpose. However, we can not go too far since the longer time step does not allow us to properly observe the network dynamics and this consequently causes a very inaccurate estimation.

As an aid in the determination of the optimum value for this important factor we can use the network step response which is supplied by the program STPRES. From the step response plots, we observe the network time constants and then we

choose the time step with respect to them. The choice of 1/5 of the minimum time constant for both subsystems usually assures good estimation results.

The subsystem step responses are shown in Figs.4.4 and 4.5 .

A sampling period of 50μs was chosen.

From the EMTDC simulation program, we obtain two sets of observations :

$$1) \quad \left\{ \left(v_S(k), v_T(k) \right) ; k = 1, 2, 3, \cdots , N \right\} \tag{4.1}$$

which is used to determine the unknown coefficient of the Thevenin source model equation :

$$\sum_{j=0}^{n} a_j \cdot v_T(k-j) = \sum_{j=0}^{n} b_j \cdot v_S(k-j) \qquad \text{with } a_0 = 1 \tag{4.2}$$

$$2) \quad \left\{ \left(i(k), v_Z(k) \right) ; k = 1, 2, 3, \cdots , M \right\} \tag{4.3}$$

which is used to determine the unknown coefficients of the impedance model :

$$\sum_{j=0}^{m} c_j \cdot v_Z(k-j) = \sum_{j=0}^{m} d_j \cdot i(k-j) \qquad \text{with } c_0 = 1 \tag{4.4}$$

When the orders of the dynamic equations n and m are unknown, then these parameters must also be determined in the identification procedure.

The least-squares identification procedure described in chapter 2 can be used to find the parameter estimates and the optimum model orders. Two computer programs are created to carry out the off-line and on-line recursive algorithms derived in sections 2 and 4 of chapter 2. These are the PREL and ONLINE programs. They are composed into the DSDYN and DSOUT subroutines of the EMTDC program and are used in the following manner.

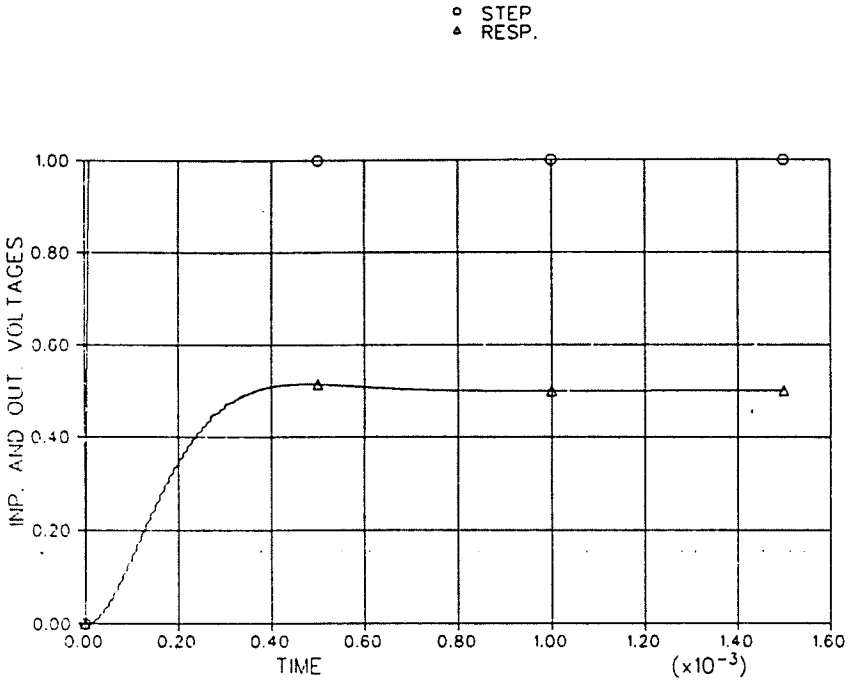


Fig.4.4 Step Response of The Active Network of Fig.4.2.

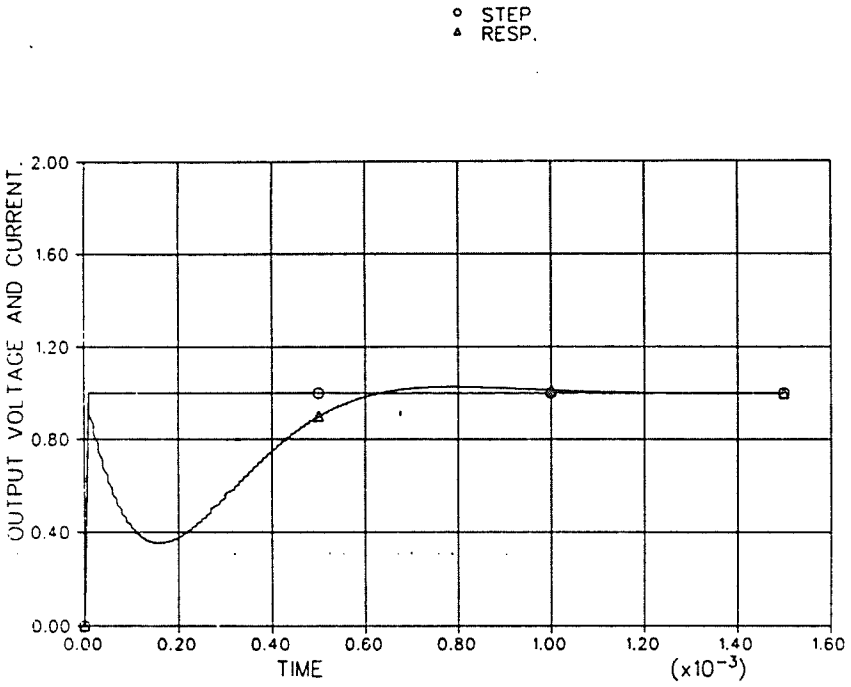


Fig.4.5 Step Response of The Passive Network of Fig.4.3.

First none of these programs are used until the initial number of observations are gathered into sequences as in (4.1) and (4.3). Then, the PREL program processes the sequence of observations to produce parameter estimates. This is done recursively by increasing model orders by one until the desired orders of the model equations or the automatically determined optimum orders are reached.

After the initial estimates and model orders are determined, the pairs of input-output observations are used step by step in an on-line procedure carried out by the ONLINE program. This is to improve the initial estimates and also to give an illustrative performance of the estimate behavior when the number of observations increase.

The identification procedure ends when either the desired number of observations to be processed or the EMTDC finish time is reached.

Since we are operating on two seperate subsystems, the identification for the two components of the Thevenin equivalent model can be parallely processed.

The estimation results for 50μs second order model are presented in Figs.4.6 and 4.7.

The complete Thevenin equivalent model can be stored as a simple array of parameter values :

Δt

$n , b_0 , a_1 , b_1 , \cdots , a_n , b_n$

$m , d_0 , c_1 , d_1 , \cdots , c_m , d_m$

When some of the signals are not measured in basic units, then the scaling factor should be added in the first line of the stored data. For example, if in the network of Fig.4.1 the current was measured in mA, then the scaling factor 1000 should be included to the stored data.

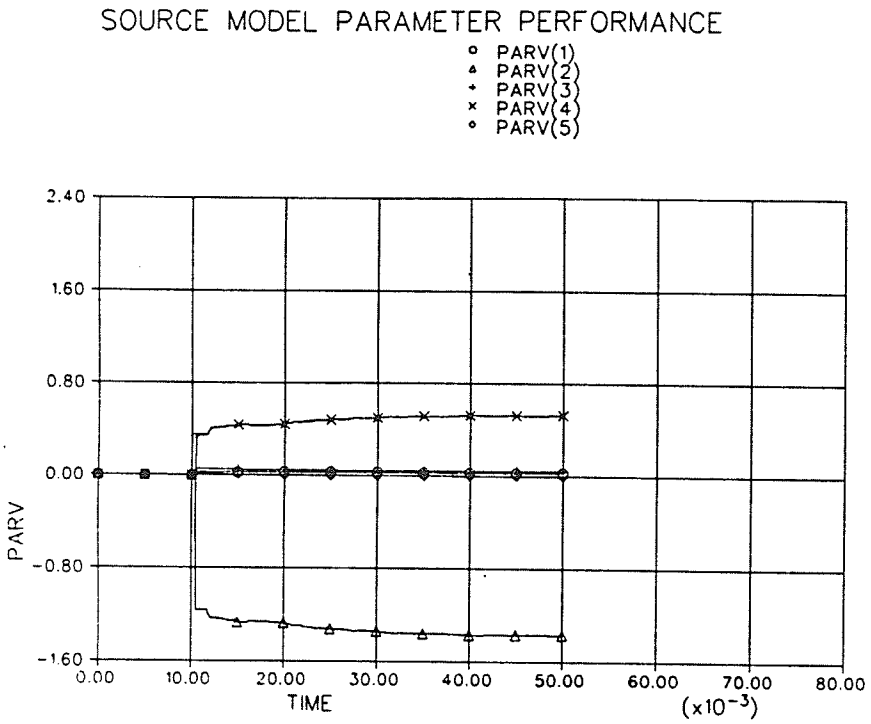


Fig.4.6 Parameter Performance of The Thevenine Source Model.

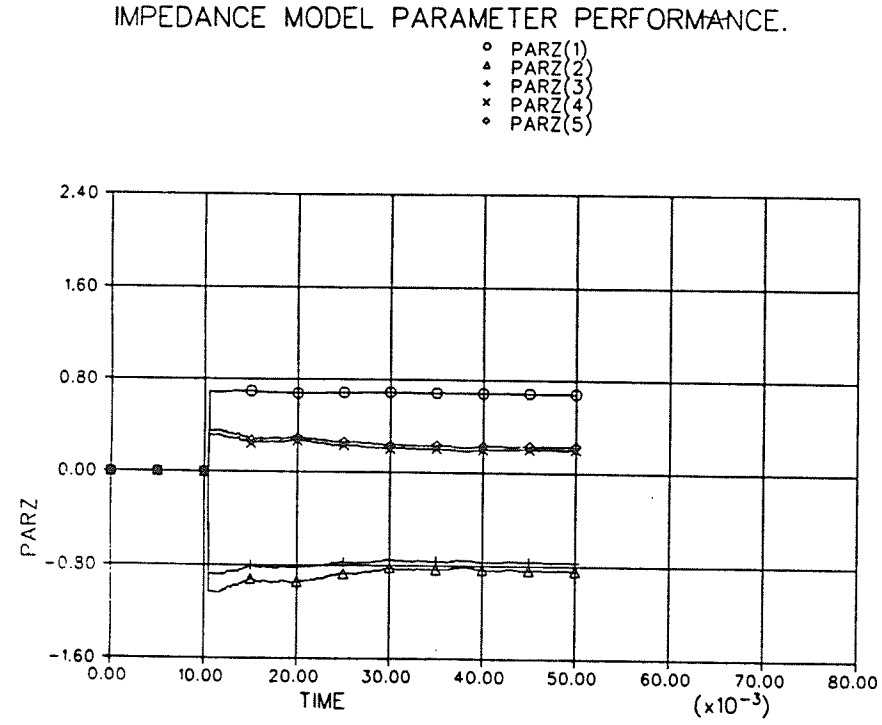


Fig.4.7 Parameter Performance of The Series Impedance Model.

Below, we present the estimation results for 5 different sampling periods $\Delta t = 10 , 20 , 30 , 50 , 100 \mu s$.

The parameters of each model are stored in the same order as the symbols in the above box.

MOD10 :

```
1.E-05  1000.
2  9.48346E-04  -1.06476  9.8915E-03  9.0456E-02  1.24329E-03
2  .914293  -1.54254  -1.48807  .576725  .583016
```

MOD20 :

```
2.E-05  1000.
2  7.13061E-03  -1.17616  9.43866E-03  .243526  1.74522E-02
2  .833536  -1.48688  -1.37045  .553718  .565405
```

MOD30 :

```
3.E-05  1000.
2  1.33383E-02  -1.21801  1.84712E-02  .326519  2.40354E-02
2  .809953  -.801947  -.813715  -2.30095E-02  0.E+00
```

MOD50 :

```
5.E-05  1000.
2  2.76794E-02  -1.21826  4.66063E-02  .397614  1.42771E-02
2  .728765  -.865751  -.81445  .192535  .258946
```

MOD100 :

```
1.E-04  1000.
2  6.37189E-02  -1.09862  9.48088E-02  .434375  -1.1233E-02
2  .736961  -.381786  -.544669  -3.18238E-02  .275296
```

In order to judge the efficiency of different network models, we produce the output voltage using equivalent model and compare it with the output voltage of the observed network (this is with the voltage provided by the EMTDC simulation program) for the same network source voltage V_S in each case.

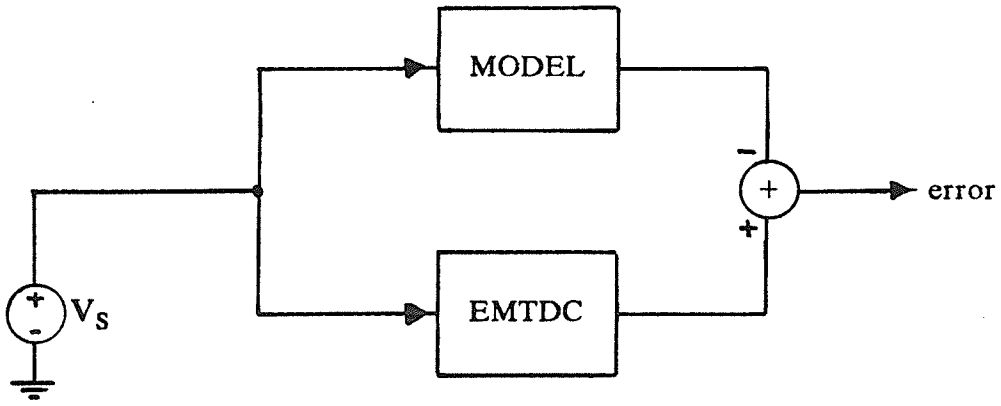
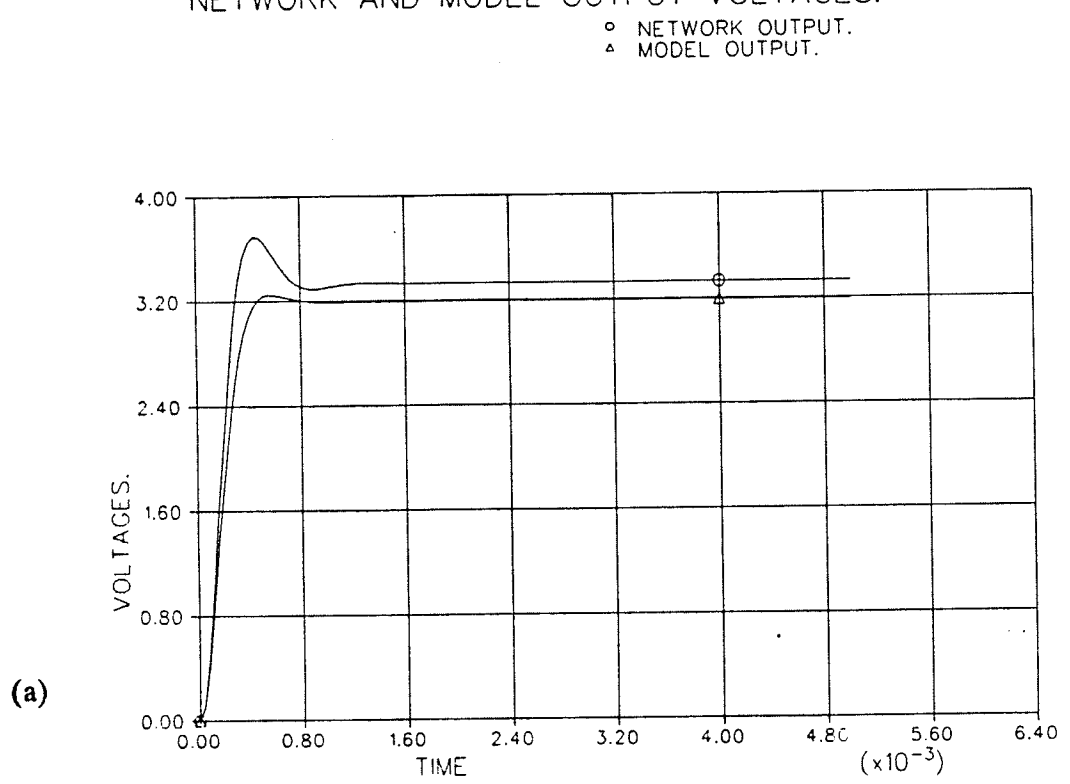


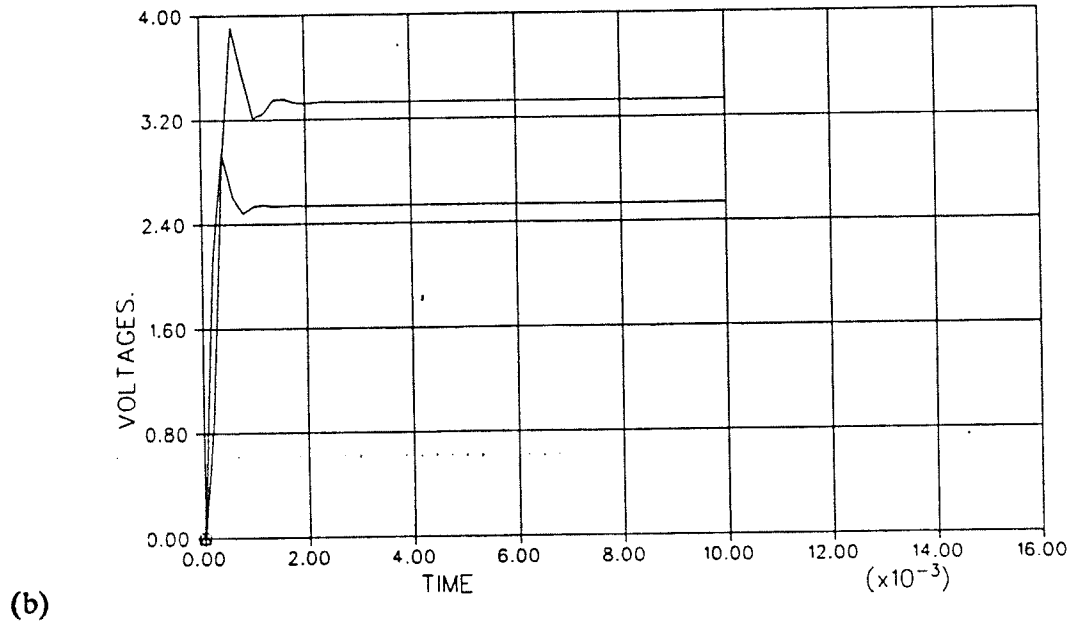
Fig.4.8. Model Efficiency Test.

In Fig.4.9 output voltages produced by the 50μs and 200μs models are compared with the corresponding output voltages produced by the EMTDC program when a 10V step function was used as a source voltage.

We can see that the time step of 200μs is to large and thus the simulation results are not as good as for the 50μs.



(a)



(b)

Fig.4.9 Model and EMTDC Output Voltages.(a) 50μs model. (b) 200μs model.

Results of more extensive tests of the network modeller can be found in [12].

Computer programs which were created for the modelling and simulation purposes are packaged in two separate packages :

A. **The Network Modelling Package** containing the following programs :

1) STPRES - program that provides step responses of the network in its active and passive configurations. From these preliminary observations the sampling period and the parameters of the test signal should be chosen.

2) PRBS - program for the pseudo-random binary test signal generation. It is used as a voltage or current source in the identification procedure.

3) PREL - program that executes the off-line least-square identification procedure which is the recursive procedure for increasing number of paramiters and does not involve matrix inversions. The sequences of input-output observations are processed to obtain the estimates of unknown model parameters. The model order is either determined automatically or it is arbitrary given by user. The model is created for a chosen timestep which is equal to the sampling period that the observations were taken from the network.

4) ONLINE - a program that executes the on-line identification procedure. The new observations that are continuously in supply are processed step by step by the recursive procedure that updates and improves the estimates of unknown parameters.

5) PRELF and ONLINEF - programs that are the modified versions of PREL and ONLINE respectively. They are intended to be used when the observations are contaminated with measurement noise. To eliminate estimate bias that we can encounter in this case, the filtering error technique is applied.

6) SFM - program for modelling networks containing a large number of single frequency sinusoidal sources. The Thevenin source is modelled as a fixed sinusoidal voltage source and the series impedance is modelled as before by alinear difference equation.

B. **The Simulation Package** containing the following programs

1) THVMOD - a program that simulates the voltage-current relationship at the output terminals using the Thevenin equivalent model.

2) THVMSF - an alternative version of the THVMOD program that is intended for single frequency models.

The detailed description of these programs and their listings can be found in [12]. Also provided are guidelines and a user manual for the use of both packages with the EMTDC program [11].

CHAPTER 5.

CONCLUSIONS AND RECOMENDATIONS.

The major task of this work was to develop the technique for simplifying the large scale linear networks. It was achieved by generalization of Thevenin's Theorem which states that any active linear network can be substituted by a simple equivalent network containing only a voltage source and a series impedance. The generalization is carried out by use of dynamic models for the two components of the equivalent network instead of modelling them as fixed elements. Use of the dynamic models provides universality of such Thevenin equivalent and makes it possible to be applied for a wide range of frequencies and network modes.

Since network models are primarily intended for computer simulation, therefore, in our approach the voltage source and the series impedance are represented by the discrete models. The form of these models was choosen to be a linear difference equation as it has close coherence with the network transfer function.

The Thevenin equivalent model contains two such linear equations. In order to find the model, we need to determine order and coefficient of both model equations.

1. CONCLUSIONS.

1) We have shown that the least-square identification technique can be used to obtain the statistical estimates of unknown model parameters and to determine the optimum order. The system identification approach to the modelling problem allows us to find the input-output relationship, describing

network dynamics, without detailed study of network branch and node equations.

In spite of the fact that we need to gather and process a number of observations, the whole procedure seems to be simple and once the model is created it can be easily stored and used.

Since the model is represented by standard linear difference equations we have to store only a set of the following parameters :

1. Sampling period which is also the timestep for simulation.
2. Source model parameters : order and a set of equation coefficients.
3. Impedance model parameters : order and a set of equation coefficients.

When using such a model for simulation,we simply substitiute the equation coefficients by the data from the model storage file and compute the Thevenin voltage and the voltage drop at the series impedance from the model equations. In each iteration (that is repeated with the time step equal to the sampling period) the output voltage is computed as a difference between the Thevenin voltage and the voltage drop.

2) It is to be pointed out that in the modelling procedures we can select model order and the timestep of the reduced model. This means that even if the order is high or the network is not quite linear, the best least-squared-error representation of a desired order and for a chosen timestep can be provided. In this way the order can be reduced and the timestep increased which results in a smaller computation time.

3) The method employed by many engineers today is to use a Thevenin reduction in terms of physical elements as in Fig.5.1 [13].

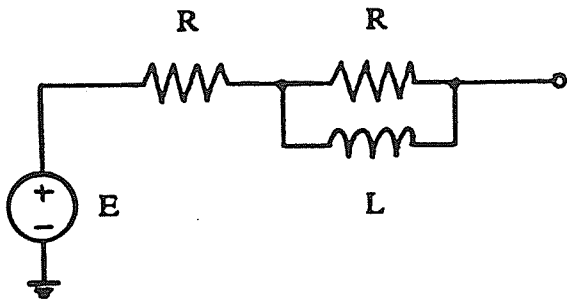


Fig.5.1 Thevenin Equivalent Model in Terms of Physical Elements.

In order to adequately model network dynamics, we often have to use more sophisticated combinations of RLC elements and then the determination of the optimum values for these elements becomes very difficult. The difference equations that are used in the proposed model reduce system to the form that is directly applicable on a computer without finding real elements of the equivalent model.

4) For the modelling and simulation purposes two packages of computer programs are created :

A. The Network Modelling Package, and

B. The Simulation Package.

The modelling package contains programs which carry out the off-line, on-line, and the real-time identification procedures. These procedures are based on the recursive algorithms which do not involve matrix inversion and consequently save computation time when dealing with high

order models.

It also contains complete tools for testing network such as : the PRBS test signal generator, step-response monitoring program, white noise generator and all data files for entrance the network under test into the EMTDC program.

The main program of the simulation package is meant to be a subroutine for EMTDC that substitutes network subsystem by the Thevenin equivalent model.

5) The parameter estimates suffer from bias when measurement noise is present. In order to eliminate this bias the identification procedure is supported with filtering error technique. In this case input-output observation are filtered and the fitting error is calculated based on these filtered signals.

2. RECOMENDATIONS.

Initially the modeller was designed for the network with one source but it is still under development and the following versions are being prepared :

1) The modeller for network with more then one sinusoidal source based on the procedure described in section 2 of chapter 3.
2) The modeller for three-phase network based on the multiple-input and multiple output scheme discussed in section 1 of chapter 2.

The existing version of the modeller can be applied to the real time modelling problem. This is made possible by use of a weighted error function which lets us gradually forget the past and use the more recent data to update the parameter estimates according to the present stage of the network.

It implies that this method can be used to track time-varying model parameters. Furthermore, since values of the model parameters depend on the stage of the network, the parameter vector estimated in real time can be used to monitor the network conditions.

When the number of possible faults is established, we can use pattern recognition techniques for classification of the continuously generated parameter vector to the appropriate class of faults. By that manner we can use the real-time modeller for the purpose of fault detection and classification.

References

[1] Hsia, T.C., " System Identification. Least-squares Methods.", Lexington Books, Toronto 1977.

[2] Eykhoff, P., " System Identification. Parameter and State Estimation.", John Wiley & Sons, N.Y. 1974.

[3] Ljung, L., Soderstrom, T., " Theory and Practice of Recursive Identication.", The MIT Press, Cambridge, 1983.

[4] Davies, W.D.T., " System Identyfication for Self-Adaptive Control.", John Wiley & Sons, London 1980.

[5] Siebert, William McC., " Circuit, Signals, and Systems.", The MIT Press, Toronto, 1986.

[6] Seaks, R., Liberty, S.R., " Rational Fault Analysis.", Marcell Dekker Inc., N.Y., 1974.

[7] Goodwin, G.C., Payne, R.L., " Dynamic System Identification. Experiment Design and Data Analysis.", Academic Press, N.Y., 1977.

[8] Kalman, R.E., " Design of a Self-optimizing Control System.", Transactions ASME, Feb. 1958.

[9] Steiglitz, K., McBride, L.E., " A Technique for the Identification of Linear Systems.", IEEE Trans. Autom. Contr. AC-10, 1965.

[10] Levin, M.J., " Optimum Estimation of Impulse Response in the Presence of Noise.", IRE Transactions on Circuit Theory Vol.CT-7, Mar. 1960.

[11] Manitoba HVDC Research Centre, " EMTDC Reference Manual.", 1986.

[12] Kwasnicki, W., " Thevenin Model Generator. Program Description and User Manual.", Project Report for Manitoba HVDC Research Centre, 1986.

[13] Ainsworth, J.D., " Colloquium on HVDC with Weak AC Systems.", International Conference on Large High Voltage Electric Systems.", 1985.