Techniques for data prediction, smoothing, and updating of operator errors in commercial nuclear power plants

by

Mohamad Ali Azarm

A Thesis Submitted to the

Graduate Faculty in Partial Fulfillment of

The Requirements for the Degree of

MASTER OF SCIENCE

Major: Nuclear Engineering

Iowa State University
Ames, Iowa

1979

TABLE OF CONTENTS

ABSTRACT

The human operator models have been reviewed. Those include the continuous and discrete performance and learning models. Appropriate relationships are derived to tailor the models for the use of the Kalman filter in prediction of future data on operator error rate. Detailed presentation is made of the Kalman filter methodology. This is applied to both the performance and learning models. The observability and controllability problems are discussed. A general review of the least square error fitting and the impulse moment updating methods is also given. A comparison is made between the methods discussed here and the Bayes estimation technique. The data of operator errors in BWR's and PWR's are collected. The proper collapsing and smoothing of the data taking into account the plants availability is made. The effect of age, power and the type of the reactor on the operator error rate is studied. The problems with the available data are discussed and certain suggestions are made. A simple statistical code is developed to treat the problem.

It is found that for Pressurized Water Reactors there is a direct correlation between operator error rate and facility size; the larger the PWR, the greater the number of errors committed. While for Boiling Water Reactors, reactor

size does not seem to have any direct affect upon the operator error rate, though the overall error rate for BWR's was larger and considerably more scattered with respect to facility age than similar effects for PWR's. Also, the assumption of constant parameter in the learning function for both BWR's and PWR's can not be satisfied, so a time variant learning parameters are estimated for both BWR's and PWR's.

I. INTRODUCTION

There is always the question, "Why bother to devise a model to fit the data? Why not just use the data?" If we are only interested in frequent historical events and in one set of results, this would be reasonable. If, on the other hand, we wish to forecast future performance from past data, not only is a model (however crude) essential for describing the trend, but also some effective predictive method, such as the Kalman filter, is essential. Furthermore, we must mathematically model the data, especially when the data available is scarce and when dealing with relatively rare events. In other instances, we may wish to compare historically the performance of different operators, different groups of operators, different shifts, even similar functions under the control of different plant managers. Under those circumstances, a model is the most compact way of describing the trend. Vast quantities of data may thus be compressed into lists of parameters which may be easily interpreted. Where the scatter varies statistically from one case to another, statistical properties such as variance may also be used to advantage to condense the data.

In the case of nuclear power plants, the data base under consideration are the operator errors extracted from Licensee Event Reports (LER's). The data are random in nature, and in order to facilitate retrieval of useful information, predictive modeling is necessary.

The nuclear industry and power generation companies in specific, are very concerned about the safe and economical operation of nuclear power plants. The data base has shown that operator error directly effects these goals. Operator errors have caused plant shutdown, delays and reductions in electrical generation, and in a few cases low level radiation release. The object then is to determine first, if an unacceptable level of operator errors now exists, and if so, the determination of methods that can reduce the error rate to an acceptable level. Data modeling techniques are extremely useful for this purpose.

The reduction of operator errors to an acceptable level necessitates the proper design of nuclear power plants from the viewpoint of Human-Factor-Engineering to ease the operation, and a proper choice of training periods. The comparison of estimated learning models for different designs can provide certain guidelines for better design, and the period of training can be chosen from the estimated model for some initial acceptable number of operator errors. In system availability analysis in nuclear power plants, human reliability is one of the most important factors, so a model for human error is necessary for such an analysis.

An attempt has been made in this thesis to construct a model for operator error rate for two different types of LWR's (PWR's and BWR's) with respect to power rate and time. If learning parameters are time variant, a time variant learning model parameter has been estimated.

To construct an operator error rate model the following major steps have been taken:

a. Smoothing the data extracted from LER's;

b. Estimating a static model (time invariant model); and

c. Estimating a dynamic model (time variant model).

To smooth the data two methods have been introduced; window and integral smoothing. Only integral smoothing was used in this study. Least-square-method and Impulse-moment-updating were used for static estimation and Kalman filtering was used for dynamic estimation.

The only reference found similar to this study for dynamic estimation of operator performance using Kalman filtering was written by H. Sriyananda and D. R. Towill in 1974 (7). In this study only forward Kalman filter had been used and the main object was prediction of human operator performance in industry, which can be considered as a primary study of human dynamic modeling.

The operator error rate model and its theoretical justification is discussed in Chapter II. The static and dynamic estimation theory is explained in Chapter III and Chapter IV respectively. Data collection and smoothing techniques are discussed in Chapter V and Chapter VI. The explanation of (OPEXM-K) code and the results can be found in Chapter VII and Chapter VIII.

II. OPERATOR PERFORMANCE MODELS

A. Exponential Model

The human operator performance model may be represented by

$$y(t) = y_c + y_f(1-e^{-t/\tau}) \tag{1}$$

where

y_c = initial performance

$y_c + y_f$ = final performance

τ = learning time constant.

The model is illustrated in Figure 1. The first data point is considered at $t = 0$, and $t = \tau$ implies that $y = y_c + 0.63\ y_f$. Therefore, if $\tau = 6$ weeks the $y = y_c + 0.63\ y_f$ is at the 7th week. Scatter from the original curve can be random (white noise), periodic, or indicate a false ceiling by virtue of a plateau effect (8).

Determination of the model parameters is done through estimation of y_c and y_f by inspection of historical data. By changing the form of Equation (1) into a logarithmic form

$$(t/\tau)\log_{10}e = -\log_{10}(1 - \frac{y-y_c}{y_f}), \tag{2}$$

which is a straight line. τ can be estimated from the

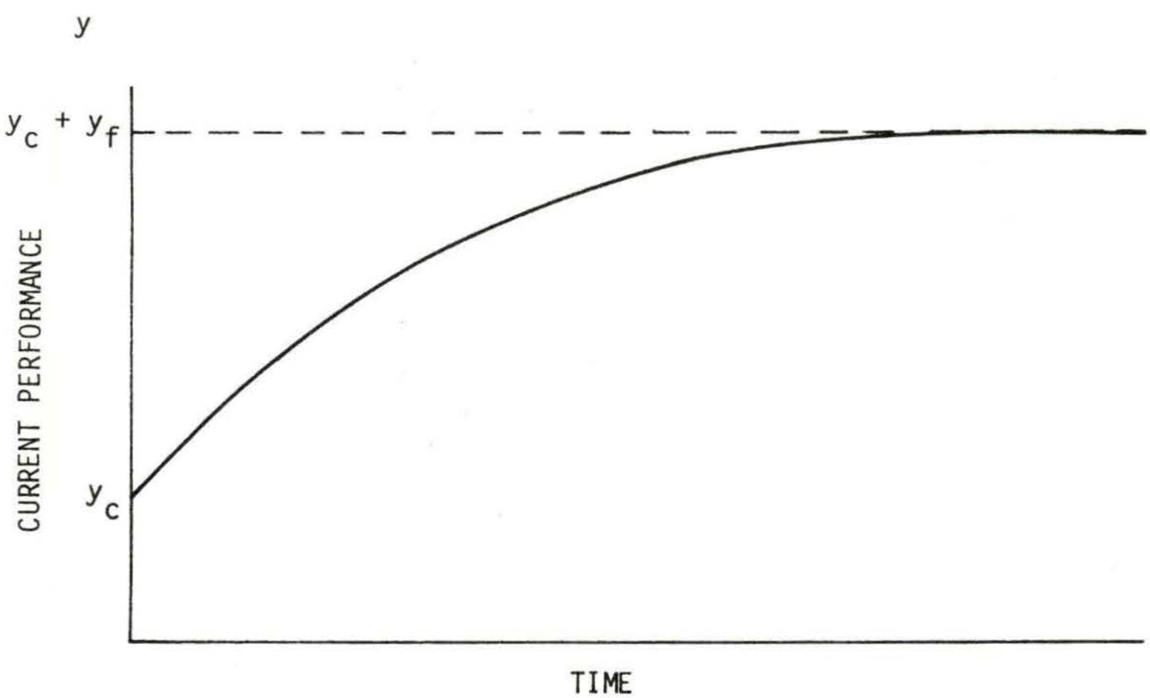

Figure 1. The human performance exponential model

slope of the graph shown in Figure 2.

The impulse response can be derived in the form,

$$h(t) = \frac{dy}{dt} = (y_f/\tau)e^{-t/\tau}. \tag{3}$$

The Laplace transformation of h(t) gives the system transfer function, H(s), as,

$$H(s) = \int_0^\infty h(t)e^{-st}\,dt$$

$$= \frac{y_f}{1 + s\tau} \tag{4}$$

Thus, the block diagram shown in Figure 3 can be used to describe the observed-improvement by a constant parameter exponential model of the type given by Equation (1).

1. Theoretical justification

The validity of the exponential model given in Equation (1) has been verified in practice by successful application to many case studies (7, 3). However, some theoretical justification is obtained by considering the speed-skill acquisition hypothesis of Crossman, in which the improvement curve is explained in terms of the operator's experimenting with alternate methods, rejecting the less successful ones and retaining the better ones (3).

Crossman developed this argument further by considering

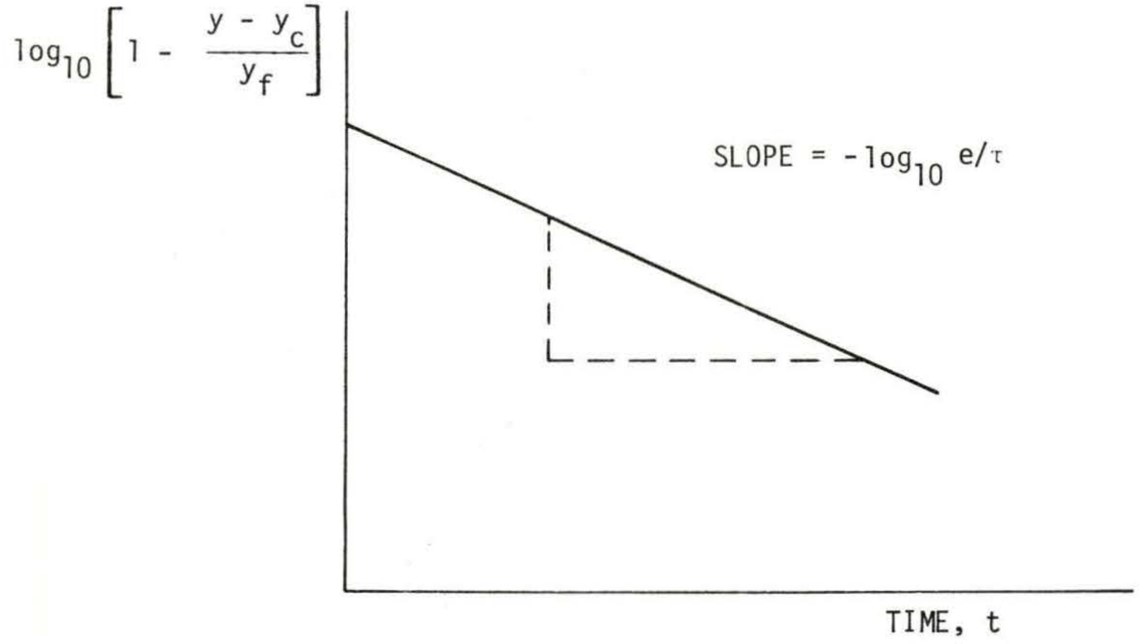

Figure 2. Graphical time constant evaluation

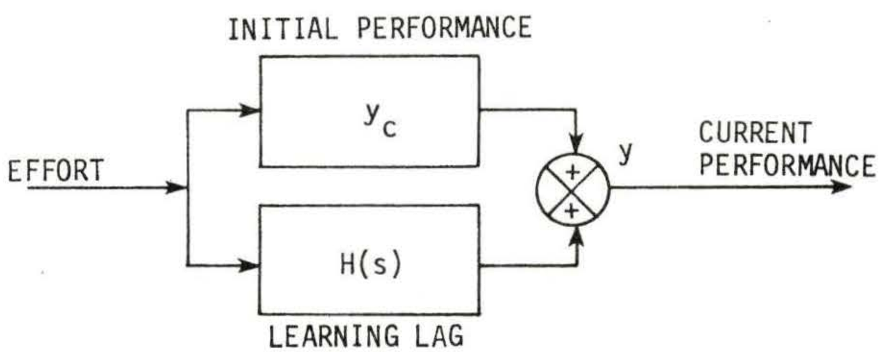

Figure 3. Improvement exponential model

a task with r alternate methods with performance times stepped from 1 to r units. After assuming an initial equal probability for choosing any of the r methods, it is assumed that for task (n+1), the probability of choosing method (i) is reduced by an amount proportional to the difference between the task time using method i and the average task time at task n multiplied by the probability of choosing method i for task n.

2. Time conservative operator

Using the interpretation given by Crossman (3), the exponential model holds everywhere except at the origin. The theory necessitates a second order transfer function for good matching, but may be approximated by an exponential model with time delay as shown in Figure 4. The operator improvement rate in a selective process may be represented analytically by

$$y = \begin{cases} y_c, & t < \theta; \\ y_c + y_f[1-\exp(\frac{t-\theta}{\tau})], & t > \theta; \end{cases} \qquad (5)$$

where θ is the time at which the exponential curve starts to rise. The learning lage transfer function becomes

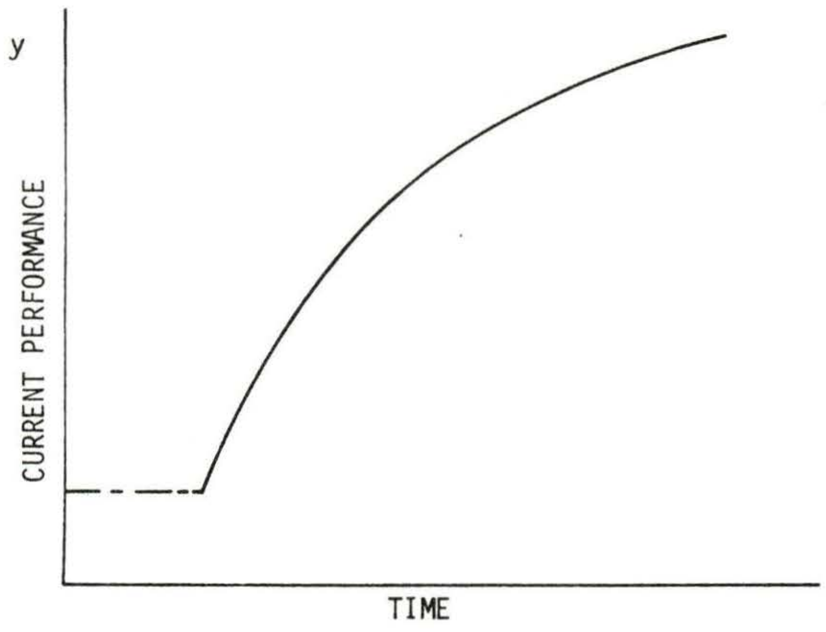

Figure 4. Crossman operator improvement curve

$$H(s) = \frac{y_f e^{-\theta s}}{1+s\tau} \qquad (6)$$

and the block diagram of the learning lag is also as shown in Figure 3. The details of the block diagram is illustrated in Figure 5, where

n = old task number,

$n+1$ = new task number,

i = method index, $i = 1,2,\ldots,r$,

t_i = execution times, $i = 1,2,\ldots,r$,

k = human operator selective constant,

$T(n)$ = old task average time,

$T(n+1)$ = new task average time,

$P_i(n)$ = old probability of using method i,

$P_i(n+1)$ = new probability of using method i,

$\delta P_i(n)$ = change in probability of using method i.

3. <u>Modification to permit time-varying parameters</u>

Assume that

$$\alpha = -1/\tau, \qquad (7)$$

$$c = (y_c + y_f)/\tau. \qquad (8)$$

Thus, Equation (1) can be written in the differential form,

$$\frac{dy}{dt} = \alpha y + c, \qquad (9)$$

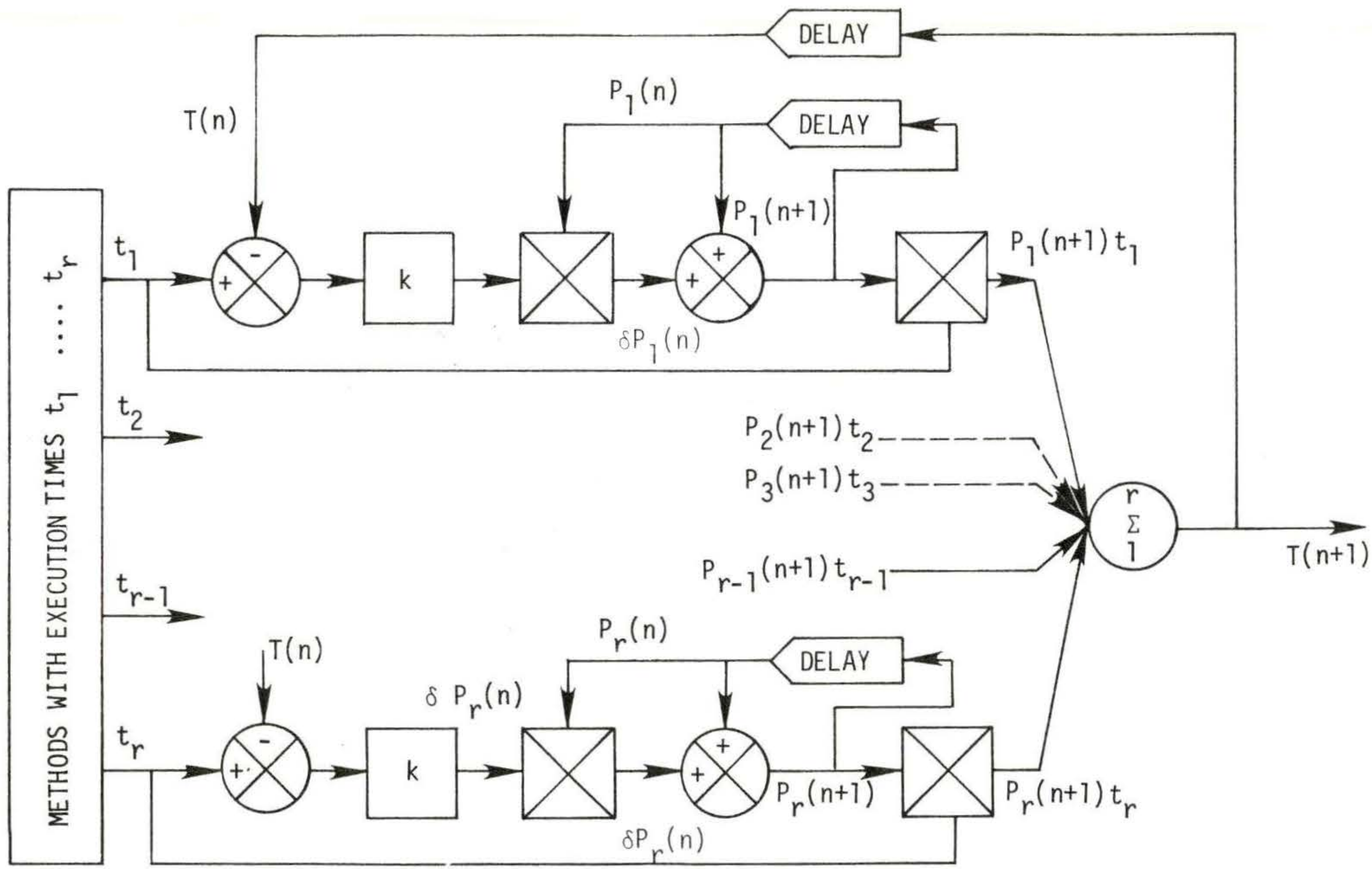

Figure 5. Block diagram of Crossman human operator selective process

$$\frac{d\alpha}{dt} = \frac{dc}{dt} = 0. \qquad (10)$$

The changes in α and c during the estimation procedure can be covered by noise parameters which can be included in the formulation of the model.

B. Linear Discrete Model

The advantages of discretizing the exponential model are:

1. in practice, data are available only at discrete intervals, and
2. the computation for a discrete model is very much simpler than for a continuous model.

The model is approximated by the discrete form

$$y_{t+\Delta t} = y_t + \alpha_t y_t \Delta t + c_t \Delta t \qquad (11)$$

$$\alpha_{t+\Delta t} = \alpha_t \qquad (12)$$

$$c_{t+\Delta t} = c_t \qquad (13)$$

These equations are nonlinear. To linearize Equation (11) we may expand the parameters about the estimated values of y_t, α_t and c_t in a Taylor's series, that is

$$y_{t+\Delta t} = y_t(1+\hat{\alpha}_y \Delta t) + \alpha_t(\hat{y}_t \Delta t) + c_t \Delta t - \hat{\alpha}_t \hat{y}_t \Delta t \qquad (14)$$

where (^) is used to designate estimated values.

Equations (12), (13) and (14) can then be represented in the matrix form

$$\begin{bmatrix} y \\ \alpha \\ c \end{bmatrix}_{t+\Delta t} = \begin{bmatrix} 1+\hat{\alpha}\Delta t & \hat{y}\Delta t & \Delta t \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y \\ \alpha \\ c \end{bmatrix}_t + \begin{bmatrix} -\hat{\alpha}\hat{y}\Delta t \\ 0 \\ 0 \end{bmatrix}_t . \tag{15}$$

C. Instantaneous Error Rate Model

1. Continuous learning model

The instantaneous operator error rate may be represented by a model similar to the performance model which represents an increase in learning or a decrease in error rate. The dynamic learning model can take either of the two forms

$$\lambda = a(1+be^{-t/\tau}) \tag{16}$$

$$\frac{d\lambda}{dt} = -(ab/\tau)e^{-t/\tau} \tag{17}$$

The differential form given in Equation (17) may be written in the form of Equation (9) by defining

$$\alpha = -1/\tau \tag{18}$$

$$c = a/\tau \tag{19}$$

Thus,

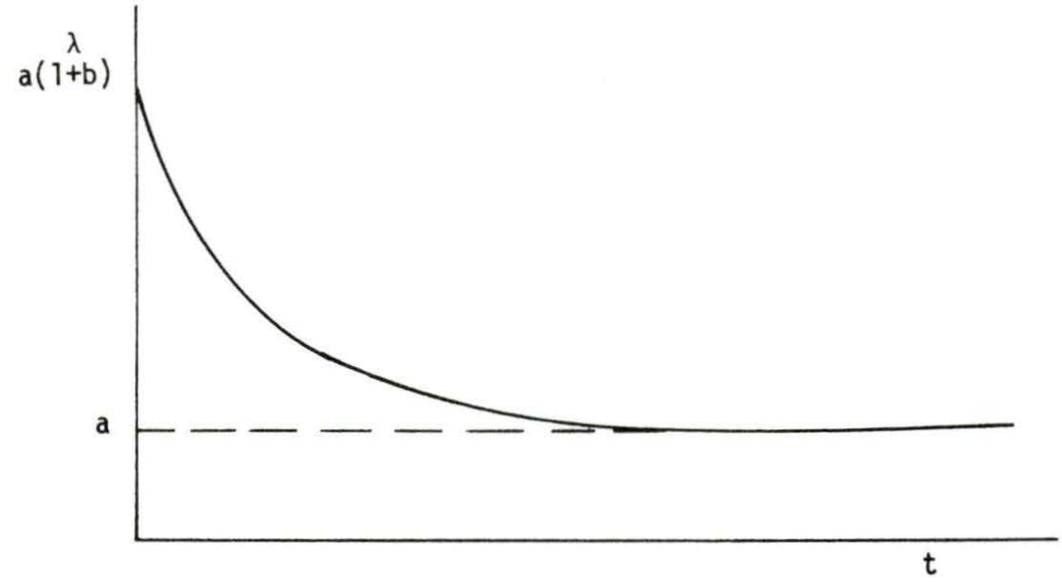

Figure 6. Learning model

$$\frac{d\lambda}{dt} = \alpha\lambda + c \tag{20}$$

and

$$\frac{d\alpha}{dt} = \frac{dc}{dt} = 0 \tag{21}$$

These equations can be represented by the matrix form

$$\begin{bmatrix} \dot{\lambda} \\ \dot{\alpha} \\ \dot{c} \end{bmatrix} = \begin{bmatrix} \alpha & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \alpha \\ c \end{bmatrix} \tag{22}$$

where ($\dot{}$) is used to indicate the time differential.

2. <u>Linear discrete learning model</u>

Equation (20) may be rewritten in the discretized form

$$\dot{\lambda}\Delta t = \lambda_{t+\Delta t} - \lambda_t = (\alpha_t\lambda_t + c_t)\Delta t \tag{23}$$

then

$$\lambda_{t+\Delta t} = \lambda_t + \alpha_t\lambda_t\Delta t + c_t\Delta t \tag{24}$$

and similarly

$$\alpha_{t+\Delta t} = \alpha_t \tag{25}$$

and

$$c_{t+\Delta t} = c_t \ . \tag{26}$$

The nonlinear relationship of Equation (24) can be linearized by the same method as that used in Equations (11) and (14), thus, Equation (22) becomes

$$\begin{bmatrix} \lambda \\ \alpha \\ c \end{bmatrix}_{t+\Delta t} = \begin{bmatrix} 1+\hat{\alpha}\Delta t & \hat{\lambda}\Delta t & \Delta t \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda \\ \alpha \\ c \end{bmatrix}_t + \begin{bmatrix} \hat{\alpha}\hat{\lambda}\Delta t \\ 0 \\ 0 \end{bmatrix}_t \qquad (27)$$

The same procedure can be used to discretize and linearize the form given by Equation (17).

III. ESTIMATION TECHNIQUES

Once a model is developed to describe a given phenomenon or behavior and is justified by using field data, then by means of different estimation techniques the parameters of the model can be determined. Modern estimation techniques can be divided into two categories:

1. Static estimation
2. Dynamic estimation

The "least square error" fitting and "impulse moment" updating are considered static estimators and they will be discussed in this chapter. The Kalman filter which is considered as dynamic estimator, will be discussed in the next chapter. To use the model parameters in Kalman filter process as initial conditions, one can filter the data and estimates the time dependence of model parameters. By means of Kalman filter, the false ceiling or plateau can be detected. Kalman filter can be used as a predictor as well.

A. Static Estimation Techniques

1. Least square estimation

The static least square method which is used as one of the a priori estimation, is based upon Taylor series expansion of learning or performance function at points of estimation. Assuming $Z = \frac{1}{\tau}$ and the estimated value of (y) at time (T_i) as $(\bar{y}_i)$, one can write

$$\bar{y}_i = f(y_c, y_f, Z, T_i) \tag{28}$$

with the first order expansion as

$$\bar{y}_i \simeq f(\bar{y}_c, \bar{y}_f, \bar{Z}, T_i) + \frac{\partial f}{\partial y_f} \Delta y_f + \frac{\partial f}{\partial z} \Delta z + \frac{\partial f}{\partial y_c} \Delta y_c \tag{29}$$

in which

$$\frac{\partial f}{\partial y_c} = 1 \tag{30}$$

$$\frac{\partial f}{\partial y_f} = [1-\exp(-T_i\bar{Z})] \tag{31}$$

$$\frac{\partial f}{\partial z} = \bar{y}_f \; T_i \; \exp(-T_i\bar{Z}). \tag{32}$$

Substituting Equations (30, 31, and 32) in Equation (29) results in Equation (33)

$$y_{ir} \simeq \bar{y}_{cr} + \bar{y}_{fr}(1-\exp(-T_i\bar{Z})) + \Delta y_c + [1-\exp(-T_i\bar{Z})]\Delta y_f$$

$$+ [T_i \cdot y_{fr} \cdot \exp(-T_i\bar{Z})\Delta z] \tag{33}$$

where subscript "r" is the number of iterations.

The estimation can be done in the sense of the least square method by minimizing the sum of error squared values as follows

$$E^2 = \sum_{i=1}^{N} \{y_i - \bar{y}_{cr} - \bar{y}_{fr}[1-\exp(-T_i Z_r)] - \Delta y_c$$

$$-[1-\exp(-T_i \bar{Z}_r)]\Delta y_f - \bar{y}_{fr} \times T_i \times \exp(-T_i \bar{Z}_r)\Delta Z\}^2.$$

Assuming (34)

$$\Delta y_i = y_i - \bar{y}_{cr} - \bar{y}_{fr}[1-\exp(-T_i \bar{Z}_r)] \quad (35)$$

and substituting Equation (35) into Equation (34) the simplified error squared value can be obtained as;

$$E^2 = \sum_{i=1}^{N} \{\Delta y_i - \Delta y_c - [1-\exp(-t_i \bar{Z}_r)]\Delta y_f$$

$$- \bar{y}_{fr} t_i \exp(-t_i \bar{Z}_r)\Delta Z\}^2. \quad (36)$$

To get the local minimum of the sum of errors squared, the following constraints have to be met;

$$\frac{\partial E^2}{\partial \Delta y_c} = 0 \quad (37)$$

$$\frac{\partial E^2}{\partial \Delta y_f} = 0 \quad (38)$$

$$\frac{\partial E^2}{\partial \Delta Z} = 0. \quad (39)$$

The above constraints will result into the following set of simultaneous equations;

$$S_1 = R_1 \Delta y_c + Q_1 \Delta y_f + P_1 \Delta Z \quad (40)$$

$$S_2 = R_2 \Delta y_c + Q_2 \Delta y_f + P_2 \Delta Z \quad (41)$$

$$S_3 = R_3 \Delta y_c + Q_3 \Delta y_f + P_3 \Delta Z \quad (42)$$

where the parameters are defined as;

$$S_1 = \sum_{i=1}^{N} \Delta y_i \tag{43}$$

$$S_2 = \sum_{i=1}^{N} \Delta y_i [1-\exp(-T_i \overline{Z}_r)] \tag{44}$$

$$S_3 = \overline{y}_{fr} \sum_{i=1}^{N} \Delta y_i \times T_i \times \exp(-T_k \overline{Z}_r) \tag{45}$$

$$R_1 = N \tag{46}$$

$$R_2 = \sum_{i=1}^{N} [1-\exp(-T_i \overline{Z}_r)] \tag{47}$$

$$R_3 = \overline{y}_{fr} \sum_{i=1}^{N} T_i \exp(-T_i \overline{Z}_r) \tag{48}$$

$$Q_1 = \sum_{i=1}^{N} [1-\exp(-T_i \overline{Z}_r)] \tag{49}$$

$$Q_2 = \sum_{i=1}^{N} [1-\exp(-T_i \overline{Z}_r)]^2 \tag{50}$$

$$Q_3 = y_{fr} \sum_{i=1}^{N} [1-\exp(-T_i \overline{Z}_r)] T_i \cdot \exp(-T_i \overline{Z}_r) \tag{51}$$

$$P_1 = \overline{y}_{fr} \sum_{i=1}^{N} [T_i \cdot \exp(-T_i \overline{Z}_r)] \tag{52}$$

$$P_2 = \overline{y}_{fr} \sum_{i=1}^{N} [1-\exp(-T_i \overline{Z}_r)] T_i \cdot \exp(-T_i \overline{Z}_r) \tag{53}$$

$$P_3 = \overline{y}_{fr}^{\,2} \sum_{i=1}^{N} [T_i \exp(-T_i \overline{Z}_r)]^2 . \tag{54}$$

The new iterated parameters can be obtained by

$$\overline{y}_{c(r+1)} = \overline{y}_{cr} + \Delta y_c \tag{55}$$

$$\overline{y}_{f(r+1)} = \overline{y}_{fr} + \Delta y_f \tag{56}$$

$$\overline{Z}_{r+1} = \overline{Z}_r + \Delta Z, \tag{57}$$

The number of necessary iteration can be determined by putting an accuracy condition on the estimated parameters.

2. Impulse moment updating

Impulse moment updating was first introduced by Ba Hli (5) to estimate the coefficients of a transfer function. This method is modified to estimate the parameters of a learning curve model as follows;

Consider the exponential model for performance as given in Equation (1), if we define $y-y_c$ as inspection task, that is

$$y-y_c = y_f[1-\exp(-t/\tau)] \tag{1}$$

then in the system shown in Figure 7 the input can be defined as effort. Unit step may be assumed as a normalized effort. The transfer function of the system or the Laplace transform of the impulse response is

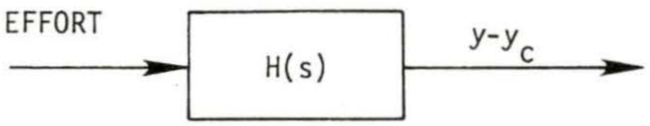

Figure 7. Inspection task performance

$$H(s) = \frac{y_f}{1+\tau s} \tag{58}$$

or

$$H(s) = \int_0^{\infty} h(t)\ \exp(-st)dt \tag{59}$$

where

$$h(t) = \frac{y_f}{\tau}\ e^{-t/\tau}. \tag{60}$$

Expanding Equation (58, 59) is the "s" domain, we get

$$H(s) = \int_0^{\infty} h(t)dt - s\int_0^{\infty} th(t)dt + \frac{s^2}{2!}\int_o^{\infty} t^2h(t)dt + \ldots \tag{61}$$

$$H(s) = y_f[1-\tau s + (\tau s)^2 - (\tau s)^3 + \ldots]$$

$$= \sum_{n=0}^{\infty} y_f(-1)^n(\tau s)^n\ . \tag{62}$$

Equating Equation (61) with Equation (62) term by term results in

$$y_f = \int_0^{\infty} h(t)dt \tag{63}$$

$$y_f\tau = \int_0^{\infty} th(t)dt\ . \tag{64}$$

By using the "Ba Hli" approach (5), the left hand side of Equation (63) and Equation (64) can be evaluated as follows

$$\{h\}_A = \frac{\{f_0\}}{\{f_i\}} \tag{65}$$

where $\{f_0\}$ is the output sequence set of the system, that is

$$\{f_0\} = \{f_{01}, f_{02}, f_{03}, \ldots\}, \tag{66}$$

$\{f_i\}$ is the input sequence set to the system

$$\{f_i\} = \{f_{i1}, f_{i2}, f_{i3}, \ldots\} \tag{67}$$

and $\{h\}_A$ denotes the area under the impulse response curve corresponding to the time sequence which is the unit response. Using Equation (65, 64, 63) the model parameters can be estimated, that is

$$y_f = \sum_{i=1}^{N} \{h\}_{Ai} \tag{68}$$

and

$$\tau = \frac{\sum_{i=1}^{N} \bar{t}_i \{h\}_{A_i}}{\sum_{i=1}^{N} \{h\}_{Ai}} \tag{69}$$

where $\bar{t}_i$ is a weighted time between two adjacent points when data are sampled in different time intervals, which may be assumed as

$$\bar{t}_i = \frac{t_i + t_{i+1}}{2}\ . \tag{70}$$

Another formula which is usually used instead of

Equation (69) in the case of exponential models is

$$\tau = \frac{1}{y_f} \sum_{i=0}^{N-1} (y_{i+1}-y_i) \frac{t_i+t_{i+1}}{2} . \qquad (71)$$

This formula resulted from impulse moment updating for special case of the learning performance of operator.

A simpler approach than impulse moment updating can be used to derive Equation (71). The shaded area in Figure 8, that is,

$$\text{shaded area} = \int_0^\infty y_f - y_f(1-e^{-t/\tau})dt = y_f\tau \qquad (72)$$

$$\text{shaded area} = \sum_{i=0}^{N-1} (y_{i+1}-y_i)t_i + \frac{\Delta t_i}{2}$$

$$= \sum_{i=0}^{N-1} (y_{i+1}-y_i) \frac{t_i+t_{i+1}}{2} . \qquad (73)$$

Equating Equation (72) with Equation (73) will be resulted in Equation (71).

Impulse moment updating may result in wrong estimation especially if there is too much scattering at large intervals of time.

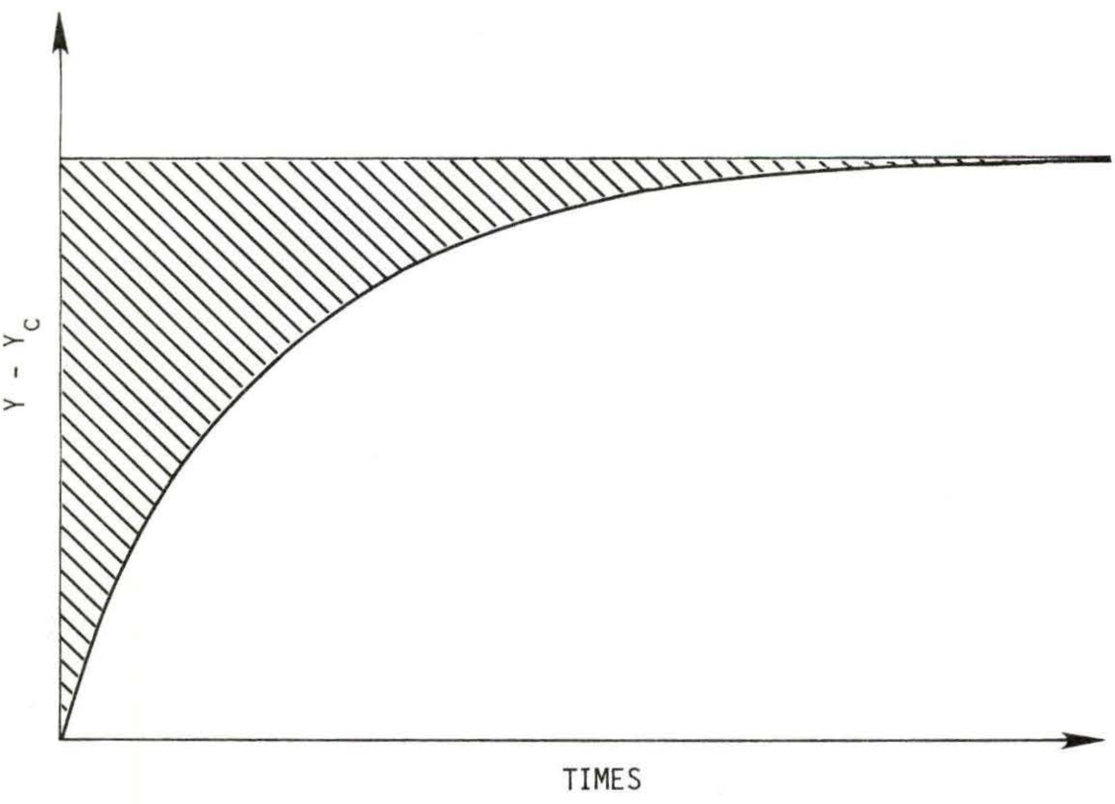

Figure 8. Illustration of impulse moment updating

IV. KALMAN FILTER

A. Definitions and Notations

1. A "consistent" estimate of a system vector $\underline{x}$; that is $\hat{\underline{x}}$, is one which converges to the true values of $\underline{x}$, as the number of measurements increases.

2. A minimum variance (unbiased) estimate has the property that its error variance is less than or equal to that of any other unbiased estimate.

3. A linear continuous dynamic system may be represented by the state vector and the observation vector

$$\dot{\underline{x}} = F(t)\ \underline{x}(t) + L(t)\ \underline{u}(t) + G(t)\ \underline{w}(t) \tag{74}$$

$$\underline{z} = H(t)\ \underline{x}(t) + \underline{v}(t) \tag{75}$$

where $\underline{w}(t)$ is the system noise and $\underline{v}(t)$ is the observation noise.

4. A linear discrete dynamic system is represented by the state vector and the observation vector

$$\underline{x}_{k+1} = \phi_k \underline{x}_k + \Gamma_k \underline{w}_k + \Lambda_k \underline{u}_k \tag{76}$$

$$\underline{z}_k = H_k \underline{x}_k + \underline{\nu}_k \tag{77}$$

where $\underline{w}_k$ and $\underline{\nu}_k$ are the system and the observation noise vector respectively.

5. The continuous controllability matrix is

$$\theta_1 = [L \vdots FL \vdots \ \dots \ \vdots F^{n-1}L] \tag{78}$$

6. The discrete controllability matrix is

$$\theta_2 = [\Lambda \vdots \Lambda\phi \vdots \ \dots \ \vdots \phi^{n-1}\Lambda] \tag{79}$$

7. The continuous observability matrix is

$$\theta_3 = [H^T \vdots F^T H^T \vdots (F^T)^2 H^T \vdots \ \dots \ \vdots (F^T)^{n-1} H^T] \tag{80}$$

8. The discrete observability matrix is

$$\theta_4 = [H^T \vdots \phi^T H^T \vdots (\phi^T)^2 H^T \vdots \ \dots \ \vdots (\phi^T)^{n-1} H^T] \tag{81}$$

9. The system error may be represented by

$$\tilde{x} = \underline{\hat{x}} - \underline{x} \tag{82}$$

where the tilde is used to designate estimation errors.

10. The continuous system error covariance is

$$P = E[(x-E(x))(x-E(x))^T] = E[\tilde{x}\ \tilde{x}^T] \tag{83}$$

11. The discrete system error covariance is

$$P_k = E[(\underline{x}_k - E(\underline{x}_k))(\underline{x}_k - E(\underline{x}_k))^T] = E[\underline{\tilde{x}}_k \underline{\tilde{x}}_k^T] \tag{84}$$

12. The white noise error covariance is

$$E[(G(t)\underline{w}(t))(G(\tau)\underline{w}(\tau))^T] = G(t)Q(t)G(t)^T\delta(t-\tau) \qquad (85)$$

where Q(t) is spectral density matrix.

13. The white sequence noise error covariance is

$$E[(\Gamma_k\underline{w}_k)(\Gamma_\rho\underline{w}_\rho)^T] = \begin{cases} \Gamma_k Q_k \Gamma_k^T & k = \rho \\ 0 & k \neq \rho \end{cases} \qquad (86)$$

for equivalent continuous system of the discrete system of vice versa, it can be shown

$$\Gamma_k Q_k \Gamma_k^T = \int_{t_k}^{t_{k+1}} \phi(t_{k+1},\tau)G(\tau)Q(\tau)G(\tau)^T\phi(t_{k+1},\tau)^T d\tau \qquad (87)$$

14. The continuous observation noise covariance is

$$R = E[(\underline{z}-H\underline{x})(\underline{Z}-H\underline{x})^T] \qquad (88)$$

15. The discrete observation noise covariance is

$$R_k = E[(\underline{z}_k-H\underline{x}_k)(\underline{z}_k-H\underline{x}_k)^T] \qquad (89)$$

16. "A" is positive definite if

$$\underline{z}^T A\underline{z} > 0 \quad \text{for all} \quad \underline{z} \neq 0 \qquad (90)$$

and "A" is positive-semi-definite if

$$\underline{z}^T A\underline{z} \geq 0 \quad \text{for all} \quad \underline{z} \neq 0 \qquad (91)$$

For example, Q as below is positive definite

$$Q = \begin{matrix} 1 & 0 \\ 0 & 0 \end{matrix} \tag{92}$$

and R is positive semi-definite

$$R = \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \tag{93}$$

17. The notations (-) and (+) are used to denote the time immediately before and immediately after a discrete measurement, respectively.

B. Discrete Kalman Filtering

To use Kalman filter to study operator learning process, we are more interested in discrete form, because of inherent recursive characteristics of discrete. The Kalman filter would be easier to use computer programming considering a discrete system represented by

$$\underline{x}_{k+1} = \phi_k \underline{x}_k + \Gamma_k \underline{w}_k + \Lambda_k \underline{u}_k \tag{94}$$

$$z_k = H_k \underline{x}_k + \underline{v}_k \tag{95}$$

The priori estimate of the system at time t_k is denoted $\hat{x}_k(-)$. We seek an update estimate $\underline{x}_k(+)$ based on the use of the measurement $\underline{z}_k$; that is

$$\underline{\hat{x}}_k(+) = K_k' \underline{\hat{x}}_k(-) + K_k \underline{z}_k \tag{96}$$

where K_k' and K_k are time variant parameters, which can be found later on. By definition,

$$\underline{\hat{x}}_k(+) = \underline{x}_k + \underline{\tilde{x}}_k(+) \tag{97}$$

$$\underline{\hat{x}}_k(-) = \underline{x}_k + \underline{\tilde{x}}_k(-) \tag{98}$$

substituting Equation (95), Equation (97) and Equation (98) into Equation (96), we get

$$\underline{\tilde{x}}_k(+) = [K_k'+K_kH_k-I]\underline{x}_k + K_k'\tilde{x}_k(-) + K_k\underline{v}_k. \tag{99}$$

To have unbiased estimates, the following conditions are required for white noise sequences

$$E[\underline{\ }_k] = 0 \tag{100}$$

for unbiased estimate,

$$E[\underline{\tilde{x}}_k(-)] = 0 \tag{101}$$

$$E[\underline{\tilde{x}}_k(+)] = 0 \tag{102}$$

By using the above conditions the expected value of K_k' is

$$K_k' = I-K_kH_k \tag{103}$$

Thus, the estimator take the form

$$\underline{\hat{x}}_k(+) = (I-K_kH_k)\underline{\hat{x}}_k(-) + K_k\underline{z}_k \tag{104}$$

or

$$\underline{\hat{x}}_k(+) = \underline{\hat{x}}_k(-) + K_k[z_k - H_k\underline{\hat{x}}_k(-)] \tag{105}$$

and

$$\underline{\tilde{x}}_k(+) = (I-K_kH_k)\underline{\tilde{x}}_k(-) + K_k\underline{\nu}_k \tag{106}$$

1. Error covariance update

The expression for the change in the error covariance matrix, P_k, when a measurement is employed can be derived as follows, from Equation (107)

$$P_k(+) = E[\underline{\tilde{x}}_k(+)\underline{\tilde{x}}_k(+)^T] \tag{107}$$

Substituting Equation (106) into Equation (107)

$$P_k(+) = E\{(I-K_kH_k)\underline{\tilde{x}}_k(-)[\underline{x}_k(-)^T(I-K_kH_k)^T + \underline{v}_k^TK_k^T] + K_k\underline{v}_k[\underline{\tilde{x}}_k(-)^T(I-K_kH_k)^T + \underline{v}_k^TK_k^T]\} \tag{108}$$

By definition

$$E[\underline{\tilde{x}}_k(-)\underline{\tilde{x}}_k(-)^T] = P_k(-). \tag{109}$$

$$E[\underline{\nu}_k\underline{\nu}_k^T] = R_k \tag{110}$$

and, as a result of measurement errors being uncorrelated

$$E[\underline{\tilde{x}}_k(-)\underline{v}_k^T] = E[\underline{v}_k\underline{\tilde{x}}_k(-)^T] = 0 \tag{111}$$

Thus,

$$P_k(+) = (I-K_kH_k)P_k(-)(I-K_kH_k)^T + K_kR_kK_k^T \tag{112}$$

2. Optimum choice of the Kalman gain

The parameter K_k shall be chosen such that a weighted scalar sum of the diagonal elements of the error covariance matrix $P_k(+)$ is minimized thus, for the cost function we choose

$$J_k = E[\underline{\tilde{x}}_k(+)^T S \underline{\tilde{x}}_k(+)] \tag{113}$$

where "S" is positive-semi-definite matrix. The optimal estimate is independent of "S", hence, we may as well choose S = I, yielding

$$J_k = \text{trace}[P_k(+)] \tag{114}$$

This is equivalent to minimize the length of the estimation error vector. To find the value of K_k which provides a minimum, it is necessary to take the partial derivative of J_k with respect to K_k and equate it to zero. Use is made of the relation for the partial derivative of the trace of the product of two matrices A and B (with B symmetric). Thus,

$$\frac{\partial}{\partial A}(\text{trace}(A\ B\ A^T))] = 2\ A\ B \tag{115}$$

and if S = I, then

$$\frac{\partial J_k}{\partial K_k} = \frac{\partial E[\underline{\tilde{x}}_k(+)^T S \underline{\tilde{x}}_k(+)]}{\partial K_k} = \frac{\partial P_k(+)}{\partial K_k} \tag{116}$$

$$\frac{\partial J_k}{\partial K_k} = -2(I-K_k H_k)P_k(-)H_k^T + 2K_k R_k \tag{117}$$

When

$$\frac{\partial J_k}{\partial K_k} = 0,$$

then

$$K_k = P_k(-)H_k^T[H_k P_k(-)H_k^T + R_k]^{-1} \tag{118}$$

which is the Kalman gain. Thus,

$$P_k(+) = P_k(-) - P_k(-)H_k^T[H_k P_k(-)H_k^T + R_k]^{-1}H_k P_k(-)$$

$$= [I-K_k H_k]P_k(-) \tag{119}$$

The extrapolation of the above quantities between measurements is,

$$\underline{\hat{x}}_k(-) = \phi_{k-1}\hat{x}_{k-1}(+) + \Lambda_{k-1}\cdot U_{k-1} \tag{120}$$

$$P_k(-) = \phi_{k-1}P_{k-1}(+)\phi_{k-1}^T + Q_{k-1} \tag{121}$$

A timing diagram for linear discrete Kalman filter is shown in Figure 9 and a summary of the parameters involved is given in Table 1.

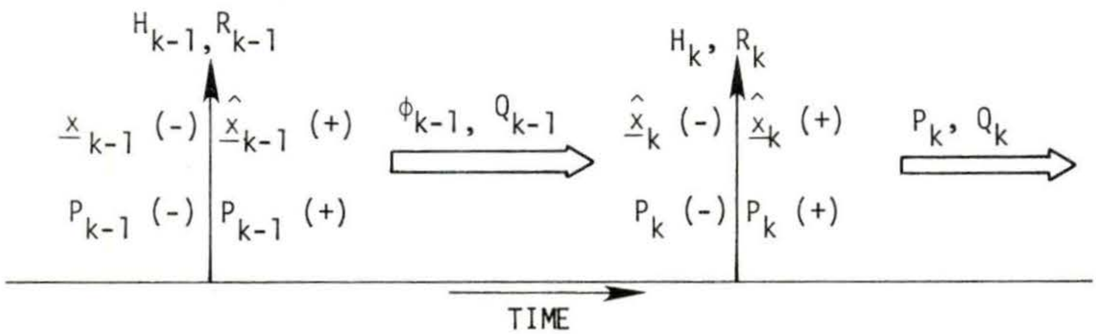

Figure 9. Timing diagram for linear discrete Kalman filter (7)

Table 1. Summary of the Kalman filter equations

System model	$\underline{x}_k = \phi_{k-1}\underline{x}_{k-1} + \underline{w}_{k-1}$, $\underline{w}_k \sim N(\underline{0}, Q_k)$	
Measurement model	$\underline{z}_k = H_k\underline{x}_k + \underline{v}_k$, $\underline{v}_k \sim N(0, R_k)$	
Initial conditions	$E[\underline{x}(0)] = \hat{\underline{x}}_0$	
	$E[(\underline{x}(0)-\hat{\underline{x}}_0)(\underline{x}(0)-\hat{\underline{x}}(0))^T] = P_0$	
Other assumptions	$E[\underline{w}_k v_j^T] = 0$ for all j,k	
State estimate extrapolation	$\hat{\underline{x}}_k(-) = \phi_{k-1}\hat{\underline{x}}_{k-1}(+)$	(120)
Error covariance extrapolation	$P_k(-) = \phi_{k-1}P_{k-1}(+)\phi_{k-1}^T + Q_{k-1}$	(121)
State estimate update	$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k[\underline{z}_k - H_k\hat{\underline{x}}_k(-)]$	(104)
Error covariance update	$P_k(+) = [I-K_kH_k]P_k(-)$	(117)
Kalman gain matrix	$K_k = P_k(-)H_k^T[H_kP_k(-)H_k^T + R_k]^{-1}$	
	$= P_k(+)H_k^TR_k^{-1}$	(125)

3. <u>A simple form of the Kalman gain</u>

If we get the inverse of $P_k(+)$ we will have

$$P_k(+)^{-1} = P_k(-)^{-1} + H_k^TR_k^{-1}H_k \tag{122}$$

thus,

$$K_k = [P_k(+)P_k(+)^{-1}]P_k(-)H_k^T[H_kP_k(-)H_k^T + R_k]^{-1} \qquad (123)$$

$$P_k(+)[P_k(-)^{-1} + H_k^TR_k^{-1}H_k]P_k(-)H_k^T[H_kP_k(-)H_kP_k(-)H_k^T + R_k]^{-1} \qquad (124a)$$

$$K_k = P_k(+)H_k^T[I+R_k^{-1}H_kP_k(-)H_k^T][H_kP_k(-)H_k^T + R_k]^{-1} \qquad (124b)$$

or

$$K_k = P_k(+)H_k^TR_k^{-1} \qquad (125)$$

C. Propagation of Errors and Optimal Propagation

Consider the problem of estimating the state of a dynamic system in which the state vector $\underline{x}$ is shown at some time t_k with an uncertainty expressed by the error covariance matrix

$$P_k(+) = E[\underline{\tilde{x}}_k(+)\underline{\tilde{x}}_k(+)^T]. \qquad (104)$$

The error in the estimate at t_{k+1} is unbiased if

$$\underline{\tilde{x}}_{k+1}(-) = \phi_k\underline{\tilde{x}}_k(+) - \Gamma_kw_k \qquad (126)$$

and the expected value of the error is

$$E[\tilde{x}_{k+1}(-)] = \phi_kE[\underline{\tilde{x}}_k(+)] - \Gamma_kE[\underline{w}_k] = 0 \qquad (127)$$

By definition,

$$P_{k+1}(-) = E[\tilde{\underline{x}}_{k+1}(-)\underline{x}_{k+1}(-)^T] \qquad (128)$$

Thus,

$$\tilde{\underline{x}}_{k+1}(-)\tilde{\underline{x}}_{k+1}(-)^T = (\phi_k\tilde{\underline{x}}_k(+)-\Gamma_k\underline{w}_k)(\phi_k\tilde{\underline{x}}_k(+)-\Gamma_k\tilde{\underline{w}}_k)^T$$

$$= \phi_k\tilde{\underline{x}}_k(+)\tilde{\underline{x}}_k(+)^T\phi_k^T - \phi_k\tilde{\underline{x}}_k(+)\underline{w}_k^T\Gamma_k^T$$

$$- \gamma_k\underline{w}_k\tilde{\underline{x}}_k^T\phi_k^T + \Gamma_k\underline{w}_k\underline{w}_k^T\Gamma_k^T. \qquad (129)$$

Assuming a consequence of the fact that $\Gamma_k\underline{w}_k$ is white sequence

$$E[\tilde{\underline{x}}_k(+)(\Gamma_k\underline{w}_k)^T] = 0 \qquad (130)$$

Thus

$$P_{k+1}(-) = \phi_k P_k(+)\phi_k^T + \Gamma_k Q_k \Gamma_k^T \qquad (131)$$

Optimal prediction can be thought of, quite simply, in terms of optimal filtering in the absence of measurement errors (thus, $R^{-1} \to 0$ and hence $K \to 0$). Therefore, if measurements are unavailable beyond some time, t_0, the optimal of $\underline{x}(t)$ for $t \geq t_0$ given $x(t_0)$ must be obtained from

$$\hat{x}(t) = \phi_k(t,t_0)\hat{\underline{x}}(t_0) + \Lambda_k u_k \quad \text{(discrete system)} \qquad (132)$$

and

$$\hat{x}(t) = F(t)\hat{x}(t) + L(t)u(t) \text{ (continuous system).} \qquad (133)$$

The corresponding equation for uncertainty in the optimal predictions, given $P(t_0)$, are

$$P_{k+1}(-) = \phi_k P_k(+)\phi_k^T + Q_k \tag{134}$$

for single time stage in discrete system or

$$\frac{dP}{dt} = FP + PF^T + GQG^T \tag{135}$$

for continuous system.

D. Optimal Filter Smoothing

An optimal smoother can be thought of as a suitable combination of two optimal filters, Figure 10. One of the filters, called a "forward filter", operates on all the data before time t and produces the estimate $\hat{\underline{x}}_f(t)$; the other filter, called a "backward filter", operates on all the data after time t and produces the estimate $\hat{\underline{x}}_b(t)$. Together these two filters utilize all the available information.

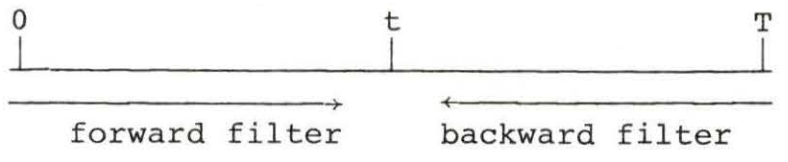

Figure 10. Forward and backward filters

Three types of optimal smoothing are possible

1. Fixed internal smoothing: the initial and final times (0 to T) are fixed and the estimate $\underline{\hat{x}}(t/T)$ is sought as T increases.

2. Fixed point smoothing: $\underline{\hat{x}}(t/T)$ is sought as T increases, with t considered constant.

3. Fixed-lag smoothing: $\underline{\hat{x}}(T-\Delta/T)$ is sought as T increases, with Δ held fixed,

Assuming the estimate of $\underline{\hat{x}}$ is a linear combination of backward and forward estimates, thus

$$\underline{\hat{s}}(t/T) = A\underline{\hat{x}}_f(t) + A'\underline{\hat{x}}_b(t) \qquad (136)$$

and the estimation error is

$$\underline{\tilde{x}}(t/T) = (A+A'-I)\underline{x}(t) + A\underline{\tilde{x}}_f(t) + A'\underline{\tilde{x}}_b(t) \ . \qquad (137)$$

For unbiased filtering errors, $\underline{\tilde{x}}_f(t)$ and $\underline{\tilde{x}}_b(t)$, we wish to obtain as unbiased smoothing error, $\underline{\tilde{x}}(t/T)$, that is

$$E[\tilde{x}(t/T)] = 0. \qquad (138)$$

Thus,

$$A' = I-A \qquad (139)$$

Therefore,

$$\tilde{x}(t/T) = A\underline{\tilde{x}}_f(t) + (I-A)\tilde{x}_b(t). \qquad (140)$$

Computing the smoother error covariance, we find

$$P(t/T) = E[\tilde{x}(t/T)\underline{\tilde{x}}(t/T)^T] =$$

$$AP_f(t)A^T + (I-A)P_b(t)(I-A)^T \tag{141}$$

and optimization of the smoother will give "A", thus

$$\frac{\partial P(t/\tau)}{\partial A} = 0 \tag{142}$$

or

$$2AP_f(t) + 2(I-A)P_b(t)(-I) = 0. \tag{143}$$

Then A is given by

$$A = P_b(P_f+P_b)^{-1} \tag{144}$$

or

$$I-A = P_f(P_f+P_b)^{-1} \tag{145}$$

Also,

$$P(t/T) = P_b(P_f+P_b)^{-1}P_f(P_f+P_b)^{-1}P_b$$

$$+ P_f(P_f+P_b)^{-1}P_b(I+P_b)^{-1}P_f \tag{146}$$

By systematically combining factors in each of the two right-side terms of this equation, we arrive at a far more compact result, that is

$$P(t/T) = P_b(P_f+P_b)^{-1}P_f(I+P_b{}^{-1}P_f)^{-1}$$

$$+ P_f(P_f+P_b)^{-1}P_b(P_f{}^{-1}P_b+I)^{-1}$$

$$= P_b(P_g+P_b)^{-1}(P_f^{-1}+P_b^{-1})^{-1}$$

$$+ P_f(P_f+P_b)^{-1}(P_f^{-1}+P_b^{-1})$$

$$= (P_f^{-1}+P_b^{-1})^{-1}. \tag{147}$$

Thus,

$$P(t/T) = (P_f^{-1}+P_b^{-1})^{-1} \tag{148}$$

or

$$(P(t/T))^{-1} = P_f^{-1} + P_b^{-1}. \tag{149}$$

Consequently,

$$\hat{x}(t/T) = P_b(P_f+P_b)^{-1}\hat{x}_f(t) + P_f(P_f+P_b)^{-1}\hat{x}_b(t), \tag{150}$$

$$\hat{x}(t/T) = (P_f^{-1}+P_b^{-1})^{-1}P_f^{-1}\underline{\hat{x}}(t) + (P_f^{-1}+P_b^{-1})^{-1}P_b^{-1}\underline{\hat{x}}_b(t), \tag{151}$$

or

$$\hat{x}(t/T) = P(t/T)[P_f^{-1}(t)\underline{\hat{x}}(t) + P_b^{-1}(t)\underline{\hat{x}}_b(t)]. \tag{152}$$

E. The Choice of Initial Covariances' Values

The steps which should be considered in applying the Kalman filter into a problem are:

1. to construct a dynamic system model for the problem,

2. to calculate the initial values of the system covariance matrix, and

3. to find the behavior of the system covariance matrix with respect to time.

A dynamic system model for both cases of discrete and continuous situations was derived in Chapter II (pages 15 and 16). To calculate the initial values of the system covariance matrix, the standard deviation of the data from the static model should be estimated. Since the model in hand which is represented by Equation (15) for the discrete case or Equation (22) for the continuous case is neither controllable nor observable (Appendix A), it would not be possible to determine the standard deviation for each element of covariance matrix. The lack of observable data for α and c makes the calculation of the initial values of those elements of the system covariance matrix which are related to the standard deviation of α and c, almost impossible. The lack of controllability of the system may cause unstability of the parameters α and c.

To overcome the above complications which are due to the poor model, certain assumptions may be made. The unstability of the Kalman filter can be prevented by proper choice of initial covariance matrices' elements. To find the proper initial values, one has to solve the characteristic differential equation for a system covariance matrix of the form

$$\dot{P} = FP + PF^T + GQG^T. \tag{153}$$

where Q is a constant symmetric matrix. The time dependence of the system noise covariance matrix can be covered by suitable choice of observation noise covariance matrix. To simplify the problem one can assume very small values of Q's element with respect to the values of P's elements. Therefore;

$$\begin{aligned} &Q(\dot{J}) \ll P(i,\dot{J}) && \text{if } i = \dot{J} \\ &Q(i,\dot{J}) = 0 && \text{if } i \neq \dot{J}\ . \end{aligned} \tag{154}$$

From Equation (27) it can be shown that,

$$G = I \quad \text{(identity matrix)} \tag{155}$$

$$F = \begin{bmatrix} \alpha & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{156}$$

To solve Equation (153) assuming a diagonal initial system covariance matrix, the most important terms from the viewpoint of stability can be found for large value of t as follows:

$$P(1,1) = \frac{Q_{33}}{2\alpha^2}\, t + \left(P_{11}(0) - \frac{Q_{11}}{2\alpha} - \frac{Q_{33}}{4\alpha^3}\right) e^{2\alpha t} \tag{157}$$

$$P(1,2) = P(2,1) = P_{21}(0)e^{\alpha t} = P_{12}(0)e^{\alpha t} \tag{158}$$

$$P(1,3) = P(3,1) = -Q_{33}\frac{t}{\alpha} \tag{159}$$

$$P(2,2) = Q_{22}t + P_{22}(0) \tag{160}$$

$$P(2,3) = P(3,2) = 0 \tag{161}$$

$$P(3,3) = P_{33}(0) + Q_{33}t\ . \tag{162}$$

To choose the proper initial values to stabilize the Kalman filter; considering that we are interested in less deviation, the following relations between the elements of "P" and "Q" can be constructed

$$Q_{33} = \frac{P_{11}(0)}{T_{max}} \cdot 2\alpha^2 \cdot \frac{1}{3} \tag{163}$$

$$Q_{22} = -\alpha\,\frac{P_{22}(0)}{T_{max}} \tag{164}$$

$$Q_{11} = Q_{33} \tag{165}$$

where T_{max} is the maximum time under consideration. The initial values used in this study are

$$P_0 = \begin{bmatrix} s^2 y_0{}^2 & 0 & 0 \\ 0 & -\alpha s^2 & 0 \\ 0 & 0 & (\alpha^2 s^2 y_0{}^2) \end{bmatrix}, \tag{166}$$

$$Q = \begin{bmatrix} \frac{2}{3}\,\alpha^2\,\frac{P_{11}(0)}{T_{max}} & 0 & 0 \\ 0 & -\alpha\,\frac{P_{22}(0)}{T_{max}} & 0 \\ 0 & 0 & \frac{2}{3}\,\alpha^2\,\frac{P_{11}(0)}{T_{max}} \end{bmatrix} \quad (167)$$

and

$$R = s^2\hat{y}^2 \quad (168)$$

where R is the observation noise covariance and s^2 is the standard deviation of the data from static model. However, the chosen values yielded satisfactory estimates.

V. OPERATOR ERROR DATA COLLECTION

A. Definition of Operators Under Study

Data of operator error were extracted from "LER" by the Iowa State University, Engineering Research Institute Nuclear Safety Research Group (ERI-NSRG). The data were only extracted for those operators classified as follows:

Senior control operator:

The duties of a senior control operator are to instruct, train and assign work to personnel engaged in controlling the operation of the reactor-generator unit and associated equipments.

Control operator:

Control operators are responsible for the actual control and operation of the reactor-turbine-generator unit and associated equipments.

Equipment operators:

Equipment operators are responsible for the operation and inspection of individual equipment throughout the plant (assisting the control room operators), and the other operations of radwaste system.

Other operation personnel were not included in this study.

B. Data Collapsing for BWR's

Data related to operator errors for 24 BWR's power plants which are listed in Table 2 were available from the year 1972 up to 1978. The calculated availability is given in Table 3. Under the assumption that the availability of a plant in a month of a year is the same as the availability of that plant in that year, the data were collapsed for each 2 month of operation. The equivalent number of operator errors versus the age of equivalent plant is given in Table 4. However, the assumption of uniform availability distribution through a year is not rigorous, but is satisfactory due to lack of complete information about power plants availability.

The effect of the power level on operator errors can be estimated by considering only those plants with the same power. The data were collapsed for seven BWR's with power levels between (750 to 1000 MWe), and seven BWR's with power levels between (500 to 750 MWe), the list of power plants for each case and the collapsed data are given in Tables 5 through 8.

Table 2. The list of BWR's power plants

Name of Plants	Code
Peach Bottom 3	126B
Pilgrim 1	136B
Quad Cities 2	143B
Quad Cities 1	142B
Vermont Yankee 1	180B
Edwin Hatch 1	58B
Humboldt Bay	81B
Lacrosse	92B
Millstone 1	104B
Monticello 1	109B
Nine Mile Point 1	110B
Oyster Creek 1	119B
Peach Bottom 2	125B
Big Rock Point	016B
Brown's Ferry 2	024B
Dresden 1	054B
Brown's Ferry 1	023B
Brown's Ferry 3	25B
Brunswick 1	26B
Brunswick 2	27B
Cooper 1	43B
Dresden 2	55B
Dresden 3	56B
Duane Arnold	57B
Fitzpatrick 1	62B

Table 3. The availability of BWR's from 1972 to 1977

Name	Age at 72	Code	Initial Criticality	Commercial Criticality	Availability at:					
					72	73	74	75	76	77
Peach Bottom 3		126B	8,7,74	12,23,74			100.0	87.2	80.1	62.21
Pilgrim 1		136B	6,16,72	12,0,72	100.0	74.4	76.8	89.2	79.0	61.4
Quad cities 2		143B	4,26,72	3,10,73	65.0	87.7	84.9	52.3	83.8	87.9
Quad cities 1		142B	11,18,71	2,18,73	79.3	87.9	64.6	86.9	72.9	80.2
Vermont Yankee 1		180B	3,24,72	10,30,72	100.0	74.4	76.8	89.2	79.0	85.1
Edwin I Hatch 1		58B	9,12,74	12,31,75			3.9	76.2	88.0	66.3
Humboldt Bay 3	86.8	81B	2,16,63	8,0,63	83.0	89.3	84.4	84.7	46.8	0
LaCrosse	17.0	92B	7,11,67	9,13,69	70.6	58.0	89.4	71.7	49.8	33.7
Millstone 1	10.0	104B	11,26,70	3,0,71	60.0	46.4	80.9	78.1	83.5	89.6
Monticello	3.75	109B	12,10,70	6,30,71	83.9	71.3	79.6	73.6	93.7	79.9
Nine Mile Point 1	16.0	110B	9,5,69	12,0,69	72.0	80.0	72.9	77.8	94.4	55.1
Oyster Creek 1	20.5	119B	5,3,69	12,0,69	82.4	74.0	72.2	75.5	80.0	70.1
Peack Bottom 2		125B	9,16,73	7,5,74			92.7	77.4	70.0	55.2
Big Rock Point	76.0	16B	9,27,62	3,29,63	81.8	80.5	70.8	60.3	51.4	73.4
Brown's Ferry 2		24B	7,20,74	3,1,75			95.2	72.6	100.0	79.5

Table 3 (Continued)

Name	Age at 72	Code	Initial Criticality	Commercial Criticality	Availability at: 72	73	74	75	76	77
Dresden 1	104	54B	11,5,59	7,4,60	79.9	76.6	36.6	58.6	85.8	66.9
Brown's Ferry 1		23B	8,17,73	8,1,74		88.2	94.3	22.1	62.9	66.4
Brown's Ferry 3		25B	8,8,76	3,1,77						88.5
Brunswick 1		26B	11,8,76	3,18,77						56.7
Brunswick 2		27B	3,20,75	10,3,75				95.5	59.9	55.7
Cooper 1		43B	2,21,74	7,1,74			61.2	85.4	76.9	71.9
Dresden 2	NA	55B	1,7,70	6,19,72	64.6	95.0	66.8	57.8	78.3	71.2
Dresden 3	2.	56B	1,31,71	10,16,71	92.7	75.5	68.5	53.0	83.8	76.6
Duane Arnold		57B	3,23,74	2,1,75			64.0	83.0	79.7	78.9
Fitzpatrick 1		62B	10,17,74	7,28,75				74.0	74.3	68.4

Table 4. The average number of operator errors versus age for BWR's power plants

Age of the power plant	Average number of operator error per 2 months	Age of the power plant	Average number of operator error per 2 months
0	1.5	80	0.0
2	.69	82	0.0
4	.29	84	.5
6	.52	86	0.0
8	.776	88	0.0
10	.656	90	0.0
12	.83	92	1.0
14	.81	94	.5
16	.65	96	0.0
18	.77	98	1.
20	.85	100	.33
22	.39	102	0
24	.2	104	.33
26	.215	106	0.0
28	.416	108	0.0
30	.34	110	1.0
32	.4	112	0.0
34	.1	114	0.0
36	.2	116	0.0
38	.18	118	1.5
40	0.0	120	1.
42	.025	122	.5
44	.114	124	0.0
46	.285	126	0.0
48	.285	128	0.0
50	.5	130	1.
52	.05	132	0.0
54	.167	134	0.0
56	.143	136	0.0
58	1.3	138	1.
60	.83	140	0.0
62	.5	142	0.0
64	.33	144	0.0
66	0.0		
68	0.0		
70	1.		
72	0.0		
74	0.0		
76	.5		
78	0.0		

Table 5. The name and code of the BWR's with the power levels from 750 to 1000 MWe

Name of BWR's power plants ($750<P<1000$ MWe)	Code
Dresden 2	55B
Dresden 3	56B
Fitzpatrick	62B
Hatch 1	58B
Peack Bottom 2	125B
Quad Cities 1	142B
Quad Cities 2	143B

Table 6. The name and code of the BWR's with the power levels from 500 to 750 MWe

Name of BWR's power plants with ($500<P<750$ MWe)	Code
Duane Arnold	57B
Millstone 1	104B
Monticello	109B
Nine Mile Point 1	110B
Oyster Creek 1	119B
Pilgrim 1	136B
Vermont Yankee 1	180B

Table 7. The collapsed data for operator errors in BWR's with the power levels from 750 to 1000 MWe

Age of plant in month	Average number of operator failure per 2 month
0	0.8
2	0.38
4	0.24
6	0.56
8	0.975
10	0.38
12	0.5
14	0.87
16	1.56
18	0.56
20	1.16
22	0.97
24	0.54
26	0.17
28	0.0
30	0.43
32	0.175
34	0.5
36	0.25
38	0.25
40	0.0
42	0.25
44	0.0
46	0.0
48	0.0
50	0.5
52	1.0
54	1.0
56	0.0
58	0.0

Table 8. The collapsed data for operator errors in BWR's with the power levels from 500 to 750 MWe

Age of plant in month	Average number of operator error in 2 months
0	4.0
2	1.17
4	0.38
6	0.0
8	0.55
10	0.72
12	0.24
14	0.4
16	0.34
18	0.34
20	0.63
22	0.743
24	0.143
26	0.285
28	0.43
30	0.53
32	0.53
34	0.6
36	0.17
38	0.0
40	0.34
42	0.17
44	0.0
46	0.04
48	0.2
50	0.25
52	0.0
54	0.15
56	0.36
58	0.25
60	0.0
62	0.0
64	0.0
66	0.0
68	0.0
70	0.0
72	0.0

C. Data Collapsing for PWR's

Data related to operator errors for 30 PWR's power plants which are listed in Table 9, are available from the year 1972 up to 1978. The calculated availability of PWR's power plants are given in Table 10. The availability was considered to be uniformly distributed throughout the year. The collapsed data of the average number of operator failure are given in Table 11.

To study the effect of power, seven PWR's which are given in Table 12 with the power between 400 to 600 MWe, and 10 PWR's which are given in Table 13 with the power between 800 to 1200 MWe, were collapsed. The collapsed data of the two cases are given in Tables 14 and 15, respectively.

D. Problems Related to Data

The data calculated according to the procedure explained in Chapter V, pages 49-57 for two types of LWR's do not fit the exponential learning curve. The reasons for lack of fitness may be explained as follows:

1. Different operators under different management working with different power plants which are constructed by different vendors, can make different number of errors.

Table 9. The list of PWR's power plants the data of which were collapsed

Name of power plant	Code	Name of power plant	Code
San Onofre 1	150P	Turkey Point 4	177P
Haddam Neck 1	73P	Zion 1	193P
H. B. Robinson 2	78P	Zion 2	194P
Robert E. Ginna	147P	Crystal River 3	44P
Indian Point 2	83P	Arkansas 1	007P
Kewanee	89P	Calvert Cliffs 1	32P
Main Yankee	97P	Ft. Calhoun 1	64P
Millstone 2	105P	Yankee-Rowe	192P
Oconee 1	116P	Calvert Cliffs 2	33P
Oconee 2	117P		
Salem 1	148P		
Surry 1	168P		
Surry 2	169P		
Three Mile Island 1	174P		
Trojan 1	178P		
Oconee 3	118P		
Palisades 1	120P		
Point Beach 2	139P		
Prairie Island 1	140P		
Prairie Island 2	141P		
Rancho SeCo	144P		

Table 10. The age of PWR's power plants at 1972 and their availabilities from 1972 to 1977

Name	Code and Type	Commercial Start-up	Age at 72 in year	Availability at: 72	73	74	75	76	77
Arkansas 1	7P	12,19,74				68.3	80.8	59.9	76.8
Calvert Cliffs 1	32P	5,8,75					90.4	96.1	72.1
Cook 1	48P	8,27,75					74.0	74.3	76.1
Crystal River 3	44P	3,13,77							83.8
Davis Besse 1	45P	11,20,77							81.2
Farely 1	87P	12,1,77							68.8
Ft. Calhoun 1	64P	9,26,73			92.0	86.5	70.4	71.6	79.4
Ginna	147P	7,1,70	1.1	72.0	95.3	62.9	81.5	69.0	85.5
Hadam Neck 1	73P	1,1,68	3.55	90.8	58.1	96.2	88.7	87.3	83.9
Indian Point 2	83P	8,1,73			52.8	62.6	77.6	37.0	75.7
Indian Point 3	84P	8,30,76						78.8	74.9
Kewaunee	89P	6,1,74				78.6	90.8	84.9	79.9
Main Yankee	97P	12,28,72			89.4	69.8	82.8	95.6	82.2
Millstone 2	105P	12,26,75					79.9	95.4	65.7

Table 10 (Continued)

Name	Code and Type	Commercial Start-up	Age at 72 in year	Availability at: 72	73	74	75	76	77
Oconee 1	116P	7,15,73			92.0	62.2	79.6	60.8	62.3
Oconee 2	117P	9,9,74				71.1	75.5	64.5	60.7
Oconee 3	118P	12,16,74				47.9	79.5	71.2	74.8
Palisades	120P	12,31,71		61.1	47.6	7.6	66.8	59.0	91.4
Point Beach 1	138P	12,21,70	0.937	74.5	78.7	85.9	72.4	84.8	88.6
Point Beach 2	139P	10,1,72		14.6	94.8	82.7	96.8	91.8	86.0
Prairie Island 1	140P	12,16,73				48.9	89.8	79.5	85.1
Prairie Island 2	141P	12,21,74					97.3	78.5	89.2
Rancho SeCo	144P	4,17,75					48.6	57.8	77.1
Robinson 2	78P	3,7,71	0.384	88.7	79.1	86.2	74.5	87.5	85.2
Salem 1	148P	6,30,77							42.9
San Onofre 1	150P	1,1,68	3.0	80.2	63.7	94.9	88.1	71.2	63.7
Surry 1	168P	12,22,72		49.2	79.1	59.2	65.9	69.0	76.1
Surry 2	169P	5,1,73			98.9	62.7	81.3	53.4	68.3

Table 10 (Continued)

Name	Code and Type	Commercial Start-up	Age at 72 in year	Availability at: 72	73	74	75	76	77
Three Mile Island 1	174P	9,2,74				88.9	84.8	73.4	80.9
Trojan	178P	5,20,76							92.6
Turkey Point 3	176P	12,14,74				73.3	82.8	77.7	80.4
Turkey Point 4	177P	9,7,73			83.7	76.7	73.4	69.4	63.7
Yankee Rowe	192P	7,1,61	8.5	55.2	72.7	72.0	84.4	91.2	73.9
Zion 1	193P	12,31,73			75.2	59.0	80.0	64.2	74.2
Zion 2	194P	9,17,74				36.2	88.9	63.3	75.9

Table 11. The age of PWR's power plants versus the average number of operator errors per two months

Age of power plant in months	Average number of failures in months	Age of power plant in months	Average number of failures in 2 months
0	1.24	80	.0
2	.92	82	.0
4	.5	84	.0
6	.627	86	.0
8	.49	88	.0
10	.54	90	.0
12	.61	92	.0
14	.24	94	.0
16	.53	96	.0
18	.47	98	.0
20	.68	100	.0
22	.27	102	.5
24	.267	104	.0
26	.37	106	.0
28	.3	108	.0
30	.5	110	.0
32	.31	112	.0
34	.31	114	.0
36	.46	116	.0
38	.167	118	.0
40	.22	120	.0
42	.1	122	.0
44	.0	124	.0
46	.0	126	.0
48	.0	128	.0
50	.11	130	.0
52	.0	132	.0
54	.285	134	.0
56	.143	136	.0
58	.0	138	.0
60	.0	140	.0
62	.0	142	.0
64	.167	144	.0
66	.0	146	.0
68	.0	148	1.0
70	.0	150	.0
72	.0	152	.0
74	.0	154	.0
76	.0		
78	.33		

Table 12. The names, power rates, and the codes of PWR's with power from 400 to 600 MWe

Name	Electrical power rate MWe	Code
Ft. Calhoun	457	64P
Kewaunee	535	89P
Point Beach 1	497	138P
Point Beach 2	497	139P
Prairie Island 1	530	140P
San Onofre 1	430	150P
Ginna	490	64P

Table 13. The names, power rates, and the codes of PWR's with power levels from 800 to 1200 MWe

Name	Electrical power rate MWe	Code
Oconee 1	887	116P
Oconee 2	887	117P
Oconee 3	887	118P
Salem 1	1090	148P
Palisades 1	805	120P
Millstone 2	830	105P
Zion 1	1040	193P
Zion 2	1040	194P
Surry 1	822	168P
Surry 2	822	169P

Table 14. The average number of operator errors in 2 months for PWR's with power levels from 4-0 to 600 MWe

Age of PWR's power plant with power rates (400<P<600 MWe)	Average no. of operator errors in 2 months	Age of PWR's power plant with power rates (400<P<600 MWe)	Average no. of operator errors in 2 months
0	.5	78	.0
2	.75	80	.0
4	.25	82	.0
6	.575	84	.0
8	.175	86	.0
10	.0	88	.0
12	.24	90	.0
14	.3		
16	.55		
18	.28		
20	.23		
22	.0		
24	.33		
26	.66		
28	.5		
30	.66		
32	.0		
34	.4		
36	.8		
38	.2		
40	.0		
42	.0		
44	.0		
46	.0		
48	.0		
50	.25		
52	.0		
54	.0		
56	.0		
58	.0		
60	.0		
62	.0		
64	.5		
66	.0		
68	.0		
70	.0		
72	.0		
74	.0		
76	.0		

Table 15. The average number of operator errors in 2 months for PWR's with power levels from 800 to 1200 MWe

Age of PWR's power plants with power rates (800<P<1200 MWe)	Average number of operator failures in 2 months
0	1.4
2	1.0
4	.7
6	1.0
8	.66
10	1.0
12	.95
14	.5
16	.6
18	.5
20	.5
22	.5
24	.48
26	.3
28	.28
30	.72
32	.72
34	.34
36	.2
38	.26
40	0.0
42	0.0
44	0.0
46	0.0

2. Random delay in reporting the data, e.g., six months period report or the data which are reported one month or so after the event occurrence, can accumulate in a point and make false peak (the date of events is not given in LER).
3. The number of plants, the data of which were collapsed, decreases by age. For the ages above 9 years in PWR's and 10.5 years in BWR's the data is only available for one power plant. This inherent characteristic of the data available makes the number of collapsed data variable by age, which along with (1) and (2) can cause lots of fluctuation.
4. The availability calculated is for one year while the availability for each month is not known.

All the problems mentioned above can cause too much scattering which may completely cover the learning characteristic of the data. The only persuading factor among so many problems was the decreasing trend of the average number of operator errors by increasing the age of the power plants, which strongly supported the idea of learning of operators. To estimate learning model parameters special treatment of the data is required.

VI. SPECIAL SMOOTHING

A. Window Smoothing

One of the simple smoothing methods which can be used in presence of scattered data is the "window smoothing". If the slope of two adjacent points of the data is very sharp, then by this method one can decrease this slope. The shape of the window and the block diagram of window smoothing method are given in Figures 11 and 12, respectively. To use this method let us consider the following definitions:

θ = the discrete input time interval

T = the time delay for window sampling

W = the width of the window

TT = the total period of time, for which data is available.

Those parameters are related by the following relationships

$T = K'\theta$

$W = KT$

$TT = mT$

where K', K and m are some arbitrary integers.

It would be easy to derive the output-input relation from the block diagram given in Figure 10, that is,

$$U_0(iK'\theta) = 1/K \sum_{n=iK'}^{iK'+K} f(n\theta) \qquad (169)$$

The task is how to determine K and K'. To obtain the

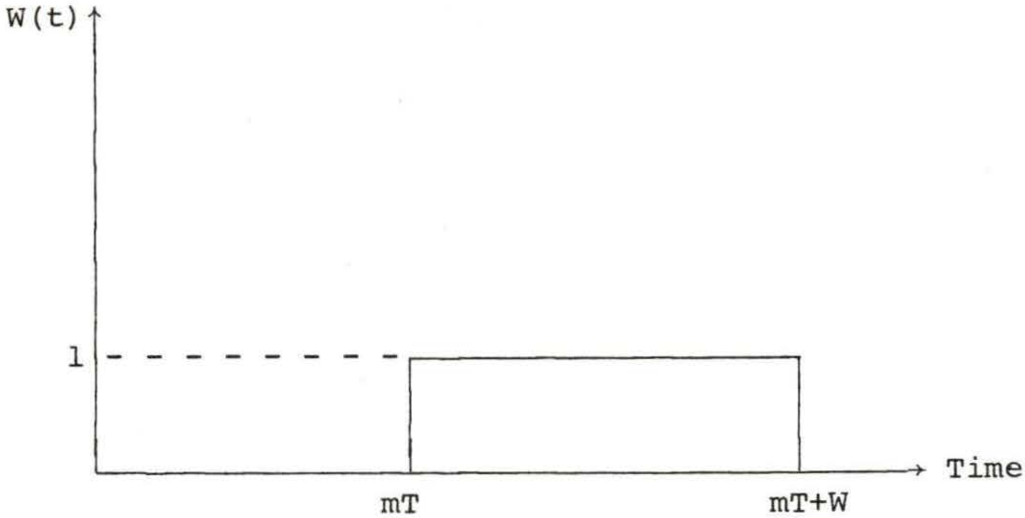

Figure 11. The shape of the window starting at mT with a width W

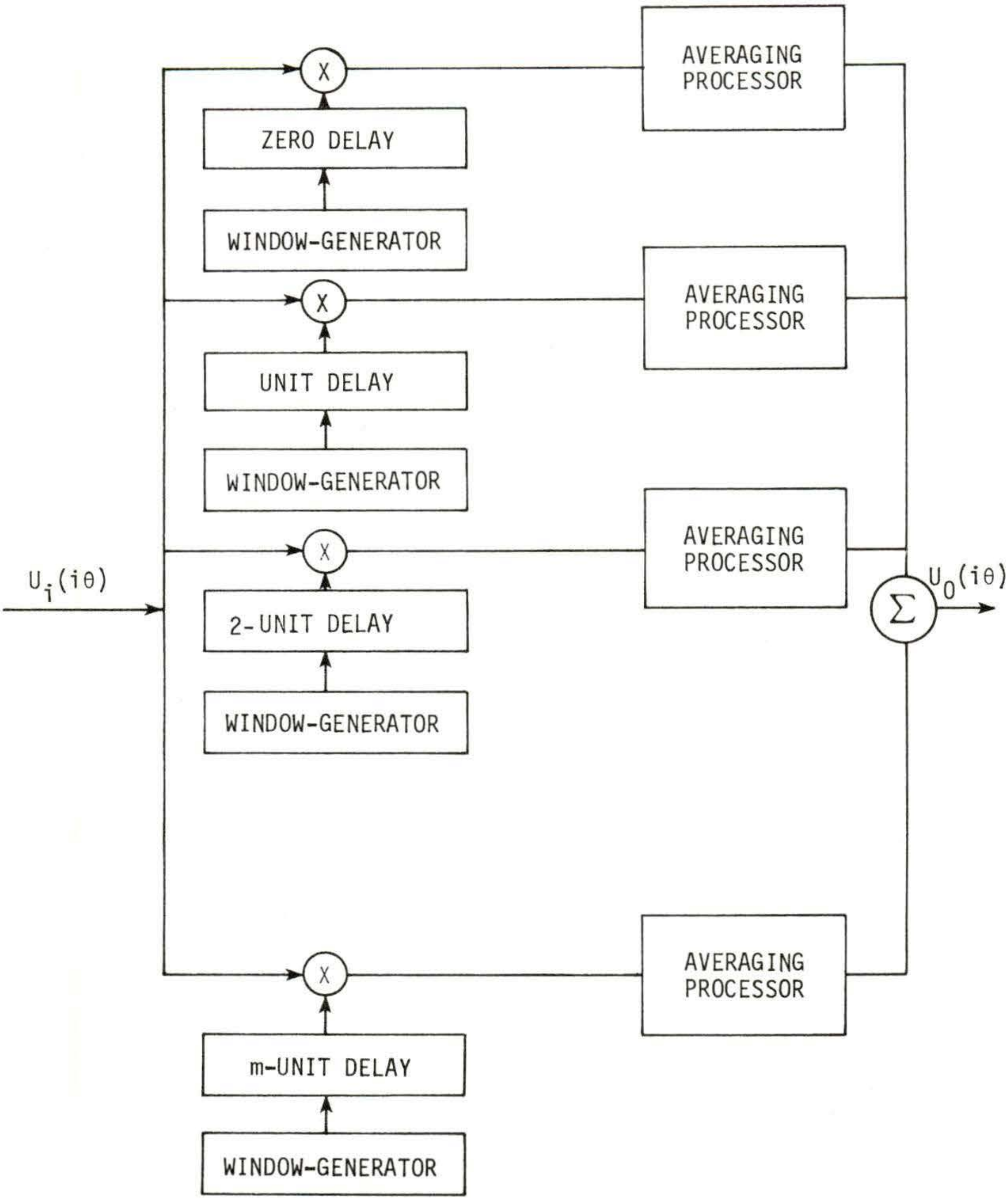

Figure 12. The block diagram of "window smoothing"

maximum number of smoothed data the value of K' can be chosen as unity. To calculate K, we need to know the smoothing factor which can be determined by natural slope of the data itself, e.g., one can make the limitation that the slope of two adjacent points shouldn't exceed a certain limit, or the difference between the two slopes of three adjacent points shouldn't exceed a certain limit.

However, by constructing a proper limitation the K value for each point can be calculated. In modern data smoothing, usually the value of K' is chosen as unity and the value of K is chosen as 3 and it is constant for all points. The process of forward and backward smoothing is possible. More study is needed for this type of smoothing and a computer code should be written which is beyond the scope of this study.

B. Integral Smoothing

The integral smoothing is used in this study to smooth the data because of simplicity. The idea behind it, is to eliminate the sharp slopes of a curve by integration. Considering a simple learning process as defined before in Equation (16).

$$\lambda = a(1+be^{-t/\tau}). \qquad (16)$$

By an integration process we define y_i as

$$y_i = \int_0^{t_i} a(1+be^{-t'\tau})dt' \qquad (170)$$

where "t_i" is the length of time from "0" to the occurrence of the "ith" point of data.

$$y_i = at_i - ab\tau e^{-t_i/\tau} + ab\tau \qquad (171)$$

or Z_i can be defined as

$$Z_i = y_i - at_i = ab\tau(1-e^{-t_i/\tau}). \qquad (172)$$

The function Z_i has the same form as the performance function defined in Equation (1) with the difference that y_c is equal to zero.

The problem with this type of smoothing is that, the value of "a" should be at least roughly estimated. Then a set of new data can be calculated as follows:

$$Z_m = \sum_{i=1}^{m} \lambda_i t_{i+1} - t_i - at_m \text{ for all } m = 1,n. \qquad (173)$$

The new data has a form of the performance model with initial performance equal to zero. The static and dynamic estimation process can be used for the new set of data.

In the statistical code developed in this study, the iteration on the value of "a" is considered. The initial value of "a" is assumed as the final value of λ, according to

$$\lim_{t\to\infty} \lambda = \lim_{t\to\infty} a(1+be^{-t/\tau}) = a. \tag{174}$$

VII. STATISTICAL CODE DEVELOPMENT

An Operator Error Exponential Model (OPEXM-K) code is developed for data smoothing, prediction and updating using the Kalman filter technique. The OPEXM-K code is especially developed for exponential operator error rates or performance. The least error squares (LE) and the impulse moment (IM) techniques are incorporated in the code for data smoothing. The OPEXM-K code is a simple statistical code developed to manipulate all types of exponential data. Two types of static estimations and Kalman dynamic filtering which were explained in Chapters III and IV, are included in this computer programming. A subroutine is developed to calculate the autocorrelation function of observation noise and to modify the static estimation in order to result in white noise process.

A brief discussion of each subroutine is given below. The program listing is given in Appendix C.

A. L-E Subroutine

The L-E subroutine is based on the least square estimation technique which was explained in Chapter III, Section A.1. This subroutine can manipulate both learning and performance or any type of exponential curve. The choice of initial values is of great importance. Testing this subroutine showed that if the initial values of exponential model is wrongly chosen or the data points are very scattered, the convergence can never be obtained due to oscillatory behavior of estimation. So, enough attention should be given to initializing the parameters.

B. I-M Subroutine

The I-M subroutine is based on the impulse moment updating technique which was explained in Chapter III, Section A.2. Testing this subroutine showed that the less scatter data for large values of time gives the better estimation. The initial value for this subroutine comes from the output of the L-E subroutine.

C. C-M Subroutine

There is always the question that, the Kalman filter equations obtained in Chapter IV are developed under the assumption of white noise process. There is no assurance that the deviation of data from the static model estimated by either least square or impulse moment updating is a white noise. So, the C-M subroutine modifies the static estimation parameters in such a way that the deviation from real data be almost a white noise process. Also, the autocorrelation of static error and the variance of the error related to static estimation can be calculated by this subroutine.

D. KALM Subroutine

The KALM subroutine is based on the Kalman filtering and prediction technique which was explained in Chapter IV. This subroutine is three dimensional subroutine written only for the case of exponential model. The forward and backward estimation, the covariance matrix related to forward and backward estimation and the optimal smooth estimation can be calculated in this subroutine.

E. R-M Subroutine

The R-M subroutine is an extra subroutine which is included in this program to calculate the time between failure from the failure rate. The reliability of an operator can be defined as;

$$R = 1 - e^{-\int_0^t \lambda(t)\,dt} , \tag{175}$$

thus

$$\frac{dR}{dt} = f(t) = -\lambda(t) e^{-\int_0^t \lambda(t)\,dt} , \tag{176}$$

The mean time between failure "MTBF" can be defined at any time as the first moment of f(t) in the form of;

$$\mathrm{MTBF}(t_K) = \int_0^{t_K} t f(t)\,dt \ . \tag{177}$$

The time between failure at time t_K can be defined as

$$\mathrm{TBF}(t_k) = (t_k \cdot \mathrm{MTBF}(t_k) - t_{k-1} \cdot \mathrm{MTBF}(t_{k-1}))/(t_k - t_{k-1}) . \tag{178}$$

The above procedure is written in discrete form in subroutine R-M. The time between failure obtained from this subroutine is much smoother than the real life time between failure data.

VIII. CONSTRUCTION OF OPERATOR ERROR RATE MODEL

A. Static Operator Error Rate Model

1. BWR's operator error rate model

The data points are smoothed by the integration smoothing techniques for three cases of interest. The smoothed data and related "A" value for each case are given in Tables 16, 17, and 18, respectively.

The static estimations from impulse moment updating and least square techniques are given in Table 19. In the least square techniques result, the initial value for the smoothed data can be used for calculation of the delay time before the learning process. To do that, consider Equation (5) and Figure 4. Thus, the integration smoothing can be defined as

$$z_i = y_i - at_i = \int_0^{\theta} a(1+b)dt' + \int_{\theta}^{t_i} a(1+be^{-ct'})dt' - at_i \tag{179}$$

instead of Equation (172).

Therefore,

$$z_i = ab\theta + ab/c(e^{-c\theta} - e^{-cT}). \tag{180}$$

Assuming $c\theta \ll 1$ or $\theta \ll \tau$ we get

$$e^{-c\theta} \simeq 1 \quad \text{if} \quad c\theta \ll 1$$

$$z_i = ab\theta + ab/c(1-e^{-cT}). \tag{181}$$

Table 16. The smoothed data for all BWR's with A = .5 error/4 months

Time	Z_i	Time	Z_i
0	0.0	76	6.86
4	1.69	80	6.36
8	2.0	84	6.36
12	2.93	88	5.86
16	4.19	92	6.86
20	5.9	96	7.36
24	5.7	100	7.19
28	5.65	104	7.02
32	5.9	108	7.52
36	5.78	112	7.02
40	5.3	116	6.52
44	5.2	120	8.5
48	5.5	124	8.5
52	5.25	128	8.0
56	6.2	132	8.5
60	7.58	136	8.0
64	6.86	140	8.5
68	7.36	144	8.0
72	6.86		

Table 17. The smoothed data for BWR's with power rate from 750 to 1000 MWe (initial value of A = .2 errors/ 4 months)

Time	Z_i
0.0	0.0
4.0	1.0
8.0	1.58
12.0	2.73
16.0	3.9
20.0	5.82
24.0	7.75
28.0	8.26
32.0	8.49
36.0	8.96
40.0	9.26
44.0	9.3
48.0	9.1
52.0	9.4
56.0	11.2
60.0	11.0

Table 18. The smoothed data for BWR's with power levels from 500 to 750 MWe (A = .15 error/4 months)

Time	Z_i
0.0	0.0
4.0	5.02
8.0	5.25
12.0	6.37
16.0	6.86
20.0	7.39
24.0	8.61
28.0	8.89
32.0	9.7
36.0	10.7
40.0	10.7
44.0	11.1
48.0	10.9
52.0	11.3
56.0	11.3
60.0	11.7
64.0	11.6
68.0	11.4
72.0	11.3

Table 19. The static learning parameters for BWR's for three different cases

Title of data	Techniques	A	B	C	θ	A.B
All BWR's	Least square method	0.112	2.18	.0306	3.9	0.244
	Impulse moment updating	0.125	1.96	.0306	0	0.245
Those BWR's with power 500 to 750 MWe	Least square method	0.004	206.8	.0735	1.5	0.827
	Impulse moment updating	0.0377	22.00	.0735	0	0.829
Those BWR's with power 750 to 1000 MWe	Least square method	0.0627	7.68	.0438	-1.3	0.4815
	Impulse moment updating	0.0417	11.56	.0438	0	0.482

To compare Equation (181) with the performance equation defined in Equation (1), the value of θ can be calculated approximately from

$$\theta = \frac{y_c}{ab} \, . \tag{182}$$

The static estimation model for each case is given in Table 20.

Table 20. The smoothed data for all PWR's, the initial value of "A" is chosen as 0.08 errors/4 months

Time	Z_i	Time	Z_i
0.0	0.0	6.80	9.42
4.0	2.08	72.0	9.34
8.0	3.12	76.0	9.26
12.0	4.8	80.0	9.55
16.0	4.84	84.0	9.47
20.0	5.76	88.0	9.38
24.0	6.64	92.0	9.31
28.0	7.15	96.0	9.23
32.0	7.91	100.0	9.15
36.0	8.45	104.0	9.57
40.0	9.00	128.0	9.11
44.0	9.24	152.0	9.61
48.0	9.16	156.0	9.57
52.0	9.19		
56.0	9.38		
60.0	9.46		
64.0	9.38		

To explain the results given in Table 19 is better to consider the expression

$$\lambda = a + abe^{-ct} + \varepsilon(t) \qquad (183)$$

where $\varepsilon(t)$ is the scattering of the data and comparable to "a".

For values of "t" not too large considering "b" much larger than unity, the following approximation can be made,

$$\lambda = abe^{-ct} \; . \qquad (184)$$

For the larger values of "t" the error rate equation can be approximated as;

$$\lambda = a + \varepsilon(t) . \qquad (185)$$

Assuming that "a" is estimated from the final value of "λ", thus the expectation of "a" can deviate from the real mean value of "a" by the expectation of "ε". Anyway, to compare the different learning parameter estimations it is sufficient to compare the product of "ab" and "c". The following results can be obtained from Table 19.

a. In BWR's with power range between 500 to 750 MWe, more error is expected but, the operator learning speed is faster than the other BWR's due to the larger value of "c". The final error rate cannot be exactly determined due to high scattering for large values of "t".

b. In BWR's with power range between 750 to 1000 MWe, less error is expected, but the learning speed is slower than in the case of 500 to 750 MWe BWR's. The final error rate "a" can not be determined exactly due to high scattering for large values of "t".

c. The estimation of all BWR's shows the least number of errors but, the slowest speed in learning.

d. The expected range for the final error rate can be estimated for all BWR's as

$$0.004 \leq A \leq 0.125 \quad \text{errors/month.}$$

e. The assumption of delay time in learning process leads to better estimation.

f. The negative value of delay time estimation for BWR's power between 750 to 1000 MWe may be due to lack of data at early stage of start-up experience which can result in underestimating the age of the plant.

2. <u>PWR's operator error rate model</u>

The data are smoothed by the integration smoothing techniques for three cases of interest. The smoothed data and related "A" value are given in Tables 20, 21, and 22 respectively. The static estimation result from programming for all the cases are given in Table 23. As the process

Table 21. The final data for PWR's with power levels from 400 to 600 MWe (the initial value of "A" is chosen equal 0.0 errors/4 months)

Time	Z_i	Time	Z_i
0	0.0	68.0	8.15
4	1.25	72.0	8.15
8	2.075	76.0	8.15
12	2.25	80.0	8.15
16	2.79	84.0	8.15
20	3.62	88.0	8.15
24	3.85	92.0	
28	4.84		
32	6.0		
36	6.4		
40	7.4		
44	7.4		
48	7.4		
52	7.65		
56	7.65		
60	7.65		
64	7.65		

Table 22. The final data for PWR's with power levels from 800 to 1200 MWe (the initial value of "A" is 0.1 errors/4 months)

Time	Z_i
0.0	0.0
4.0	2.3
8.0	3.9
12.0	5.46
16.0	6.89
20.0	7.81
24.0	8.71
28.0	8.39
32.0	9.29
36.0	10.25
40.0	10.6
44.0	10.5
48.0	10.4

Table 23. The learning parameters for PWR's for three cases of interest

Title of data	Technique	A	B	C	θ	A.B
All PWR plants	Least square method	0.0231	21.13	.051	0.1	0.4881
	Impulse moment updating	0.0212	22.977	.051	0.0	0.4871
Those PWR's with power rate between 400 to 600 MWe	Least square method	0.00243	140.6	.042	-.43	0.3416
	Impulse moment updating	-	-	-	-	-
Those PWR's with power rate between 800 to 1200 MWe	Least square method	0.0224	34.0	.07324	.015	0.7616
	Impulse moment updating	0.0224	34.0	.07324	0.0	0.7616

explained in Chapter VIII, Section A.1, the comparison between different cases are as follows:

a. The PWR's with power between 400 to 600 MWe shows much lower number of errors than other cases. The negative delay time may be due to lack of data at early stages of start-up experience, which can result in underestimating the age of the plant. The very low value of "A" leads us to the conclusion that, the number of operator errors in this case can decrease to one error in thirty-four years. The impulse moment updating did not give an acceptable result for "A" ($A<0$), thus it is not considered as a good estimation.

b. The PWR's with power between 600 to 1200 MWe show higher number of errors than any other case. The value of "A" leads us to the conclusion that even for very old plants we expect to see at least one error in each four years.

c. The estimation for average PWR's; neglecting the effect of power, gives us number of errors more than PWR's with power between 400 to 600 MWe and less than PWR's with power between 800 to 1200 MWe which is confidentially acceptable.

B. Dynamic Operator Error Rate Model

1. BWR's operator error rate model

The initial value of parameters from static estimation are used for Kalman filtering. The output of the Kalman filter is the dynamic behavior of the model. In three cases of BWR study, the values of A, B and C are given in Tables 24, 25 and 26. The graphs of real data and two dynamic estimators are also given in Figures 13, 14 and 15, respectively.

Table 24. Two dynamic estimation for learning parameters of an average BWR

Initial static model	Impulse moment updating			Least square method		
Time	A	B	C	A	B	C
0.0	.1248	1.963	0.03062	0.119	.9381	0.05325
4.0	.1248	3.352	0.03062	0.1119	1.268	0.05457
8.0	.1248	2.978	0.08709	0.1119	1.430	0.05443
12.0	.1248	3.038	0.0846	0.1119	1.798	0.05397
16.0	.1248	3.314	0.08153	0.1119	2.227	0.05097
20.0	.1248	3.665	0.07566	0.1119	2.576	0.04334
24.0	.1248	3.666	0.07232	0.1119	2.475	0.03742
28.0	.1248	3.635	0.07119	0.1119	2.353	0.03482
32.0	.1248	3.631	0.07071	0.1119	2.287	0.03426
36.0	.1248	3.615	0.07132	0.1119	2.227	0.03536
40.0	.1248	3.58	0.07232	0.1119	2.153	0.03675
44.0	.1248	3.492	0.07338	0.1119	2.032	0.03788
48.0	.1248	3.442	0.07341	0.1119	1.957	0.03804
52.0	.1248	3.472	0.03708	0.1119	1.972	0.03791
56.0	.1248	3.467	0.07293	0.1119	1.979	0.0381
60.0	.1248	3.701	0.0732	0.1119	2.215	0.0384
64.0	.1248	4.053	0.07355	0.1119	2.548	0.0386
68.0	.1248	4.050	0.07325	0.1119	2.536	0.03826
72.0	.1248	4.153	0.07291	0.1119	2.622	0.03796
76.0	.1248	4.108	0.07291	0.1119	2.559	0.03790
80.0	.1248	4.098	0.07299	0.1119	2.533	0.03798
84.0	.1248	4.016	0.07325	0.1119	2.442	0.03813
88.0	.1248	3.991	0.07325	0.1119	2.418	0.03815
92.0	.1248	3.922	0.07315	0.1119	2.365	0.03816
96.0	.1248	4.092	0.07311	0.1119	2.543	0.03826
100.0	.1248	4.24	0.0732	0.1119	2.696	0.03843
104.0	.1248	4.279	0.03714	0.1119	2.745	0.03847
108.0	.1248	4.299	0.03707	0.1119	2.777	0.03848
112.0	.1248	4.404	0.07307	0.1119	2.883	0.03852
116.0	.1248	4.367	0.07293	0.1119	2.867	0.03848
120.0	.1248	4.344	0.07285	0.1119	2.871	0.03850
124.0	.1248	4.657	0.07301	0.1119	3.155	0.03837
128.0	.1248	4.747	0.0729	0.1119	3.228	0.03808
132.0	.1248	4.740	0.0728	0.1119	3.216	0.03785
136.0	.1248	4.805	0.07263	0.1119	3.262	0.03747
140.0	.1248	4.776	0.07266	0.1119	3.229	0.03737
142.0	.1248	4.812	0.07249	0.1119	3.253	0.03701
144.0	.1248	4.781	0.07260	0.1119	3.223	0.03704

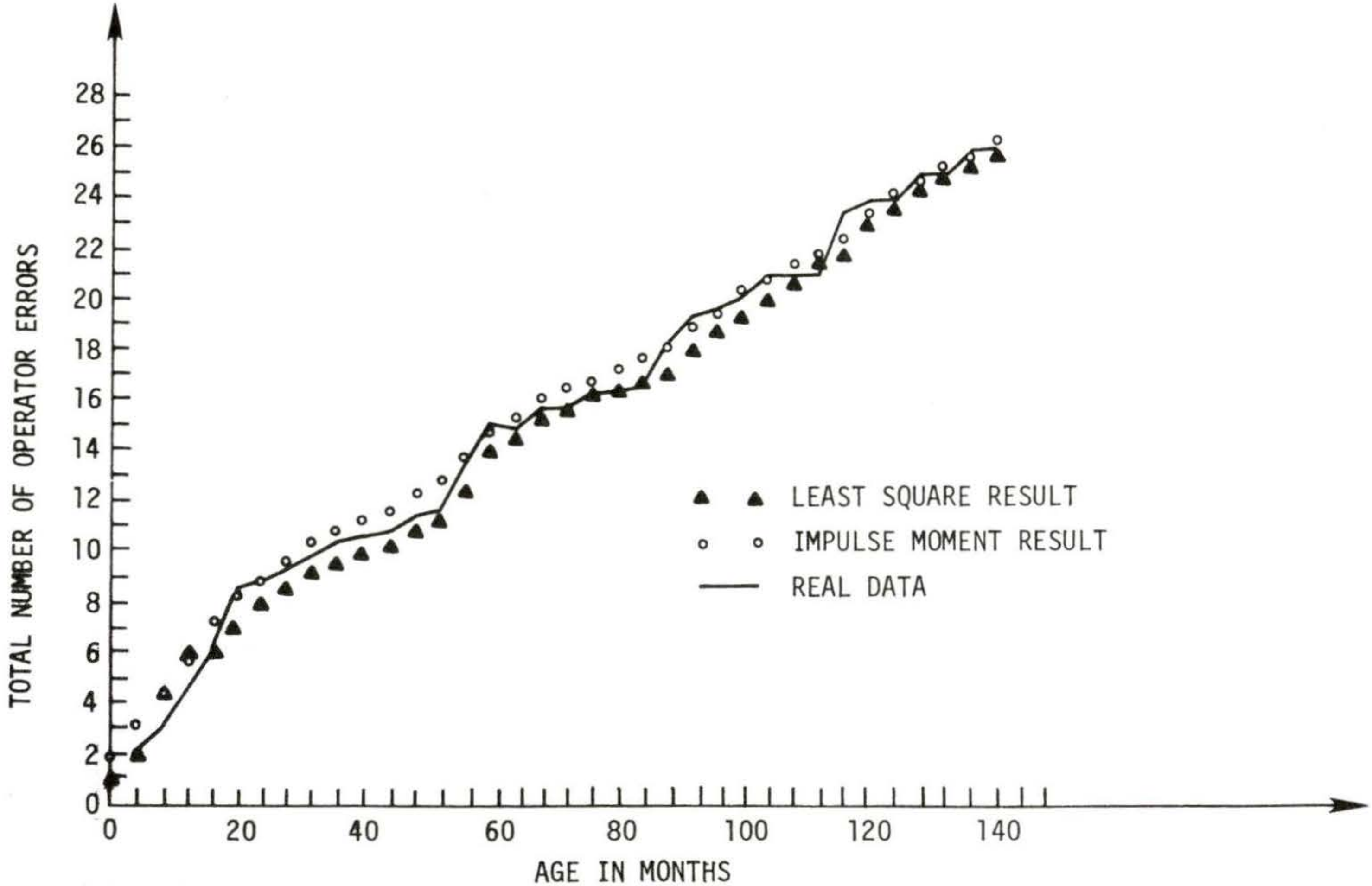

Figure 13. Dynamic estimation of operator errors for total data collapsed of BWR's

Table 25. Dynamic estimation for BWR's (500<P<740)

Initial static model	Impulse moment updating			Least square method		
Time	A	B	C	A	B	C
0.0	.03766	21.97	.07354	.004068	106.9	0.1292
4.0	.03766	33.04	.07354	.004068	161.2	0.1169
8.0	.03766	31.74	.1732	.004068	159.4	0.1142
12.0	.03766	32.52	.168	.004068	169.3	0.1098
16.0	.03766	33.59	.1659	.004068	177.6	0.1076
20.0	.03766	34.96	.1636	.004068	189.0	0.1054
24.0	.03766	37.39	.1605	.004068	209.3	0.1027
28.0	.03766	38.74	.1585	.004068	219.5	0.1005
32.0	.03766	40.47	.1563	.004068	232.3	0.09786
36.0	.03766	42.24	.1536	.004068	244.6	0.0946
40.0	.03766	42.93	.1522	.004068	248.2	0.09279
44.0	.03766	43.63	.1508	.004068	252.2	0.09115
48.0	.03766	43.87	.1504	.004068	253.3	0.0906
52.0	.03766	44.3	.1496	.004068	256.5	0.08961
56.0	.03766	44.51	.1493	.004068	258.5	0.0890
60.0	.03766	44.98	.1484	.004068	262.8	0.08789
64.0	.03766	44.98	.1484	.004068	263.4	0.0877
68.0	.03766	44.89	.1485	.004068	263.2	0.08777
72.0		44.78	.1487	.004068	262.7	0.08789

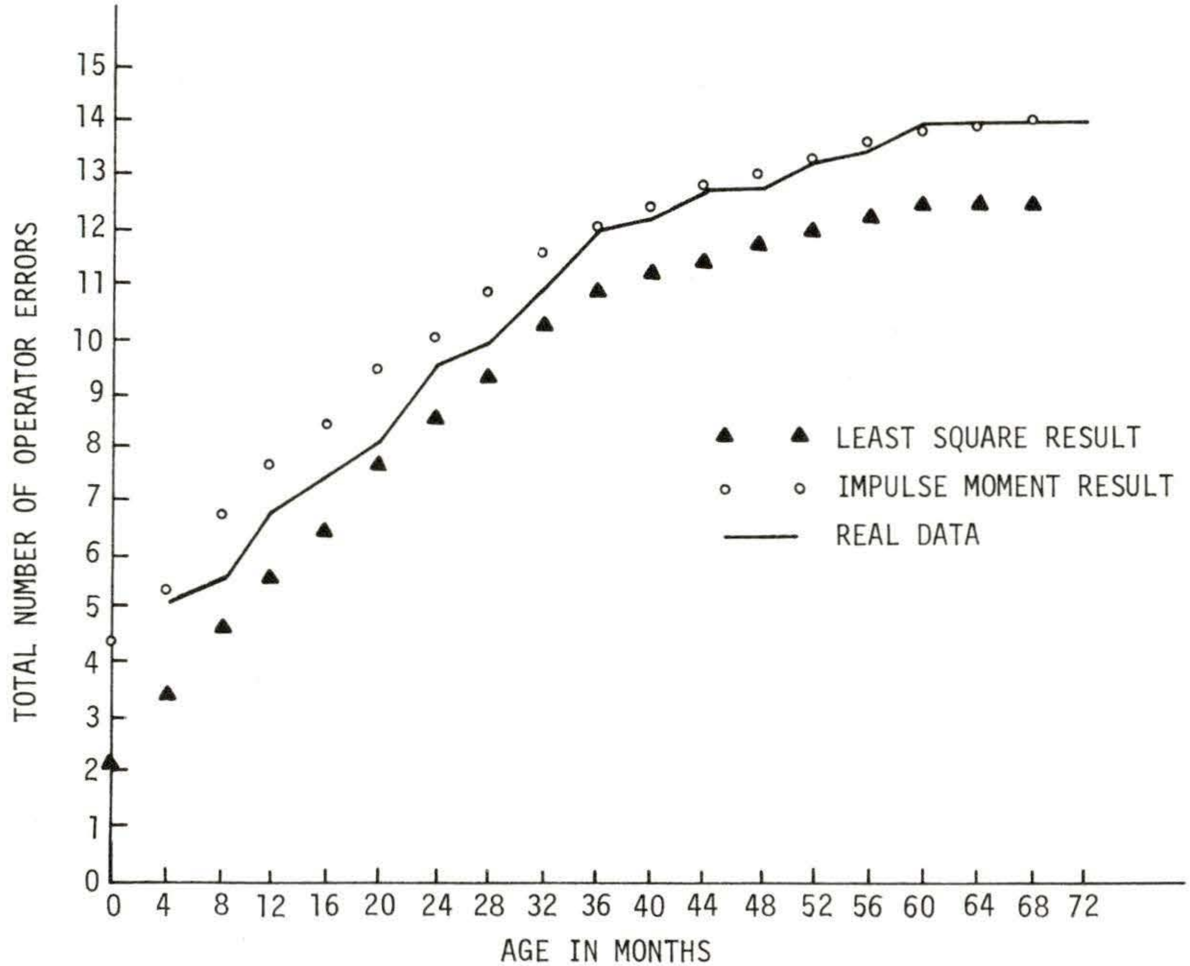

Figure 14. Dynamic estimation of operator error in BWR's (500<P<750 MWe) versus age

Table 26. Dynamic estimation for BWR's (750<P<1000)

Initial static model Time	Impulse moment updating A	B	C	Least square method A	B	C
0.0	0.04171	11.56	0.04382	0.06268	1.229	0.03936
4.0	0.04171	6.324	0.04382	0.06268	1.856	0.05066
8.0	0.04171	3.286	0.05196	0.06268	2.066	0.05625
12.0	0.04171	5.303	0.05453	0.06268	3.534	0.06126
16.0	0.04171	6.957	0.05325	0.06268	4.743	0.06102
20.0	0.04171	8.784	0.0465	0.06268	6.115	0.05467
24.0	0.04171	9.362	0.03373	0.06268	6.544	0.0403
28.0	0.04171	8.517	0.0244	0.06268	5.956	0.02975
32.0	0.04171	7.661	0.02059	0.06268	5.417	0.02618
36.0	0.04171	7.193	0.01938	0.06268	5.161	0.0256
40.0	0.04171	6.983	0.01973	0.06268	5.049	0.02664
44.0	0.04171	6.88	0.02073	0.06268	4.951	0.02808
48.0	0.04171	6.815	0.02167	0.06268	4.847	0.02915
52.0	0.04171	6.997	0.02186	0.06268	4.969	0.02918
56.0	0.04171	7.372	0.02193	0.06268	5.31	0.02894
60.0	0.04171	7.085	0.02274	0.06268	5.146	0.02950

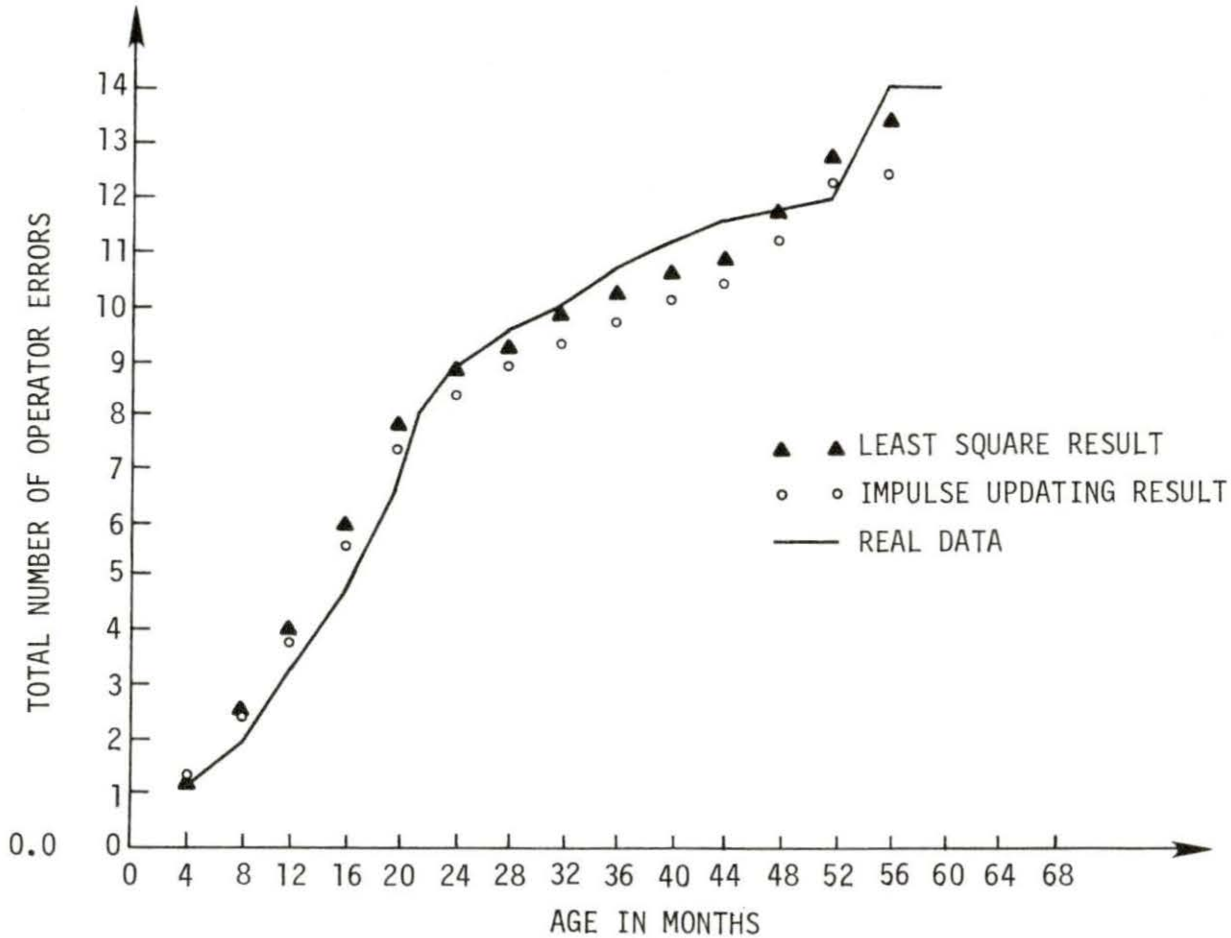

Figure 15. Dynamic estimation of BWR's (750<P<1000 MWe) versus age

By comparison among dynamic estimation for three cases, the following results can be obtained.

1. The value for "C" is decreasing for all three cases of study which can be related to the inertia of learning. In other words, the operator does not improve as fast as he improved before. This can be due to operator negligence after becoming confident on the job.

2. The product of "A.B" is increasing with time for three cases of study but the effect of time (age) is more dominant, so the number of operator errors per unit time will tend to decrease. However, the increasing trend of (A.B) products shows that, unfortunately, the expert operator does not pay enough attention to the task, as he had paid before.

3. The response of learning with respect to power in BWR's is kind of a confusing issue due to different behavior of learning parameters. The following behavior can be seen in dynamic estimation of different power BWR's, even though the number of operator errors is in the same range;

a. Learning in small power BWR's is faster than large BWR's (the value of "C" for small BWR's is larger than the value of "C" for large BWR's).

b. The product (A.B) in large BWR's is smaller than small power BWR's. This behavior can be explained as the optimum stress on operator in the case of BWR's with power between 750 MWe to 1000 MWe (the operator is aware of the importance of operation).

2. PWR's operator error rate model

The initial value of the parameters from static estimation is used for Kalman filtering. The output of Kalman filter is the dynamic behavior of the model. In three cases of the PWR study, the values of A, B, and C are given in Tables 27, 28 and 29. The graphs of real data and two dynamic estimations are also given in Figures 16, 17 and 18, respectively.

By comparison among dynamic estimation for three cases, the following results can be obtained:

1. The average behavior data for PWR's, which is obtained from collapsing the total PWR data, is not time dependent. The only scattering in dynamic estimation of parameters comes from random noise scattering with variances less than covariance matrix diagonal elements. Therefore, the data can be best approximated by white noise scattering, which is not time dependent. This allows the construction of a constant parameter learning curve for PWR's.

2. The effect of power in PWR's is very important.

The PWR's with power between 400 MWe to 600 MWe have very good learning characteristics. The operator has the ability to reduce the number of errors to one error in 30 years. Since the average value of "τ" is 2 years for this PWR's, it can be expected that constant error rate happens after 10 years.

To compare Figure 18 with Figure 17 and Table 29 with Table 28, the effect of the increase of power can be simply observed by the higher values of A, (A.B) and total number of errors. Comparing the values of C for both cases shows slightly faster learning in large PWR's than small PWR's.

Table 27. Dynamic estimation for PWR's

Initial static model	Impulse moment updating			Least square method		
Time	A	B	C	A	B	C
0	.02118	23.01	0.05092	.02314	7.498	0.06964
4	.02118	24.46	0.05092	.02314	16.73	0.08645
8	.02118	20.00	0.08058	.02314	17.17	0.07919
12	.02118	22.14	0.07325	.02314	19.38	0.07224
16	.02118	21.80	0.07202	.02314	19.00	0.07076
20	.02118	22.82	0.07013	.02314	19.90	0.06880
24	.02118	24.21	0.06852	.02314	21.13	0.06711
28	.02118	25.21	0.06733	.02314	21.99	0.06584
32	.02118	26.43	0.0659	.02314	23.04	0.06429
36	.02118	27.28	0.06449	.02314	23.76	0.06277
40	.02118	27.93	0.06307	.02314	24.29	0.06125
44	.02118	28.13	0.0622	.02314	24.41	0.06035
48	.02118	27.95	0.06207	.02314	24.22	0.06024
52	.02118	27.83	0.06205	.02314	24.08	0.06025
56	.02118	27.87	0.06194	.02314	24.09	0.06016
60	.02118	27.83	0.06195	.02314	24.03	0.06019
64	.02118	27.65	0.06207	.02314	23.84	0.06033
68	.02118	27.56	0.06211	.02314	23.73	0.06039
72	.02118	27.37	0.06219	.02314	23.53	0.06047
76	.02118	27.22	0.06223	.02314	23.37	0.06051
80	.02118	27.49	0.06214	.02314	23.6	0.06043
84	.02118	27.38	0.06217	.02314	23.48	0.06046
88	.02118	27.20	0.06221	.02314	23.28	0.06051
92	.02118	26.86	0.06227	.02314	22.91	0.06058
96	.02118	26.86	0.06227	.02314	22.91	0.06058
100	.02118	26.81	0.06228	.02314	22.84	0.06059
104	.02118	27.19	0.06221	.02314	23.17	0.06052
128	.02118	26.85	0.06226	.02314	22.84	0.06059
152	.02118	27.26	0.06226	.02314	23.16	0.06059
156	.02118	27.27	0.06226	.02314	23.16	0.06059

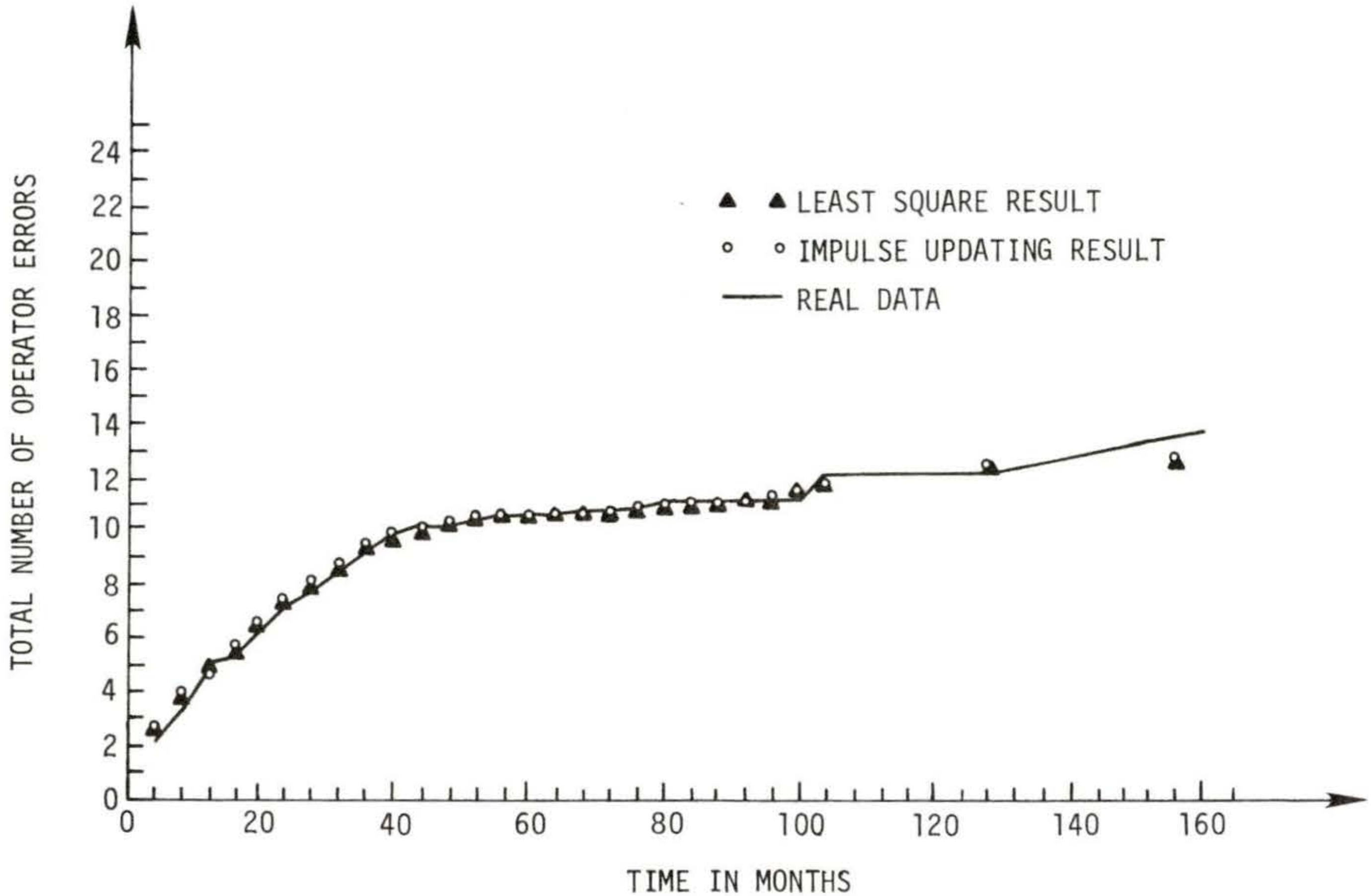

Figure 16. Dynamic estimation for average PWR's power plant

Table 28. Dynamic estimation for PWR's (400<P<600 MWe)

Initial static estimation Time	Least square method A	B	C
0	0.002434	26.68	0.04733
4	0.002434	68.32	0.05934
8	0.002434	81.51	0.05617
12	0.002434	73.98	0.05685
16	0.002434	79.98	0.05728
20	0.002434	98.97	0.05835
24	0.002434	107.4	0.05844
28	0.002434	128.2	0.05852
32	0.002434	148.6	0.05664
36	0.002434	154.2	0.05304
40	0.002434	158.5	0.04819
44	0.002434	154.1	0.04558
48	0.002434	150.3	0.04477
52	0.002434	148.6	0.04442
56	0.002434	146.6	0.04479
60	0.002434	145.0	0.04511
64	0.002434	144.0	0.04525
68	0.002434	146.4	0.04489
72	0.002434	146.4	0.04484
76	0.002434	146.0	0.04494
80	0.002434	145.4	0.04504
84	0.002434	144.8	0.04501
88	0.002434	144.3	0.04509
92	0.002434	144.1	0.04510

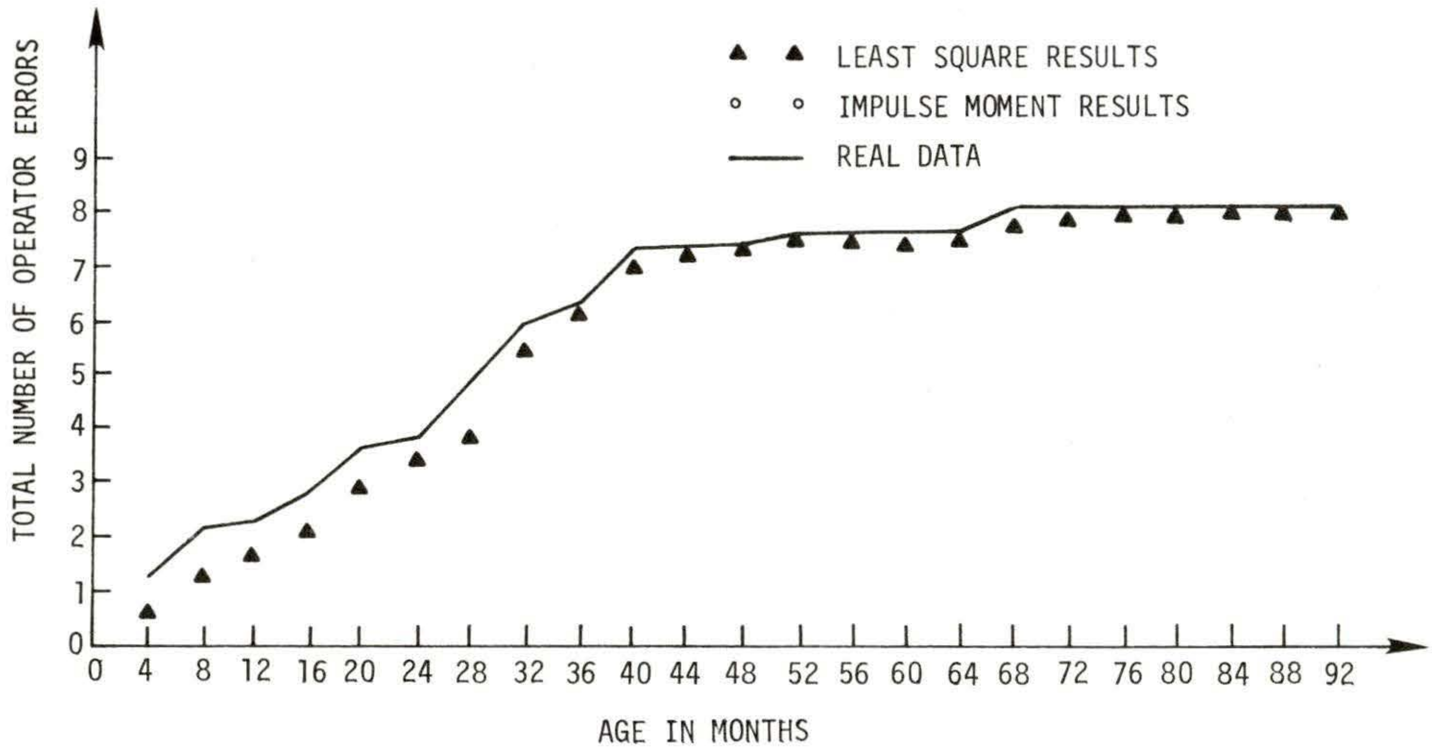

Figure 17. Dynamic estimation for PWR's with power levels from 400 to 600 MWe

Table 29. Dynamic estimation for PWR's (800<P<1200 MWe)

Initial estimation method / Time	Impulse moment updating A	B	C	Least square method A	B	C
0	.0229	33.26	.07324	.0224	11.94	.08187
4	.0229	25.4	.07324	.0224	25.98	.07324
8	.0229	21.62	.08117	.0224	22.12	.08113
12	.0229	23.26	.07195	.0224	23.79	.07191
16	.0229	24.55	.06425	.0224	25.11	.06422
20	.0229	24.94	.05941	.0224	25.51	.05938
24	.0229	25.24	.05648	.0224	25.83	.05644
28	.0229	24.43	.05703	.0224	25.00	.05698
32	.0229	25.14	.0555	.0224	25.72	.05545
36	.0229	26.22	.05377	.0224	26.83	.05371
40	.0229	26.5	.05300	.0224	27.12	.05295
44	.0229	26.13	.05317	.0224	26.74	.05311
48	.0229	25.79	.05341	.0224	26.40	.05334

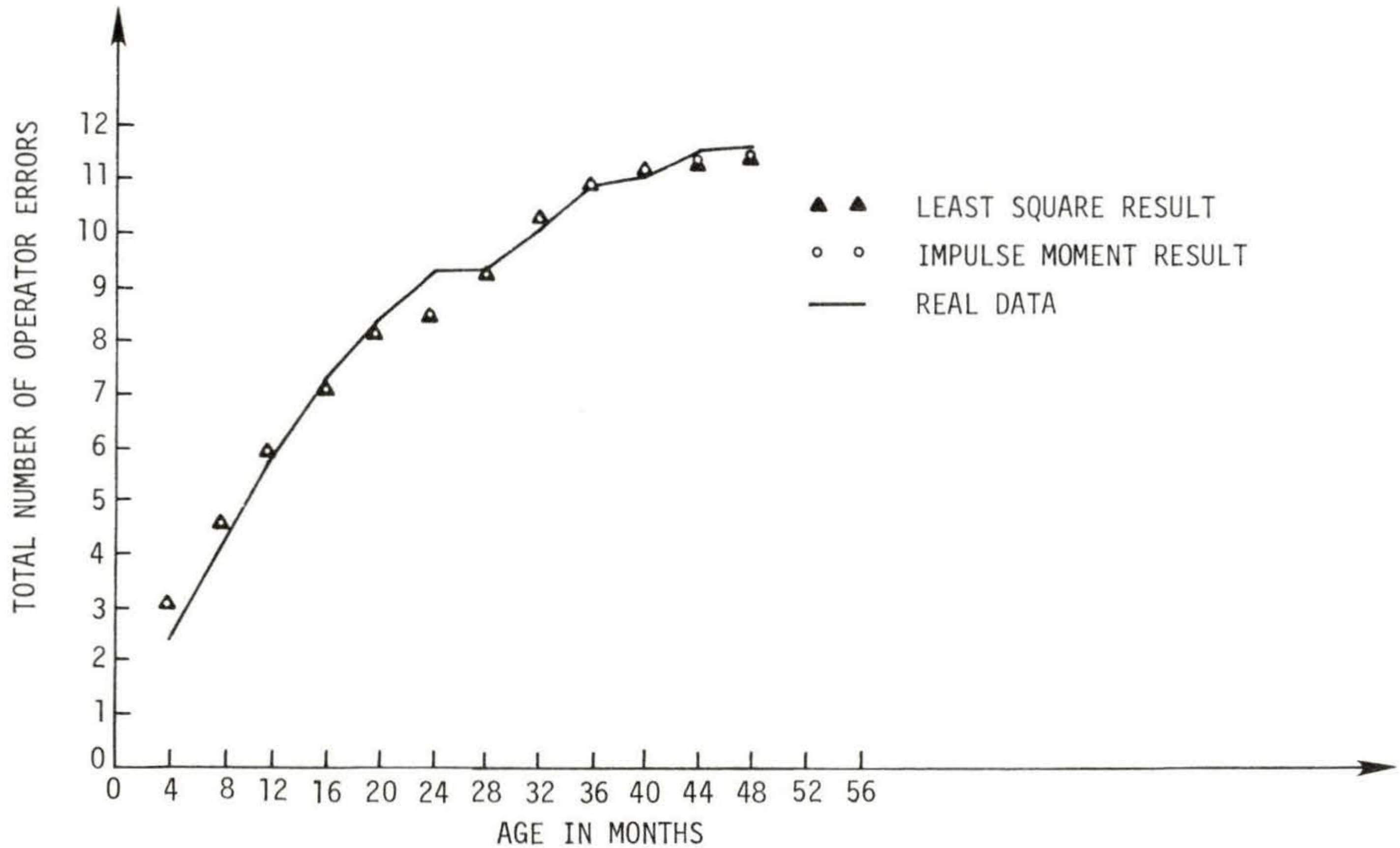

Figure 18. Dynamic estimation for PWR's with power levels from 800 to 1200 MWe

IX. SUMMARY AND CONCLUSIONS

This study has attempted to give insight into the relationship among nuclear reactor type, age and power rating (size) with respect to operator error rates. It has shown, in Chapter VIII, that for Pressurized Water Reactors there is a direct correlation between operator error rate and facility size; the larger the PWR, the greater the number of errors committed. While for Boiling Water Reactors, reactor size does not seem to have any direct affect upon the operator error rate, though the overall error rate for BWR's was larger and considerably more scattered with respect to facility age than similar effects for PWR's. The effect of size on the operator error rate for PWR's might be explained because PWR system complexity increases with facility size much more so than system complexity for BWR's. But since comparable system complexities and the number of design changes are greater for the BWR, the overall error rate is larger for BWR's.

Accordingly, it can be concluded that a constant operator error rate model is not an appropriate assumption for real life operator data, and should not be used for probabilistic analysis. The time varying operator error rate developed here by the Kalman filtering dynamic estimation is more appropriate for this purpose. However, if a time invariant model is to be used it has been shown that a time invariant model with delay will describe the operator learning process better than the classical nondelay time invariant model.

X. RECOMMENDATIONS FOR FURTHER WORK

Further work recommended to enhance the study completed are as follows:

1. To collapse the data in an accurate manner, monthly availability is more suitable than yearly availability. The data should be collapsed according to the power, design, and type of reactors with respect to availability of operation, and age.

2. To smooth the data by the window smoothing method explained in Chapter VI, Section A, a computer code should be developed.

3. To estimate the static and dynamic estimation the (OPEMX-K) computer code which have been developed here can be used.

4. Once the learning parameters for different designs are obtained then a comparison between different design learning parameters can be made for recognization of an optimum design.

XI. REFERENCES

1. Ba Hli, F., A General Method for Time Domain Network Synthesis, IRE Trans., Circuit Theory, 3 (September) 21-28 (1973).

2. Bevis, F. W., C. Finniean and D. R. Towill, "Prediction of Operator Performance During Learning on Repetitive Tasks", Int. J. Prod. Res., 8, 293 (1969).

3. Crossman, E. R. F. W., "A Theory of the Acquisition of Speed-Skill", Ergonomics, 12, 153 (1959).

4. Gelb, Arthur, editor, Applied Optimal Estimation, (MIT Press, Cambridge, Mass., 1974).

5. Hitchings, B., and D. R. Towill, "An Error Analysis of the Time Constant Learning Curve Model", Int. J. Prod. Res., 13 (No. 2) 105-134 (1975).

6. Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems", ASME Trans. Basic Eng., 82 (March), 35-45 (1960).

7. Sriyananda, H., and D. R. Towill, "Prediction of Human Operator Performance", IEEE Trans. Reliability, R-22, 148 (1973).

8. Towill, D. R., "A Model for Describing Process Operator Performance", in The Human Operator in Process Control, E. Edwards and F. P. Lees (editors) (Taylor & Frances, London, 1974), 178.

9. Towill, D. R., "A Direct Method for the Determination of Learning Curve Parameters from Historical Data", Int. J. Prod. Res., 11, 97 (1973).

XII. ACKNOWLEDGMENTS

The author wishes to express his appreciation for assistance provided by his major professors, Dr. Zeinab Sabri and Dr. Abdo Husseiney, and committee members, Dr. Herb David, Statistics, Dr. Keith Adams, Industrial Engineering, and Dr. Richard Danofsky, Nuclear Engineering, for their helpful aid. He also wishes to acknowledge the Nuclear Regulatory Commission and the Engineering Research Institute of Iowa State University for financial assistance. In addition he wishes to thank the members of the Safety Research Group who so carefully worked to provide the author with raw data for the study.

XIII. APPENDIX A

A. Stability of Linear Time Invariant System

Definition 1:

We say that the system $\underline{x}'(t) = \underset{\sim}{A}\underline{x}(t)$ is stable if (Euclidean) norm $||x(t)||$ remains bounded as $t \to \infty$ for every solution $\underline{x}(t)$ of the system. We call the system strictly stable if it is stable and if;

$$\lim_{t\to\infty} ||x(t)|| = 0 \quad \text{for any solution of } \underline{x}(t) \text{ of the system}$$

If there is a solution $\tilde{x}(t)$ such that,

$$\lim_{t\to\infty} ||\tilde{x}(t)|| = \infty$$

Then the system is unstable.

Definition 2:

The system $\dot{\underline{x}}(t) = \underset{\sim}{A}\underline{x}(t)$ is stable if and only if real points of all the eigenvalues of matrix $\underset{\sim}{A}$ are negative or zero. The system is strictly stable if and only if real points of all the eigenvalues of matrix $\underset{\sim}{A}$ are negative.

B. Linear Dynamic System

Definition:

A dynamic system is linear if it is of the form;

$$\dot{\underline{x}}(t) = \underset{\sim}{A}(t)\underline{x}(t) + \underset{\sim}{B}(t)\underline{U}(t) \tag{A1}$$

$$\underline{y}(t) = \underset{\sim}{C}(t)\underline{x}(t) + \underset{\sim}{D}(t)\underline{U}(t) \tag{A2}$$

where

$\underset{\sim}{A}(t)$ is nxn matrix of valued function or constant

$\underset{\sim}{B}(t)$ is mxn matrix of valued function or constant

$\underset{\sim}{C}(t)$ is pxn matrix of valued function or constant

$\underset{\sim}{D}(t)$ is pxm matrix of valued function or constant

x(t), u(t) and y(t) are state variable, input and output vector.

If the matrices $\underset{\sim}{A}$, $\underset{\sim}{B}$, $\underset{\sim}{C}$ and $\underset{\sim}{D}$ are constant, then the system is called time invariant.

C. The Control Problem

Suppose initially $\underline{x} = \underline{x}_0$ at $t = t_0$ and we wish to convert to $\underline{x}_1$, at $t = t_1$ ($\underline{x}_1$ is called the "target") with a suitable choice of $\underline{u}$. Control may be arbitrary or may restraint to be in a set of values (e.g., $|u_i| < M_i$ for some M_i or $u_1^2 + u_2^2 + \ldots + u_m^2 \leq M$). Associated with the control is a functional called cost function;

$$J(\underline{x}_0, t_0, \underline{x}, t, \underline{u}) = K(\underline{x}_1, t_1) + \int_{t_0}^{t} L(\underline{x}, \underline{u}, \tau)\, d\tau \qquad \text{(A3)}$$

where K is a function of the final state and L is a function of $\underline{x}$, $\underline{u}$ evaluated at $t = \tau$.

The problem is called "optimal control" if we wish to minimize or maximize J.

D. Reachable State

Definition:

A state $\underline{x}_1$ is reachable if there is an admissible control $\underline{u}$ such that $\underline{x}_0$, t_0 can be converted to $\underline{x}_1$ by applying the control at some finite time $t_1 \geq t_0$.

E. Controllability

Definition:

A system is said to be controllable if it is possible to find a control vector $\underline{v}(t)$ which is specified finite time t_f will transfer the system between two arbitrarily specified finite state $\underline{x}_0$ and $\underline{x}_f$.

F. Observability

Definition:

A system is said to be observable if measurements of the output $\underline{y}$ contains sufficient information to enable us to completely identify the state $\underline{x}$.

G. Won Ham Theorem

Definition:

To move the poles of a system to any arbitary points with a state variable feedback

$$\mu = -Kx \tag{A4}$$

where K is constant, it is necessary and sufficient for the system to be completely controllable.

H. Autocorrelation Function

Definition:

The autocorrelation function describes the general dependence of the values of the data at one time on the values at another time. Autocorrelation function can be defined as;

$$R_{\underset{\sim}{x}}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_0^T \underline{x}(t)\underline{x}(t+\tau)\,dt \qquad \text{(A5)}$$

The value of autocorrelation function for $\tau=0$ is equal to the second moment of probability function of $\underline{x}$, so in the case that the mean value of $\underline{x}$ is equal zero (white noise), then

$$\sigma_{\underset{\sim}{x}}^2 = R_{\underset{\sim}{x}}(0)$$

XIV. APPENDIX B

A. Comparison Between Maximum Likelihood and Least Square Estimation

Considering that the measurement, $\underline{z}$ and the real value of the parameter, $\underline{x}$ are linearly related

$$\underline{z} = H\underline{x} + \underline{v} \tag{B1}$$

where $\underline{v}$ is an lxl noise vector. If l>n, then the measurement set contains redundant information.

In the least square sense of estimation, one chooses as $\underline{\hat{x}}$ that value which minimizes the sum of squares of the deviations, $\underline{z}_i - \underline{\hat{z}}_i$, that is;

$$\dot{J} = (\underline{z}-H\underline{\hat{x}})^T(\underline{z}-H\underline{\hat{x}}). \tag{B2}$$

The resulting least-squares estimate is;

$$\underline{\hat{x}} = (H^TH)^{-1}H^T\underline{z}. \tag{B3}$$

If we are interested in minimizing the weighted sum of squares of deviations, then

$$\dot{J} = (\underline{z}-H\underline{\hat{x}})^TR^{-1}(\underline{z}-H\underline{\hat{x}}) \tag{B4}$$

where R^{-1} is an lxl symmetric, positive definite weighting matrix. The weighted least squares estimate is

$$\hat{x} = (H^TR^{-1}H)^{-1}H^TR^{-1}\underline{z}. \tag{B5}$$

This result doesn't make any sense, if we don't know the logic behind the weighting matrix. From the probabilistic point of view, one may use the maximum "likelihood" philosophy, which is to take as $\hat{\underline{x}}$ that value which maximizes the probability of the measurements $\underline{z}$ that actually occurred, taking into account known statistical properties of $\underline{v}$. Assuming $\underline{v}$ is taken as a zero mean, Gaussian distributed observation with covariance matrix R, we have

$$P(\underline{z}/\underline{x}) = \frac{1}{(2\pi)^{1/2} \; |R|^{1/2}} \exp[-\tfrac{1}{2}(z-Hx)^T R^{-1}(z-Hx)]. \qquad \text{(B6)}$$

To maximize $P(\underline{z}/\underline{x})$ we should minimize the term between the brackets.

Another approach is Bayesian estimation, where statistical models are available for both $\underline{x}$ and $\underline{z}$, and one seeks the a posteriori condition density function, $P(\underline{x}/\underline{z})$, since it contains all the statistical information of interest. In general

$$P(\underline{x}/\underline{z}) = \frac{P(\underline{z}/\underline{x})P(\underline{x})}{P(\underline{z})} \qquad \text{(B7)}$$

where $P(\underline{x})$ is the a priori probability density function of $\underline{x}$, and $P(\underline{z})$ is the probability density function of the measurements. According to the criterion of optimality one can compute $\hat{\underline{x}}$ from $P(\underline{x}/\underline{z})$, for example, if the object is to maximize the probability that $\hat{\underline{x}} = \underline{x}$, the solution is

$$\hat{x} = \text{mode of } P(\underline{x}/\underline{z}). \tag{B8}$$

When the a priori density function $P(\underline{x})$ is uniform (which implies no knowledge of $\underline{x}$ between its allowable limits), this estimate is equal to the maximum likelihood estimate. If the object is to find a generalized minimum variance Bayes' estimate, that is, to minimize the cost function,

$$J = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (\underline{\hat{x}}-\underline{x})^T S(\underline{\hat{x}}-\underline{x}) P(\underline{x}/\underline{z})\, dy_1 dx_2 \ldots dx_n \tag{B9}$$

where S is an arbitrary, positive semidefinite matrix, we simply set

$$\frac{\partial J}{\partial \underline{\hat{x}}} = \underline{0} \tag{B10}$$

to find, independent of S, that

$$\underline{\hat{x}} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \underline{x} P(\underline{x}/\underline{z})\, dx_1 dx_2 \ldots dx_n = E(\underline{x}/\underline{z}) \tag{B11}$$

which is a conditional mean estimate. Thus

$$J = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} L(\underline{\hat{x}}) P(x/z)\, dx_1 dx_2 \ldots dx_n \tag{B12}$$

where $L(\hat{x})$ is a scalar "loss function" of the estimation error

$$\tilde{x} = \underline{\hat{x}}-\underline{x} \tag{B13}$$

Then we can get the estimate for $\underline{\hat{x}}$ as follows:

$$\underline{\hat{x}} = (P_0^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} \underline{z} \tag{B14}$$

which is similar to least square mean method with a difference in one term P_0, which is the a priori covariance matrix.

XV. APPENDIX C

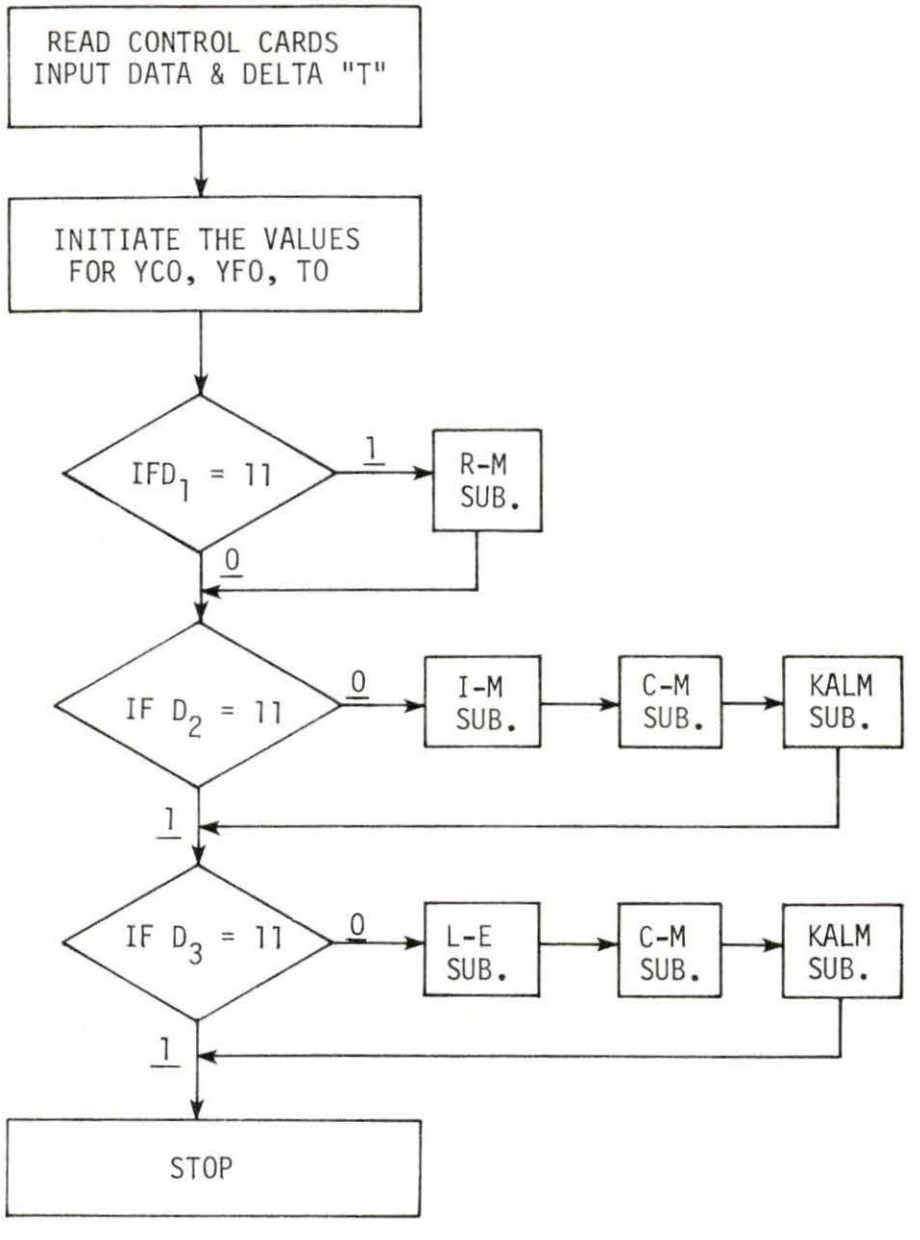

Figure C1. The flow chart of main program

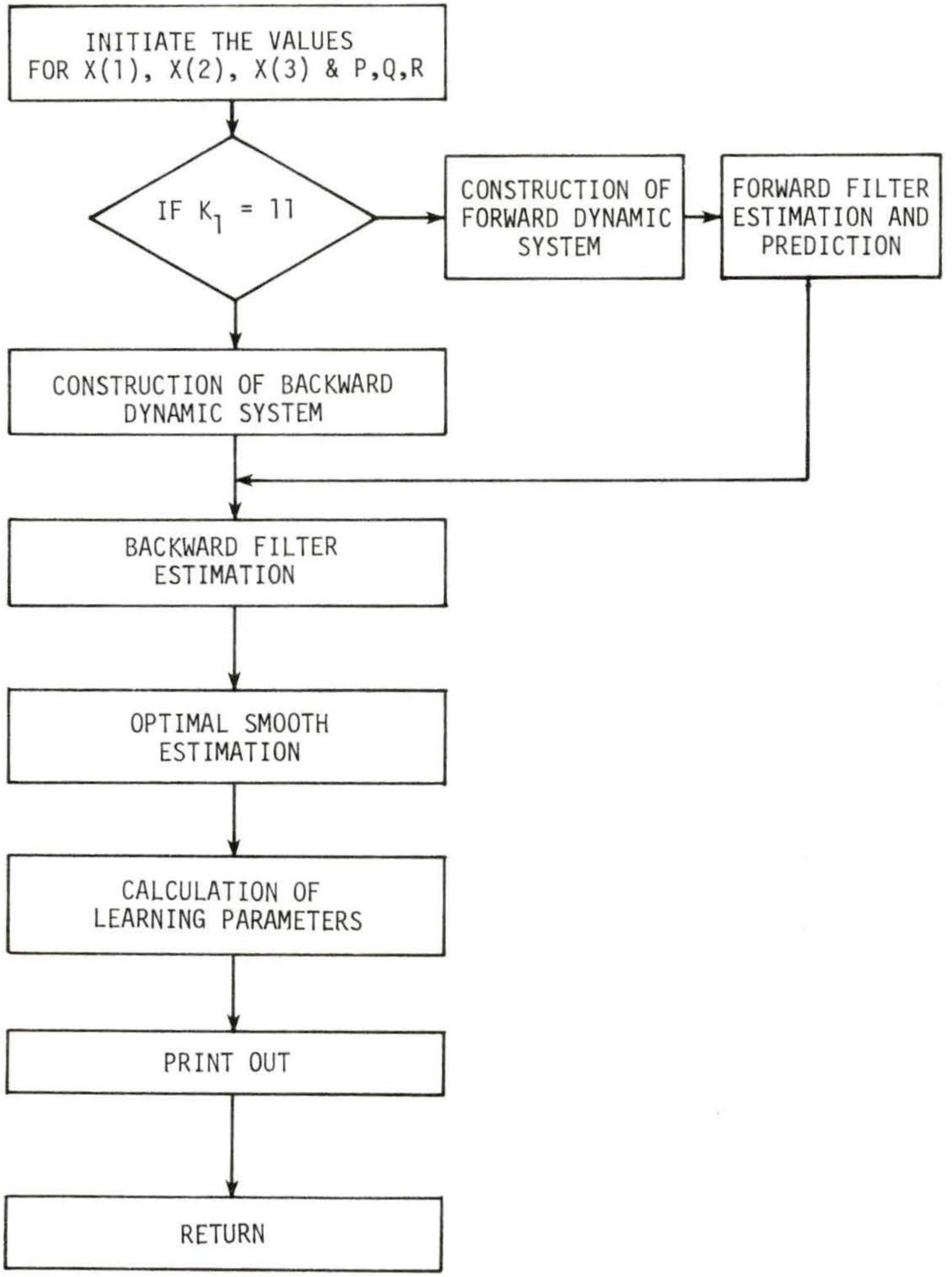

Figure C2. Flow chart for Kalman subroutine

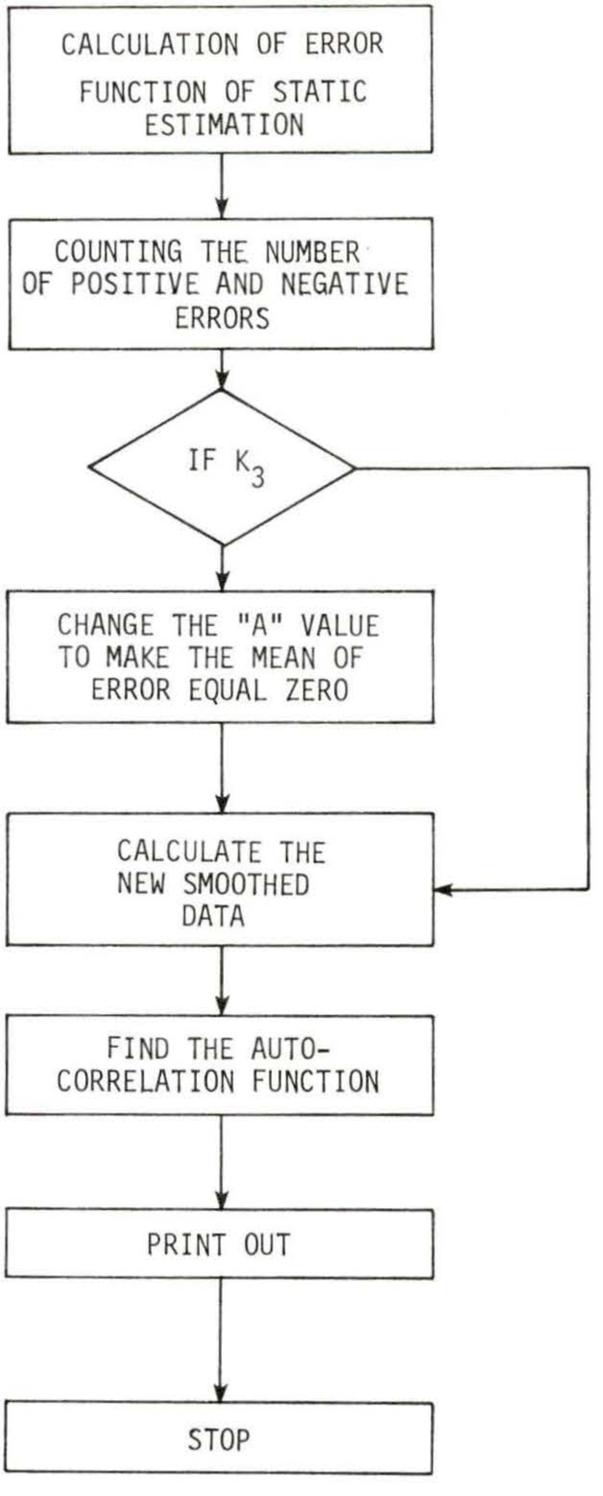

Figure C3. Flow chart of C-M subroutine

```
$JOB              'ALI',TIME=30,PAGES=30
          DIMENSION D(50),DELT(50),TTX(50),XO(3),XN(3),QO(3,3),FO(3,3),
         1PO(3,3),PN(3,3),G(3),PP(3,3),H(3,3),C(3,3),Z(50,6),FM(3,3),
         2ZY(150,6),XS(3,50),DIN(50),DMTBF(50),TBF(50),F(50),B(3,3),TT(50),
         3DD(50),DDELT(50)
          INTEGER D1,D2,D3
C         .........PARAMETER OF LEARNING CURVE........
C         .........NUMBER OF FAILURE CALCULATED IN 4MOTH....
C         ....PWR..DATA..COLLAPSE...POWER..400..TO..600..MWE......
C         .............PRIORI..ESTIMATE..OF..A..IS.0.0.ERROR/4MONTH...
          READ(5,1)N,D1,D2,D3,T,E
    1     FORMAT(4(I2,1X),F10.2,F10.5)
          READ(5,2)(D(I),DELT(I),I=1,N)
    2     FORMAT(3(F10.5,F10.5))
          READ(5,202)A
  202     FORMAT(2X,F10.5)
          TTX(1)=DELT(1)
          DO100I=2,N
          TTX(I)=TTX(I-1)+DELT(I)
  100     CONTINUE
          YFO=D(N)-D(1)
          YCO=D(1)
          A=A/4.
          NN=N*3
          WRITE(6,272)(I,D(I),DELT(I),I=1,N)
  272     FORMAT(5X,I2,'DATA=',E15.3,'TIME INTERVAL=',E15.3)
          IF(D1.EQ.0)GOTO3
          CALL RM(D,DELT,N,TTX,DIN,DMTBF,TBF,F)
    3     IF(D2.EQ.0)GOTO4
          GOTO5
    4     CALL IM(D,DELT,TTX,YCO,YFO,TO,N)
          CALL CM(D,F,YCO,YFO,TO,N,SIG,TTX,DIN,A)
          D3=D3+1
          CALL KALM(D,DELT,PN,XO,XN,QO,FO,PO,FM,G,PP,H,B,C,T,TTX,SIG,D2,D3,
         1N,D1,Z,ZY,NN,YCO,YFO,TO,DD,DDELT,A)
          D3=D3-1
    5     IF(D3.EQ.0)GOTO6
          GOTO7
    6     CALL LE(D,TTX,YCO,YFO,TO,E,N)
          D2=1
          CALL CM(D,F,YCO,YFO,TO,N,SIG,TTX,DIN,A)
          CALL KALM(D,DELT,PN,XO,XN,QO,FO,PO,FM,G,PP,H,B,C,T,TTX,SIG,D2,D3,
         1N,D1,Z,ZY,NN,YCO,YFO,TO,DD,DDELT,A)
    7     STOP
          END

          SUBROUTINE RM(D,DELT,N,TT,DIN,DMTBF,TBF,F)
          DIMENSION D(N),DELT(N),DIN(N),DMTBF(N),TBF(N),F(N),TT(N)
          DIN(1)=D(1)*DELT(1)
          DO8I=2,N
          DIN(I)=((((D(I)-D(I-1))/2)+D(I-1))*DELT(I))+DIN(I-1)
    8     CONTINUE
          DO9I=1,N
          F(I)=D(I)*(EXP(-DIN(I)))
    9     CONTINUE
          SUM=0
          DO10J=1,N
          SUM=SUM+F(J)*DELT(J)
   10     CONTINUE
          DO11K=1,N
```

```
      F(K)=F(K)/SUM
   11 CONTINUE
      DMTBF(1)=F(1)*(DELT(1)**2)
      TBF(1)=DMTBF(1)
      DO12K=2,N
      DMTBF(K)=((((F(K)*TT(K))-(F(K-1)*TT(K-1)))/2)+F(K-1)*TT(K-1))
      DMTBF(K)=DMTBF(K)*DELT(K)+DMTBF(K-1)
      TBF(K)=(TT(K)*DMTBF(K)-TT(K-1)*DMTBF(K-1))/DELT(K)
   12 CONTINUE
      WRITE(6,13)(K,TBF(K),DELT(K),K=1,N)
   13 FORMAT(5X,'TME BETWEEN FAILURE(',I2,')=',E12.5,2X,'D=',E12.5)
      DO200J=1,N
      DELT(K)=TBF(K)
  200 CONTINUE
      RETURN
      END

      SUBROUTINE LE(D,TT,YCO,YFO,TO,E,N)
      DIMENSION D(N),TT(N)
      TO=1/TO
      K=0
   14 K=K+1
      IF(K.GT.10)GOTO20
      S1=0.
      S2=0.
      SP3=0.
      R2=0.
      RP3=0.
      Q1=0.
      Q2=0.
      QP3=0.
      PP1=0.
      PP3=0.
      DO15I=1,N
      PEXP=EXP(-TT(I)*TO)
      FPEXP=1.-PEXP
      DELY=D(I)-(YCO+YFO*FPEXP)
      S1=DELY+S1
      S2=DELY*FPEXP+S2
      SP3=DELY*TT(I)*PEXP+SP3
      Q1=FPEXP+Q1
      Q2=FPEXP**2+Q2
      QP3=FPEXP*TT(I)*PEXP+QP3
      PP1=TT(I)*PEXP+PP1
      PP3=(TT(I)*PEXP)**2+PP3
      R2=FPEXP+R2
      RP3=TT(I)*PEXP+RP3
   15 CONTINUE
      S3=YFO*SP3
      R3=YFO*RP3
      Q3=YFO*QP3
      P1=YFO*PP1
      IF(YFO.LE.0.)GOTO16
      P3=(YFO**2.)*(PP3)
      GOTO17
   16 P3=((-YFO)**2)*PP3
   17 P2=Q3
      R1=N
      DELYC=Q2*P3*S1+Q3*P1*S2+Q1*P2*S3-P2*Q3*S1-Q1*P3*S2-P1*Q2*S3
      DELYF=R3*P2*S1+R1*P3*S2+R2*P1*S3-R2*P3*S1-R3*P1*S2-R1*P2*S3
```

```
      DELTO=R2*Q3*S1+R3*Q1*S2+R1*Q2*S3-R3*Q2*S1-R1*Q3*S2-R2*Q1*S3
      DEL=R1*Q2*P3+Q1*P2*R3+P1*R2*Q3-(R1*P2*Q3+Q1*R2*P3+P1*Q2*R3)
      DELYF=DELYF/DEL
      DELYC=DELYC/DEL
      DELTO=DELTO/DEL
      E1=ABS(DELYF)
      E2=ABS(DELYC)
      E3=ABS(DELTO)/TO
      WRITE(6,971)E2,E1,E3
  971 FORMAT(5X,'LE.ERROR.IN YCC YFO TO',3E8.2)
      IF(E1.LE.E)GOTO18
      YFC=YFC+DELYF
      GOTO19
   18 IF(E2.LE.E)GOTO21
      YCC=YCO+DELYC
      GOTO22
   21 IF(E3.LE.E)GOTO20
      TO=TO+DELTO
   19 IF(E2.LE.E)GOTO22
      YCO=YCO+DELYC
   22 IF(E3.LE.E)GOTO14
      YFO=YFO+DELYF
      GOTO14
   20 TO=1/TO
      WRITE(6,23)YCO,YFO,TO,K
   23 FORMAT(2X,'YCO=',F10.5,'YFC=',F10.5,'TO=',F10.5,'K=',I3,'LE OUT')
      RETURN
      END

      SUBROUTINE CM(D,F,YCO,YFO,TO,N,SIG,TT,DIN,A)
      DIMENSION D(N),F(N),TT(N),DIN(N)
      K3=0
   27 K3=K3+1
      K1=0
      K2=0
      SUM=0.
      DO24J=1,N
      F(J)=(D(J)-(YCO+YFO*(1-EXP(-TT(J)/TO))))/ABS(YFC)
      IF(F(J).GT.0.)GOTO25
      K2=K2+1
      GOTO500
   25 K1=K1+1
  500 SUM=SUM+F(J)
   24 CONTINUE
      SUM=SUM/N
      WRITE(6,26)K1,K2,SUM
   26 FORMAT(5X,'NUMBEROF',5X,'PSITIVE=',I2,5X,'NEG=',I2,'MEW=',F10.5)
      SS=0.
      DO83I=1,N
      SS=SS+TT(I)
   83 CONTINUE
      EEE=SUM*N*YFO/SS
      DO85I=1,N
      D(I)=D(I)-EEE*TT(I)
   85 CONTINUE
      A=A+EEE
      WRITE(6,84)EEE,A
   84 FORMAT(/,'DEVIATION FROM A= ',E10.3,/,10X,'A=',E10.3)
      IF(K3.EQ.2)GOTO28
      GOTO27
```

```
   28 KK=0
   29 KK=KK+1
      FF=0.
      N1=N-KK+1
      I=KK-1
      DO30J=1,N1
      FF=F(J)*F(J+I)+FF
   30 CONTINUE
      DIN(KK)=FF/N1
      IF(KK.EQ.N)GOTO159
      GOTO29
  159 WRITE(6,31)(DIN(I),I=1,N)
   31 FORMAT(5X,'AUTOCORRELLATICN=',E12.3)
      WRITE(6,271)YCO,YFO,TO
  271 FORMAT(5X,'NEW VALUES OF YCC YFO TO',3F10.3)
      IF(DIN(1).EQ.C.)GOTO166
      GOTO167
  166 DIN(1)=0.1
  167 SIG=DIN(1)*(YFO**2)
      WRITE(6,168)SIG
  168 FORMAT(30X,'SIG=',F10.7)
      RETURN
      END

      SUBROUTINE KALM(D,DELT,PN,XO,XN,QO,FO,PO,FM,G,PP,H,B,C,T,TT,SIG,
     1D2,D3,N,D1,Z,ZY,NN,YCO,YFO,TO,DD,DDELT,A)
      DIMENSICN D(N),DELT(N),PN(3,3),XO(3),XN(3),QO(3,3),FO(3,3),Z(N,6),
     1PO(3,3),FM(3,3),G(3),PP(3,3),H(3,3),B(3,3),C(3,3),TT(N),ZY(NN,6),
     2DD(N),DDELT(N)
      INTEGER D1,D2,D3
      IF(D1.EQ.1)GOTO38
      WRITE(6,39)D1
   39 FORMAT(5X,'KALMAN FILTER OUTPUT',5X,'PERFCRMANCE',2X,'D1=',I2)
      GCTO41
   38 WRITE(6,40)D1
   40 FORMAT(5X,'KALMAN FILTER OUTPUT',5X,'LEARNING',2X,'D1=',I2)
   41 IF(D2.EQ.1)GOTO42
      WRITE(6,43)D2
   43 FORMAT(5X,'DATA FROM LEASTSQUARE',5X,'D2=',I2)
   42 IF(D3.EQ.1)GOTO45
      WRITE(6,44)D3
   44 FORMAT(5X,'DATA FROM IMPULSEMOMENT',5X,'D3=',I2)
   45 XO(1)=YCO
      XO(2)=-(1/TO)
      XO(3)=(YCC+YFO)/TO
      DC46I=1,3
      DO47J=1,3
      PC(I,J)=0.
      QO(I,J)=0.
      FO(I,J)=0.
   47 CONTINUE
   46 CONTINUE
      PO(1,1)=SIG
      PO(2,2)=-(XO(2)*SIG)/YFO
      PO(3,3)=SIG/(TO**2)
      QO(1,1)=(2.*PO(1,1))/(3.*(TO**2))
      QO(2,2)=PC(2,2)/(TO*T)
      QO(3,3)=QC(1,1)
      WRITE(6,48)((I,J,PO(I,J),QO(I,J),J=1,3),I=1,3)
   48 FORMAT(2X,I2,2X,I2,'PO=',E12.3,'QO=',E12.3)
```

```
      RO=SIG*(XO(1)**2)
      FO(2,1)=0.
      FO(3,1)=0.
      FO(3,2)=0.
      FO(2,3)=0.
      FO(2,2)=1.
      FO(3,3)=1.
      DO900I=1,N
      DD(I)=D(I)
      DDELT(I)=DELT(I)
  900 CONTINUE
      NN1=N-1
      K1=0
   49 K1=K1+1
      IF(K1.EQ.2)GOTO901
      GOTO902
  901 DO903J=1,NN1
      DD(J)=D(N-J+1)
      DDELT(J+1)=DELT(N-J+1)
  903 CONTINUE
      DD(N)=D(1)
      DDELT(1)=T-TT(N)
  902 DO50K=1,N
      IF(K1.EQ.2)GOTO910
      FO(1,1)=1.+XO(2)*DELT(K)
      FO(1,2)=XO(1)*DELT(K)
      FO(1,3)=DELT(K)
      UO=-XO(2)*XO(1)*DELT(K)
      GOTO911
  910 FO(1,1)=1.-XO(2)*DDELT(K)
      FO(1,2)=XO(1)*DDELT(K)
      FO(1,3)=DDELT(K)
      UO=XO(2)*XO(1)*DDELT(K)
  911 XN(1)=FO(1,1)*XO(1)+FO(1,2)*XO(2)+FO(1,3)*XO(3)+UO
      XN(2)=XO(2)
      XN(3)=XO(3)
      CALL MULT(FO,PO,FM,0.0)
      CALL MULT(FM,FO,PO,1.)
      DO51I=1,3
      DO52J=1,3
      PN(I,J)=PO(I,J)+QO(I,J)
   52 CONTINUE
   51 CONTINUE
      AA=PN(1,1)+RO
      DO59L=1,3
      G(L)=PN(L,1)/AA
      XO(L)=XN(L)+G(L)*(DD(K)-XN(1))
      L1=L
      IF(K1.EQ.2)L1=L+3
      Z(K,L1)=XO(L)
   59 CONTINUE
      RO=SIG*((XO(1)/YFO)**2)
      PP(1,2)=0
      PP(1,1)=1.-G(1)
      PP(1,3)=0.
      PP(2,1)=-G(2)
      PP(3,1)=-G(3)
      PP(2,2)=1.
      PP(2,3)=0.
      PP(3,3)=1.
```

```
      PP(3,2)=0.
      CALL MULT(PP,PN,PO,0.)
      K2=3*(K1-1)+1
      K3=3*(K-1)+1
      ZY(K3,K2)=PO(1,1)
      ZY(K3,K2+1)=PO(1,2)
      ZY(K3,K2+2)=PO(1,3)
      ZY(K3+1,K2)=PO(2,1)
      ZY(K3+1,K2+1)=PO(2,2)
      ZY(K3+1,K2+2)=PO(2,3)
      ZY(K3+2,K2)=PO(3,1)
      ZY(K3+2,K2+1)=PO(3,2)
      ZY(K3+2,K2+2)=PO(3,3)
   50 CONTINUE
      IF(K1.EQ.1)GOTO904
      GOTO905
  904 XN(1)=FO(1,1)*XO(1)+FO(1,2)*XO(2)+FO(1,3)*XO(3)+UO
      XO(1)=XN(1)
      XN(2)=XO(2)
      XN(3)=XO(3)
      WRITE(6,906)(XN(I),I=1,3),T
  906 FORMAT(5X,'PERDICTICN',3(2X,E12.3),F10.5)
  972 FORMAT(5X,'ERROR IN COV',3E10.2)
      WRITE(6,972)((PN(I,J),J=1,3),I=1,3)
  905 IF(K1.EQ.2)GOTO53
      GOTO49
   53 WRITE(6,70)K1
   70 FORMAT(5X,'FORWARD FILTER ESTIMATICN',5X,'BACKWARD FILTER',2X,I2)
      WRITE(6,71)((Z(I,J),J=1,6),I=1,N)
   71 FORMAT(5X,3E12.3,5X,3E12.3)
      DO72J=1,N
      N7=3*(J-1)+1
      N8=NN+1-(3*J)
      PO(1,1)=ZY(N7+1,2)*ZY(N7+2,3)-ZY(N7+1,3)*ZY(N7+2,2)
      PO(1,2)=ZY(N7+2,2)*ZY(N7,3)-ZY(N7,2)*ZY(N7+2,3)
      PO(1,3)=ZY(N7,2)*ZY(N7+1,3)-ZY(N7,3)*ZY(N7+1,2)
      PO(2,1)=ZY(N7+2,1)*ZY(N7+1,3)-ZY(N7+1,1)*ZY(N7+2,3)
      PO(2,2)=ZY(N7,1)*ZY(N7+2,3)-ZY(N7+2,1)*ZY(N7,3)
      PO(2,3)=ZY(N7+1,1)*ZY(N7,3)-ZY(N7,1)*ZY(N7+1,3)
      PO(3,1)=ZY(N7+1,1)*ZY(N7+2,2)-ZY(N7+1,2)*ZY(N7+2,1)
      PO(3,2)=ZY(N7+2,1)*ZY(N7,2)-ZY(N7,1)*ZY(N7+2,2)
      PO(3,3)=ZY(N7,1)*ZY(N7+1,2)-ZY(N7,2)*ZY(N7+1,1)
      DET=ZY(N7,1)*PO(1,1)+ZY(N7,2)*PO(2,1)+ZY(N7,3)*PO(3,1)
      PN(1,1)=ZY(N8+1,5)*ZY(N8+2,6)-ZY(N8+1,6)*ZY(N8+2,5)
      PN(1,2)=ZY(N8+2,5)*ZY(N8,6)-ZY(N8,5)*ZY(N8+2,6)
      PN(1,3)=ZY(N8,5)*ZY(N8+1,6)-ZY(N8,6)*ZY(N8+1,5)
      PN(2,1)=ZY(N8+2,4)*ZY(N8+1,6)-ZY(N8+1,4)*ZY(N8+2,6)
      PN(2,2)=ZY(N8,4)*ZY(N8+2,6)-ZY(N8+2,4)*ZY(N8,6)
      PN(2,3)=ZY(N8+1,4)*ZY(N8,6)-ZY(N8,4)*ZY(N8+1,6)
      PN(3,1)=ZY(N8+1,4)*ZY(N8+2,5)-ZY(N8+1,5)*ZY(N8+2,4)
      PN(3,2)=ZY(N8+2,4)*ZY(N8,5)-ZY(N8,4)*ZY(N8+2,5)
      PN(3,3)=ZY(N8,4)*ZY(N8+1,5)-ZY(N8,5)*ZY(N8+1,4)
      BET=ZY(N8,4)*PN(1,1)+ZY(N8,5)*PN(2,1)+ZY(N8,6)*PN(3,1)
      WRITE(6,201)DET,BET
  201 FORMAT(5X,'DETOFPF=',E10.4,5X,'DETOFPB=',E10.4)
      IF(DET.EQ.0.)GOTO72
      IF(BET.EQ.0.)GOTO1000
      DO73M=1,3
      DO74L=1,3
      PO(M,L)=PO(M,L)/DET
```

```
      PN(M,L)=PN(M,L)/BET
      QO(M,L)=PC(M,L)+PN(M,L)
   74 CONTINUE
   73 CONTINUE
      FM(1,1)=QO(2,2)*QO(3,3)-QO(2,3)*QO(3,2)
      FM(1,2)=QO(3,2)*QO(1,3)-QO(1,2)*QO(3,3)
      FM(1,3)=QO(1,2)*QO(2,3)-QO(1,3)*QO(2,2)
      FM(2,1)=QO(3,1)*QO(2,3)-QO(2,1)*QO(3,3)
      FM(2,2)=QO(1,1)*QO(3,3)-QO(3,1)*QO(1,3)
      FM(2,3)=QO(2,1)*QO(1,3)-QO(1,1)*QO(2,3)
      FM(3,1)=QO(2,1)*QO(3,2)-QO(2,2)*QO(3,1)
      FM(3,2)=QO(3,1)*QO(1,2)-QO(1,1)*QO(3,2)
      FM(3,3)=QO(1,1)*QO(2,2)-QO(1,2)*QO(2,1)
      DF=FM(1,1)*QO(1,1)+FM(1,2)*QO(2,1)+FM(1,3)*QO(3,1)
      DO75M=1,3
      DO76L=1,3
      FM(M,L)=FM(M,L)/DF
   76 CONTINUE
   75 CONTINUE
      DO90I=1,3
      LL=N-J+1
      G(I)=PO(I,1)*Z(J,1)+PO(I,2)*Z(J,2)+PO(I,3)*Z(J,3)+PN(I,1)*Z(LL,4)
     1+PN(I,2)*Z(LL,5)+PN(I,3)*Z(LL,6)
   90 CONTINUE
      DO930 I=1,3
      Z(J,I)=FM(I,1)*G(1)+FM(I,2)*G(2)+FM(I,3)*G(3)
  930 CONTINUE
      GOTO72
 1000 Z(J,1)=Z(N+1-J,4)
      Z(J,2)=Z(N+1-J,5)
      Z(J,3)=Z(N+1-J,6)
   72 CONTINUE
      WRITE(6,77)D1
   77 FORMAT(5X,'THE SMOOTH KALMAN RESULT',I2)
      WRITE(6,78)(D(I),(Z(I,J),J=1,3),I=1,N)
   78 FORMAT(5X,'D=',4E15.4)
      DO454I=1,N
      BF=(Z(I,3)+Z(I,1)*Z(I,2))/A
      CF=-Z(I,2)
      WRITE(6,456)A,BF,CF,TT(I)
  456 FORMAT(2X,'LEARNING PARAMETERS',3E10.4,'TIME=',E10.4)
      TNF=A*TT(I)+A*BF*(1.-EXP(-CF*TT(I)))/CF
  455 FORMAT(10X,'TOTAL NUMBER OF ERROR',2X,E10.4)
      WRITE(6,455)TNF
  454 CONTINUE
      RETURN
      END

      SUBROUTINE MULT(H,B,C,Z1)
      DIMENSION H(3,3),B(3,3),C(3,3)
      DO86I=1,3
      DO87L=1,3
      SUM=0.
      DO88J=1,3
      IF(Z1.EQ.1.)GOTO89
      SUM=H(I,J)*B(J,L)+SUM
      GOTO88
   89 SUM=H(I,J)*B(L,J)+SUM
   88 CONTINUE
      C(I,L)=SUM
```

```
   87 CONTINUE
   86 CONTINUE
      RETURN
      END

      SUBROUTINE IM(D,DELT,TT,YCO,YFO,TO,N)
      DIMENSION D(N),DELT(N),TT(N)
      CO=0.
      C1=TT(1)*D(1)/2.
      DO100I=2,N
      CO=CO+(D(I)-D(I-1))
      C1=C1+(D(I)-D(I-1))*((TT(I)+TT(I-1))/2)
  100 CONTINUE
      TO=C1/CO
      YF=CO
      WRITE(6,101)TO,YF
  101 FORMAT(2X,'IM OUTPUT TO=',E12.3,'IF(TO.LE.0)SCATTEE','YF=',E12.3)
      YC=D(1)
      RETURN
      END
```