MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES
3 4456 0148724 2

ORNL/TM-10087

ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

Analysis of Loop Current Step Response Data Obtained from in Situ Tests of Temperature Detectors

L. F. Miller
R. L. Shepard

OAK RIDGE NATIONAL LABORATORY
CENTRAL RESEARCH LIBRARY
CIRCULATION SECTION
4500N ROOM 175
LIBRARY LOAN COPY
DO NOT TRANSFER TO ANOTHER PERSON
If you wish someone else to see this report, send in name with report and the library will arrange a loan.

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A03; Microfiche A01

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

ORNL/TM-10087

Instrumentation and Controls Division

ANALYSIS OF LOOP CURRENT STEP RESPONSE DATA OBTAINED FROM IN SITU TESTS OF TEMPERATURE DETECTORS

L. F. Miller*
R. L. Shepard

*The University of Tennessee

Previously issued as ORNL/CF-77/467
Date Published: May 1986

Research conducted in conjunction with the
Faculty Research Participation Program
of Oak Ridge Associated Universities.

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831
operated by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-84OR21400

MARTIN MARIETTA ENERGY SYSTEMS LIBRARIES
3 4456 0148724 2

ABSTRACT

Methods for obtaining unbiased parameter estimates from data obtained by in situ tests of temperature detectors are evaluated. A computer program that calculates unbiased estimates of parameters that define a dynamic model of the temperature detector and that calculates standard deviations of the model parameters and of the response time is presented.

The computer program, along with the associated theoretical development, represent an extension of the previous capability for analyzing data from in situ tests of temperature detectors.

INTRODUCTION

Methods for analyzing in situ test data of temperature detectors have heretofore been essentially limited to obtaining unbiased estimates of parameters that define a dynamic model of a temperature detector.[1,2] Statistics related to the parameter estimates can be obtained by analyzing many sets of data or by analyzing the residual of a particular set of data in conjunction with the model.

The research reported herein extends previous analysis capability in several ways: (1) a nonlinear minimization algorithm is developed (and implemented with a computer program) that, in theory, always converges to a local minimum[3]; (2) the computer program is tailored to implementation on a microprocessor; and (3) standard deviations of the model parameters and of the response time (calculated from the model parameters) are obtained from a single data set.

The overall research effort includes: (1) evaluation of three minimization algorithms, (2) development of the capability to obtain standard derivations of desired parameters, and (3) computer program verification.

Several nonlinear minimization algorithms commonly used for practical problems are: (1) steepest descent in conjunction with a line search, (2) linearization of the functional or approximating function in conjunction with a variable stepsize (a generalization of Newton's method), and (3) Marquardt's method. The best algorithm for a particular application is usually not known a priori. Thus, three nonlinear minimization algorithms are evaluated for application to the LCSR data analysis: (1) linearization, (2) Marquardt's method, and (3) a combination of Marquardt's method and linearization. The linearization method is preferred to Marquardt's method or the combined method for the LCSR data analysis problem.

GENERAL COMMENTS ON NONLINEAR MINIMIZATION ALGORITHMS

Two basic problems associated with any iterative nonlinear search aogirithm are: (1) determining a direction vector and (2) determining the optimal stepsize in the specified search direction. Most methods provide an implicit estimate of the stepsize in addition to determining the search direction. Nevertheless, it is frequently desirable to expend computational overhead to obtain an optimal (or suboptimal) estimate of the stepsize for at least two reasons: (1) to improve overall computational efficiency and (2) to ensure convergence properties.

Nonlinear Minimization Using Linearization

One advantage of the linearization method over some second order methods is that the coefficient matrix for determining the search direction is positive

definite (at least semidefinite)[4]; consequently, the functional value decreases in the selected direction. Thus, if the optimal (or suboptimal) stepsize is obtained for each iteration, the linearization method is theoretically guaranteed to converge to a local minimum. In practice, guaranteed convergence cannot always be achieved.[3] A disadvantage of the linearization method is that convergence is typically slow near the minimum, compared to a second order method.

The function which approximates the data is given by,

$$f(t) = a_0 + \sum_{i=1}^{M} a_i e^{\lambda_i t} \quad . \tag{1}$$

Typically, two exponential terms (M=2) are sufficient to adequately approximate LCSR data. In order to estimate the model parameters (a_0, ..., a_M, λ_1, ..., λ_M), the variance of the residual is minimized; in particular, the following functional is minimized

$$\Phi(\underline{\theta};\underline{Y}_e,\underline{t}) = \frac{1}{N-K} \sum_{k=1}^{N} (f_k - Y_{ek})^2 \ , \ \theta \epsilon R^K \tag{2}$$

where

$$\underline{\theta}^T = (a_0, a_1, \ldots, a_M, \lambda_1, \ldots, \lambda_M) \ , \ \underline{\theta} \epsilon R^K \tag{3}$$

$$f_k = a_0 + \sum_{i=1}^{M} a_i e^{\lambda_i t_k} \tag{4}$$

Y

$$Y_{ek} = \text{the observed data} \tag{5}$$

$$\underline{t} = (t_1, \ldots, t_n) \ . \tag{6}$$

If the approximating function is expanded in a first order Taylor series, one obtains

$$\tilde{\Phi}(\underline{\Theta};\underline{Y}_e,\underline{t}) = \frac{1}{N-K} \sum_{k=1}^{N} \left[f_{ok} + \left(\frac{\partial f}{\partial \underline{\Theta}}\right)_k \underline{\delta\Theta} - Y_{ek} \right]^2 . \tag{7}$$

Minimization of $\tilde{\Phi}$ is equivalent to the linear least squares problem for the set of equations,

$$\left(\frac{\partial f}{\partial \underline{\Theta}}\right)_k \underline{\delta\Theta} = Y_{ek} - f_{ok} \; ; \; k = 1, \ldots, N . \tag{8}$$

Thus, minimization of the functional given by Eq. (2) is converted into a sequence of linear least squares problems by linear least squares iterations. At each iteration, the optimum stepsize is calculated so that $\Phi(\underline{\Theta} + \rho\, \underline{\delta\Theta}; \underline{Y}_e, t)$ is minimized; in particular, Φ is minimized with respect to ρ for each iteration.

The optimum stepsize is estimated by using three points: (1) the point where the functional minimum occurred for the previous iteration, (2) a point which reduces the functional value (this point is found by halving the internal), and (3) a point which increases the functional value (this point is found by increasing the stepsize). A quadratic curve is defined by these three points; thus, the optimum stepsize is easily estimated.

Since the direction is obtained through the use of a positive definite (at least semi-definite)[4] coefficient matrix, the functional is reduced in the direction chosen.

Three criteria are used for terminating the minimization algorithm: (1) if the norm of the functional gradient is small with respect to the norm of the

functional, (2) if the functional value cannot be reduced in the selected direction by halving the internal a selected number of times, and (3) if the specified number of iterations is exceeded.

Nonlinear Minimization Using Marquardt's Method

Marquardt's method uses the Hession matrix to calculate the search direction and stepsize. Since this is a second order method, it has very good convergence properties near a local minimum. On the other hand, the Hessian matrix for calculating the search direction may be negative definite as well as positive definite; thus, a method based only on the Hessian matrix could result in finding a maximum instead of a minimum. Marquardt has devised a method to circumvent this problem to ensure that the search algorithm always leads to decreasing the functional. Details of Marquardt's method are given by Bard.[4]

Marquardt's method requires the calculation of the second partial derivatives of the functional with respect to the model parameters as a preliminary step. The appropriate derivatives are as follows:

$$\frac{\partial\phi}{\partial\Theta_i}(\underline{\Theta};\underline{Y}_e,\ t_k) = \frac{2}{N-K}\sum_{k=1}^{N} R_k \frac{\partial f_k}{\partial\Theta_i} \tag{9}$$

where

$$R_k = f_k - Y_{ek} \tag{10}$$

and

$$\frac{\partial^2\phi}{\partial\Theta_j\partial\Theta_i} = \frac{2}{N-K}\sum_{k=1}^{N}\left[\left(\frac{\partial f_k}{\partial\Theta_j}\right)\left(\frac{\partial f_k}{\partial\Theta_i}\right) + R_k\frac{\partial f_k}{\partial\Theta_j\partial\Theta_i}\right] . \tag{11}$$

The second partials of the approximating function are specified by:

$$\frac{\partial^2 f_k}{\partial\Theta_j \partial\Theta_i} = t_k \frac{\partial f_k}{\partial\Theta_i} \; ; \; i \leq M+1, \; j > M+1 \text{ and } j = M+i \tag{12}$$

$$\frac{\partial^2 f_k}{\partial\Theta_j \partial\Theta_i} = t_k \frac{\partial f_k}{\partial\Theta_j} \; ; \; i > M+1, \; j \leq M+L \text{ and } i = M+j \tag{13}$$

$$\frac{\partial^2 f_k}{\partial\Theta_j \partial\Theta_j} = t_k \frac{\partial f_k}{\partial\Theta_i} \; ; \; i > M+1, \; j > M+1 \text{ and } i = j \; ; \tag{14}$$

otherwise, the second partials are zero. The first partials are easily computed. For example, if K=5

$$\frac{\partial f_k}{\partial a_0} = 1 \; ; \; \frac{\partial f_k}{\partial a_1} = e^{\lambda_1 t_k} \; , \; \frac{\partial f_k}{\partial a_2} = e^{\lambda_2 t_k} \tag{15}$$

$$\frac{\partial f_k}{\partial \lambda_1} = a_1 t_k \, e^{\lambda_1 t_k} \; , \; \frac{\partial f}{\partial \lambda_2} = a_2 t_k \, e^{\lambda_2 t_k} \; , \; \ldots \tag{16}$$

where

$$(\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5) = (a_0, a_1, a_2, \lambda_1, \lambda_2) \; . \tag{17}$$

Standard Deviations of the Model Parameters and of Response Time

Data obtained from a LCSR test are analyzed to obtain unbiased estimates of several parameters. Some of the estimated parameters are subsequently used to estimate a time constant that characterizes a plunge test of the temperature detector.

Uncertainty in the estimates of the parameters that characterize the LCSR data arises from at least two factors: (1) the data is contaminated with noise and (2) the model is not exact. A method for estimating uncertainty in unbiased parameter estimates has been discussed with Bayne[5] and is described by Bard.[4] If one is given the standard deviation of the parameter estimates, two methods could be used to estimate the standard deviation of the plunge test time constant obtained from the parameter estimates: (1) the propagation of error formula[6] could be employed in conjunction with an analytical approximation for the time constant,[7] or (2) the parameters used to obtain the "plunge test time constant" could be randomly varied in conjunction with a direct calculation of the time constant. The first method described above is the one used for the work reported herein.

The model that approximates the LCSR data is assumed to be

$$f(\hat{\theta};t) = a_0 + a_1 e^{\lambda_1 t} + a_2 e^{\lambda_2 t} \tag{18}$$

where

$$\theta^T = (a_0, a_1, a_2, \lambda_1, \lambda_2), \ \theta \epsilon R^5 \tag{19}$$

and $\hat{\theta}$ denotes the unbiased parameter estimates. In terms of the model and the residuals (noise), the observed data are given by

$$Y_j = f(\hat{\theta},t_j) + \epsilon_j \ ; \ j = 1, \ldots, N \ . \tag{20}$$

where

Y_j = an observed datum

ϵ_j = the residual.

It is assumed that the expected value of the residuals is near zero; in particular,

$$E(\varepsilon_j) \cong 0 \quad . \tag{21}$$

Also, the best estimate of the variance of the residuals is used to estimate the true variance of the residuals,

$$E\left(\varepsilon_j^{\,2}\right) = S^2 \tag{22}$$

and

$$S^2 \cong \sigma^2 \tag{23}$$

where σ^2 is the unknown variance of the residuals. If the model is good, the estimate of the variance should also be good.

In order to obtain the variances of interest (i.e., $\sigma_{\lambda_1}^{\;2}$ and $\sigma_{\lambda_2}^{\;2}$), it is necessary to calculate the covariance matrix of $\hat{\theta}$ (Var $(\hat{\theta})$). The diagonal elements of Var $(\hat{\theta})$ are the variances of each of the elements of the vector $\hat{\theta}$. The covariance matrix is approximated by

$$\text{Var } (\hat{\theta}) \cong S^2 \left[\sum_{j=1}^{N} \left(\frac{\partial f}{\partial \hat{\theta}}\right)_j \left(\frac{\partial f}{\partial \hat{\theta}}\right)_j^T\right]^{-1} \tag{24}$$

or

$$\text{Var } (\hat{\theta}) = [Z^T Z]^{-1} \, S^2 \tag{25}$$

where

$$Z^T = \begin{bmatrix} \frac{\partial f}{\partial \theta_{11}} & \frac{\partial f}{\partial \theta_{21}} & \cdots & \frac{\partial f}{\partial \theta_{K1}} \\ \frac{\partial f}{\partial \theta_{12}} & \frac{\partial f}{\partial \theta_{22}} & \cdots & \frac{\partial f}{\partial \theta_{K2}} \\ \vdots & & & \\ \frac{\partial f}{\partial \theta_{1N}} & \frac{\partial f}{\partial \theta_{2N}} & \cdots & \frac{\partial f}{\partial \theta_{KN}} \end{bmatrix} . \tag{26}$$

The first subscript on θ denotes the position in the vector $\hat{\theta}$ and the second subscript denotes the observation number. The estimate of the variance (S^2) is given by

$$S^2 = \left\{ \frac{\sum_{j=1}^{N} [Y_j - f(\hat{\theta}, t_j)]^2}{N - K} \right\} . \tag{27}$$

It is of interest to point out that the elements of the Z matrix (Z_{ij}) can be computed from an analytical expression. In particular:

$$\left(\frac{\partial f}{\partial \theta_{1j}}\right) = 1 . \tag{28}$$

$$\left(\frac{\partial f}{\partial \theta_{2j}}\right) = e^{\lambda_1 t_j} \tag{29}$$

$$\left(\frac{\partial f}{\partial \theta_{3j}}\right) = e^{\lambda_2 t_j} \tag{30}$$

$$\left(\frac{\partial f}{\partial \theta_{4j}}\right) = a_1 \, t_j \, e^{\lambda_1 t_j} \tag{31}$$

$$\left(\frac{\partial f}{\partial \theta_5}\right)_j = a_2 \, t_j \, e^{\lambda_2 t_j} \quad . \tag{32}$$

The above descriptions of the basic theory are appropriate for calculating estimates of $\sigma^2_{\theta_i}$. Next, the method for calculating the variance of the time constant is given.

An estimate of the plunge test response (obtained from the LCSR data) is given by

$$Y_{pl}(t) \cong K\left[\frac{1}{(-\lambda_1)(-\lambda_2)} + \frac{e^{\lambda_1 t}}{\lambda_1(\lambda_1-\lambda_2)} + \frac{e^{\lambda_2 t}}{\lambda_2(\lambda_2-\lambda_1)}\right] \quad . \tag{33}$$

The time constant τ is defined when

$$Y_{pl}(\tau) = Y_{pl}(\infty)(1 - e^{-1}) \quad , \tag{34}$$

and is accurately approximated by

$$\tau = \left[1 - \ln\left(1 - \frac{\lambda_1}{\lambda_2}\right)\right]\left[\frac{-1}{\lambda_1}\right] \quad . \tag{35}$$

The propagation of error formula can be used to estimate the variance in $\tau(\sigma_\tau{}^2)$ as follows:

$$\sigma_\tau{}^2 = \left(\frac{\partial \tau}{\partial \lambda_1}\right)^2 \sigma_{\lambda_1}{}^2 + \left(\frac{\partial \tau}{\partial \lambda_2}\right)^2 \sigma_{\lambda_2}{}^2 \tag{36}$$

where $\sigma_{\lambda_1}{}^2$ and $\sigma_{\lambda_2}{}^2$ are obtained from Var $(\hat{\theta})$. The applicable partial derivatives are:

$$\frac{\partial\tau}{\partial\lambda_1} = \frac{1}{\lambda_1}\left[\frac{1}{(\lambda_1-\lambda_2)} - \tau\right] \tag{37}$$

and

$$\frac{\partial\tau}{\partial\lambda_2} = \frac{1}{\lambda_2(\lambda_2-\lambda_1)} \quad . \tag{38}$$

The development presented in this section is a concise and straightforward method for estimating the standard deviation of the time constant τ. The accuracy of the estimate is primarily dependent on the validity of the model. Although methods are available for determining if a particular model is "good," these methods have not been incorporated into the overall procedure for estimating the time constant. Nevertheless, the estimate of the variance σ_τ^2 using the methods described herein should be sufficiently accurate for setting applicable bounds on the time constant.

RESULTS AND DISCUSSION

Results of several computational experiments are presented:

1. Survey calculations to evaluate the linearization method, Marquardt's method and a combination of the Marquardt and linearization methods.
2. Calculations to determine the effect of white noise and of 60 Hz in theoretical LCSR data on parameter estimates and on standard deviations of parameter estimates.
3. Calculations for parameter estimates and for uncertainties of parameter estimates using experimental data.

Each of the three nonlinear minimization algorithms tested performed well. Although Marquardt's method (in theory) converges much faster than

the linearization method near the minimum, the overall performance of the linearization method is better than Marquardt's method for the functional of interest (given by Eq. (2)). A combination of the two methods performed no better than the linearization method; consequently, the computer program (EXP. F4) for estimating the model parameters uses only the linearization method.

Results relating to the effect of white noise and 60 Hz on theoretical data illustrate that the parameter estimates are accurate; in order to verify that the estimates of the standard derivations are accurate, detailed simulation studies are needed. Regarding some of the results which follow, input data to the computer program (NØISY) preceds the output from the computer program (EXP.) which estimates the model parameters for the results relating to the effect of white noise and 60 Hz.

If the input data (to NØISY) specifies a loop current step response (LCSR) calculation, the theoretical data is composed of two (M=2) exponential terms. If a drop (DRØP) test is specified, the theoretical data is composed of three (M=3) exponentials. The value of R (input to NØISY) specifies the higher eigenvalues using the relation

$$\rho_{i+1} = (1 + R)^2 \rho_i \quad ;$$

P(1) specifies the first eigenvalue and TM specifies the time span of the data.

Even though the LCSR data are significantly contaminated with white noise (or 60 Hz), the parameter estimates are correct. Also the uncertainty in the estimates varies as is expected.

```
.RUN NOISY

  IF DROP TEST,TYPE 1,LOSR TYPE 2
2
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
20
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
0
 ENTER VALUE OF A(NOT EQUAL TO 1.)
.5
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          = -0.72812E-04
STANDARD DEVIATION OF THE RESIDUAL =  0.11854E+02
VARIANCE OF THE RESIDUAL            =  0.14051E+03
THE NORMALIZED GRADIENT             $=  0.93332E-06

EXPANSION COEFFICIENTS ARE:
    0.98650E+02 + OR -  0.15144E+01
    0.45389E+03 + OR -  0.65252E+01
    0.44219E+03 + OR -  0.70789E+01

EIGENVALUES ARE:
   -0.18318E+01 + OR -  0.37839E-01
   -0.11590E+02 + OR -  0.31660E+00

THE TIME CONSTANT =  0.63984E+00 + OR -  0.57281E-01
```

```
..RUN NOISY

  IF DROP TEST,TYPE 1,LCSR TYPE 2
2
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
0
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
20
 ENTER VALUE OF A(NOT EQUAL TO 1.)
.5
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          = -0.55385E-04
STANDARD DEVIATION OF THE RESIDUAL =  0.14192E+02
VARIANCE OF THE RESIDUAL            =  0.20142E+03
THE NORMALIZED GRADIENT             =  0.20299E-06

EXPANSION COEFFICIENTS ARE:
     0.99546E+02 + OR -  0.18187E+01
     0.44969E+03 + OR -  0.80009E+01
     0.44943E+03 + OR -  0.86309E+01

EIGENVALUES ARE:
    -0.18347E+01 + OR -  0.46343E-01
    -0.11437E+02 + OR -  0.37032E+00

THE TIME CONSTANT =  0.64036E+00 + OR -  0.63163E-01
STOP
```

```
.RUN NOISY

  IF DROP TEST,TYPE 1,LOSR TYPE 2
2
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
20
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
20
 ENTER VALUE OF A(NOT EQUAL TO 1.)
.5
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          = -0.70088E-04
STANDARD DEVIATION OF THE RESIDUAL =  0.19194E+02
VARIANCE OF THE RESIDUAL            =  0.36840E+03
THE NORMALIZED GRADIENT             =  0.31094E-06

EXPANSION COEFFICIENTS ARE:
    0.98705E+02 + OR -  0.24540E+01
    0.45362E+03 + OR -  0.10609E+02
    0.44153E+03 + OR -  0.11498E+02

EIGENVALUES ARE:
   -0.18320E+01 + OR -  0.61444E-01
   -0.11562E+02 + OR -  0.51270E+00

THE TIME CONSTANT =  0.64002E+00 + OR -  0.72971E-01
STOP

END OF EXECUTION
```

```
.RUN NOISY

  IF DROP TEST,TYPE 1,LCSR TYPE 2
2
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
100
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
0
 ENTER VALUE OF A(NOT EQUAL TO 1.)
.5
TYPE RAD.DAT
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          = -0.54574E-03
STANDARD DEVIATION OF THE RESIDUAL =  0.59256E+02
VARIANCE OF THE RESIDUAL            =  0.35113E+04
THE NORMALIZED GRADIENT             =  0.26418E-05

EXPANSION COEFFICIENTS ARE:
     0.95535E+02 + OR -  0.74315E+01
     0.47068E+03 + OR -  0.29852E+02
     0.41031E+03 + OR -  0.33170E+02

EIGENVALUES ARE:
   -0.18278E+01 + OR -  0.17363E+00
   -0.12244E+02 + OR -  0.17640E+01

THE TIME CONSTANT =  0.63557E+00 + OR -  0.12345E+00
STOP
```

```
.RUN NOISY

  IF DROP TEST,TYPE 1,LCSR TYPE 2
1
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.0
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
0
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
0
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          =  0.23207E-04
STANDARD DEVIATION OF THE RESIDUAL  =  0.57958E+00
VARIANCE OF THE RESIDUAL            =  0.33591E+00
THE NORMALIZED GRADIENT             =  0.58851E-03

EXPANSION COEFFICIENTS ARE:
    0.10038E+03 + OR -  0.77393E-01
    0.11514E+04 + OR -  0.40435E+00
   -0.25885E+03 + OR -  0.42130E+00

EIGENVALUES ARE:
   -0.18485E+01 + OR -  0.83426E-03
   -0.10169E+02 + OR -  0.24804E-01

THE TIME CONSTANT =  0.64953E+00 + OR -  0.84782E-02
```

```
.RUN NOISY

  ?F DROP TEST,TYPE 1,LCSR TYPE 2
1
  TYPE VALUES OF R,P(1), TM SPAN(SEC)
1.5 -1.835 2.0
 HOW MUCH NOISE?,10=1% INITIAL,10%FINAL,ETC.
10
 MAGNITUDE OF 60HZ? 10=1%INITIAL,ETC.
0
STOP


RESULTS FORM THE MINIMIZATION ALGORITHM


EXPECTED VALUE OF RESIDUAL          =  0.32484E-04
STANDARD DEVIATION OF THE RESIDUAL =  0.59608E+01
VARIANCE OF THE RESIDUAL            =  0.35531E+02
THE NORMALIZED GRADIENT             =  0.24980E-05

EXPANSION COEFFICIENTS ARE:
    0.99953E+02 + OR -  0.79956E+00
    0.11537E+04 + OR -  0.42231E+01
   -0.26296E+03 + OR -  0.43937E+01

EIGENVALUES ARE:
   -0.18481E+01 + OR -  0.86388E-02
   -0.10081E+02 + OR -  0.25012E+00

THE TIME CONSTANT =  0.65068E+00 + OR -  0.27290E-01
```

Experimental data is analyzed using two computer programs (EXP and LCSR) for parameter estimation; both programs yield essentially the same results. Although several experimental data files have been analyzed with both programs, only results for H3727 are presented. The following results are from LCSR.

PARAMETERS AFTER LEAST SQUARES CYCLE 6

	OLD	CHANGE	NEW	'ERROR'
1	-0.6058	0.0000	-0.6057	0.0019
2	1.0247	0.0002	1.0248	0.0019
3	0.6536	-0.0000	0.6536	0.0016
4	3985.8394	0.0166	3985.8543	0.8443
5	201.6351	-0.0451	201.5945	1.4054

Results from EXP are:

```
RESULTS FORM THE MINIMIZATION ALGORITHM:

ABNORMAL CONVERGENCE, NOPT =  1

EXPECTED VALUE OF RESIDUAL            =  0.46652E-02
STANDARD DEVIATION OF THE RESIDUAL =  0.30850E+01
VARIANCE OF THE RESIDUAL              =  0.95169E+01
THE NORMALIZED GRADIENT               =  0.14890E+02

EXPANSION COEFFICIENTS ARE:
    0.20191E+03 + OR -  0.13669E+01
    0.24744E+04 + OR -  0.54419E+01
    0.13096E+04 + OR -  0.63062E+01

EIGENVALUES ARE:
   -0.60614E+00 + OR -  0.18687E-02
   -0.24854E+01 + OR -  0.10189E-01

THE TIME CONSTANT =  0.21110E+01 + OR -  0.11448E+00
```

The first and second eigenvalues from LCSR are -0.6058 and -2.47 compared to -0.6061 and -2.48 from EXP. Final values are 201.6 from LCSR and 201.9 from EXP. Note that EXP converged by not being able to reduce the functional value after halving the interval ten times rather than by obtaining a small normalized

gradient. Also the expected value of the residual is not near zero. This indicates that parameters are slightly biased and that the model is apparently inadequate. This conclusion is supported by other unusual features of this particular data set. Other experimental data sets converged on the criteria of a small normalized gradient.

CONCLUSIONS AND RECOMMENDATIONS

Comparisons of the linearization minimization algorithm with Marquardt's method or a combination of Marquardt's method with linearization indicate that the linearization algorithm is slightly better for the functional of interest. Also, in theory, the linearization algorithm should always converge to a local minimum.

Test calculations using EXP on theoretical data and on experimental data indicate that this computer program (EXP) has no apparent errors. It is an improvement over an existing computer program (LCSR) for estimating model parameters related to LCSR data in that EXP does not require initialization and that it also provides useful statistical information about model parameters and a calculated parameter (i.e., the 63% response time).

It is recommended that the existing computer program (EXP) be upgraded in two ways: (1) that the data be digitally filtered (or smoothed) consistent with the sampling theorem so that the number of data points used in the analysis is not more than is needed, and (2) that a statistical analysis of the residual error be included to check model validity.

REFERENCES

1. T. W. Kerlin, L. F. Miller, J. E. Mott, B. R. Upadhyaya, H. M. Hashemian, and J. S. Arendt, *In-Situ Response Time Testing of Platinum Resistance Thermometers*, EPRI RP503-3 (1976).

2. R. M. Carroll, "Monthly Report for December 1974 and Summary of Model," Intra-Laboratory correspondence (ORNL).

3. N. R. Draper and H. Smith, *Applied Regression Analysis*, John Wiley & Sons, Inc. (1966).

4. Y. Bard, *Nonlinear Parameter Estimation*, Academic Press, pp. 176-179 (1974).

5. C. Bayne, "Personal Communications Related to Uncertainty in Unbiased Parameter Estimates," Mathematics Research (ORNL).

6. Y. Beers, *Introduction to the Theory of Error*, Addison-Wesley (1957).

7. T. W. Kerlin, "Notes for a TIW-12 Short Course," Department of Nuclear Engineering, University of Tennessee, Knoxville, Tennessee (June 1977).

APPENDIX

LISTING OF THE COMPUTER PROGRAM

EXP. F4

```
*************************************************************************
*************************************************************************
*************************************************************************
C  THIS PROGRAM IDENTIFIES PARAMETERS OF AN EXPONENTIAL MODEL THAT
C  APPROXIMATES THE LOOP-CURRENT-STEP-RESPONSE(LCSR) DATA.  THE OVERALL
C  PROGRAM LOGIC IS AS FOLLOWS:
C   1. CONDUCT A DIRECT SEARCH FOR A STARTING POINT
C   2. MINIMIZE THE VARIANCE OF THE RESIDUAL ERROR
C   3. ESTIMATE THE STANDARD DEVIATION OF THE 63% RESPONSE TIME FROM
C      THE VARIANCE-COVARIANCE MATRIX
C*************************************************************************
      DIMENSION YO(1024),T(1024),X(7),Y(7),A(7,7),AIV(7,7),
     1 ASAV(7,7),P(7),G(7),WK(7),DIR(7),PSAV(7)
      ND=7
      NEXP=0
      NOUT=10
      NMAX=10
      NSRH=16
      NOMX=2
      EPS=1.E-4
      TAU=1.0
C*************************************************************************
C  INITIALIZE THE INPUT DATA FILE AND READ THE INPUT DATA FROM A
C  FILE NAMED 'RAD'
C  NOMX: NUMBER OF OBSERVATIONS
C  NEXP: IF SET TO 1, A THREE EXPONENTIAL CALCULATION IS PERFORMED
C  NPT : IF SET TO 1, SELECTED INPUT DATA IS PRINTED
C  INC : SKIP FACTOR FOR PRINTING OBSERVATION DATA
      CALL IFILE(1,'TST')
      READ(1,1015)NOMX,NEXP,NPT,INC
1015  FORMAT(4I)
      CALL IFILE(2,'RAD')
      TL=0.0
      NO=0
10    NO=NO+1
      READ(2,1007,END=20,ERR=20)YO(NO),DUMY,T(NO)
1007  FORMAT(5X,3E)
      IF(NO.GE.1024.OR.T(NO).LT.TL)GO TO 20
      TL=T(NO)
      IF(NO.GT.NOMX)GO TO 20
      GO TO 10
20    NO=NO-1
      IF(NPT.EQ.1)PRINT 1005,(T(K),YO(K),K=1,NO,INC)
1005  FORMAT(' ','  SELECTED INPUT OBSERVATION DATA:'/
     1 '       TIME          OBSERVATION'/(' ',2(2X,E12.5)))
```

```
C*********************************************************************
C  CONDUCT A DIRECT SEARCH TO ESTIMATE A STARTING  POINT
      AR=1.0
      IOPT=0
      NX=2
      RHO=0.0
      FMIN=1.0E+12
      NV=2*NX+1
      BU=TAU*5.0
      BL=TAU*0.2
      RTO=(BU/BL)**(1.0/FLOAT(NSRH-1))
      NLP=NSRH
      DO 40 I=1,NLP
      P(4)=-BU/AR
      P(5)=P(4)*7.0
      IER=0
      CALL FUNCT(NO,NX,ND,A,X,Y,P,T,YO,IOPT,IER,FVAL)
      IF(IER.EQ.0)GO TO 35
      PRINT 1013, IER
1013  FORMAT('0','  ERROR IN THE COEFFICIENT CALCULATION, IER=',I5)
      STOP
35    CONTINUE
      AR=AR*RTO
      IF(FVAL.GE.FMIN)GO TO 40
      FMIN=FVAL
      CALL VECDP(NV,RHO,P,WK,DIR)
40    CONTINUE
      CALL VECDP(NV,RHO,WK,P,DIR)
C*********************************************************************
C  PRINT RESULTS OF THE DIRECT SEARCH
      EIG1=P(4)
      EIG2=P(5)
      CALL TAUA(TAU,EIG1,EIG2)
      PRINT 1000
      FVAL=FMIN
      PRINT 1001, FVAL,TAU,P(1),P(2),P(3),P(4),P(5)
1000  FORMAT('0','  RESULTS OF THE DIRECT SEARCH:')
1001  FORMAT('0','  THE FUNCTIONAL VALUE        = ',3X,E12.5/
     1' ','  THE TIME CONSTANT            = ',3X,E12.5/
     2' ','  THE EXPANSION COEFFICIENTS = ',3(3X,E12.5)/
     3' ','  THE EIGENVALUES              =  ',2(3X,E12.5))
C*********************************************************************
C  MINIMIZE THE RESIDUAL VARIANCE
      NX=2
50    CONTINUE
      PRINT 1008, NX
1008  FORMAT('0','  THE FOLLOWING RESULTS PERTAIN TO AN '/
     1' APPROXIMATING MODEL WITH ',I2,' EXPONENTIALS:')
      CALL LNR(NO,NX,ASAV,P,T,ND,WK,DIR,A,X,Y,YO,EPS,NMAX,NOUT,
     1 NOPT,FMIN,FVAL)
```

```
      PRINT 1004
1004  FORMAT('0','  RESULTS FORM THE MINIMIZATION ALGORITHM:')
      IF(NOPT.EQ.0)GO TO 60
      PRINT 1010, NOPT
1010  FORMAT(' ','  ABNORMAL CONVERGENCE, NOPT = ',I2)
60    CONTINUE
      CALL SWAP(NX,WK,P)
C*****************************************************************************
C  CALCULATE THE STANDARD DEVIATIONS OF THE ESTIMATED PARAMETERS AND OF
C  THE 63% TIIME CONSTANT
      CALL TSG(A,ASAV,AIV,X,Y,YO,T,P,ND,NO,NX)
      IF(NEXP.NE.1)GO TO 70
      P(NV+2)=3.0*P(NV)
      P(NV+1)=P(NV)
      P(NV)=P(NV-1)
      NX=NX+1
      IF(NX.GT.3)GO TO 70
      IOPT=0
      CALL FUNCT(NO,NX,ND,A,X,Y,P,T,YO,IOPT,IER,FVAL)
      FMIN=FVAL
      IF(IER.EQ.0)GO TO 65
      PRINT 1013,IER
65    CONTINUE
      GO TO 50
70    CONTINUE
      STOP
      END
C*****************************************************************************
C*****************************************************************************
      SUBROUTINE LNR(NO,NX,ASAV,P,T,ND,WK,DIR,A,X,Y,YO,EPS,NMAX,
     1 NOUT,NOPT,FMIN,FVAL)
      DIMENSION ASAV(ND,1),P(1),T(1),WK(1),DIR(1),A(ND,1),X(1),
     1 Y(1),YO(1),F(7),R(7)
C*****************************************************************************
C  THIS SUBROUTINE USES THE LINEARIZATION METHOD IN CONJUNCTION WITH
C  A VARIABLE STEP SIZE.  SINCE THE COEFFICIENT MATRIX FOR CALCULATING
C  THE SEARCH DIRECTION IS POSITIVE DEFINITE THE DIIRECTION VECTOR
C  SHOULD ALWAYS BE IN A DIRECTION THAT DECREASES THE FUNCTIONAL.  BY
C  DETERMINING THE OPTIMUM STEP SIZE, THIS MINIMIZATION ALGORITHM
C  SHOULD ALWAYS CONVERGE.
C*****************************************************************************
      NV=2*NX+1
      NOLSS=0
      FOLD=FVAL
      ZRO=0.0
      NS=0
5     NS=NS+1
      IOPT=0
      IF((NS/2)*2.EQ.NS)IOPT=1
```

```
C*********************************************************************
C  CALCULATE THE SEARCH DIRECTION
10      CONTINUE
        CALL ZTZ(NO,NX,RSAV,P,T,ND)
        DO 15 I=1,NV
15      Y(I)=0.0
        DO 20 K=1,NO
        TM=T(K)
        RID=YO(K)-FAPP(P,NX,TM)
        DO 20 I=1,NV
        K1=I
        CALL DPV(K1,NX,TM,P,DR)
20      Y(I)=Y(I)+RID*DR
        CALL GAUS(ASAV,Y,DIR,ND,NV)
C*********************************************************************
C  DETERMINE THE OPTIMUM STEP SIZE
        NOPT=0
        NFAIL=0
        ISA=1
        RHO=RSAV
25      CONTINUE
C  VERIFY THAT THE SEARCH POINT DOES NOT INCLUDE A POSITIVE EIGENVALUE
        CALL DRCHK(DIR,P,NX,RHO,WK,ISET)
        CALL VECDP(NV,RHO,P,WK,DIR)
30      IER=0
        CALL FUNCT(NO,NX,ND,A,X,Y,WK,T,YO,IOPT,IER,FVAL)
        IF(IER.EQ.0)GO TO 40
        IOPT=1
        GO TO 30
40      CONTINUE
        IF(FVAL.GE.FMIN.AND.ISA.EQ.1)GO TO 45
        PCT=(FMIN-FVAL)/FMIN
        IF(PCT.LT.ZRO)GO TO 42
C  EXTEND THE LINE SEARCH UNTIL THE FUNCTIONAL VALUE IS INCREASED
        RSAV=RHO
        FMIN=FVAL
        CALL VECDP(NV,ZRO,WK,X,DIR)
        IF(ISA.EQ.3)GO TO 50
        IF(ISET.EQ.1.AND.ISA.NE.1)GO TO 50
        ISA=2
        RHO=RHO*1.618
        GO TO 25
42      CONTINUE
C  CALCULATE THE OPTIMUM STEP SIZE USING A QUADRATIC APPROXIMATION
        IF(ISA.EQ.3)GO TO 50
        ISA=3
        Y1=FMIN-FOLD
        Y2=FVAL-FOLD
        DET=RSAV*RHO**2-RHO*RSAV**2
        A1=(Y1*RHO**2-Y2*RSAV**2)/DET
        B1=(RSAV*Y2-RHO*Y1)/DET
        RHO=-0.5*A1/B1
        GO TO 25
```

```
C  IF THE FUNCTIONAL IS NOT LOWERED, THE INTERVAL IS HALVED NMAX
C  TIMES IN BOTH DIRECTIONS OF THE LINE SEARCH
45      NFAIL=NFAIL+1
        RHO=RHO*0.5
        IF(NFAIL.LE.NMAX)GO TO 25
        NOPT=1
        IF(RHO.LE.0.0)GO TO 70
        RHO=-RSAV
        NFAIL=0
        NOPT=0
        GO TO 25
C*********************************************************************
C  CHECK CONVERGENCE
50      CALL VECDP(NV,ZRO,X,P,DIR)
        PCT=(FOLD-FMIN)/FOLD
        IF(PCT.GT.EPS)GO TO 55
        CALL GRAD(NO,NX,YO,P,T,X,GRD)
        NOLSS=NOLSS+1
        IF(GRD.GT.10.0*EPS*FMIN)GO TO 60
        NOPT=0
        GO TO 70
55      NOLSS=0
60      CONTINUE
        PRINT 1000, NS,FMIN,FOLD
1000    FORMAT(' ','  NEW AND OLD FUNCTIONAL VALUES FOR ITERATION ',I2/
     1  '      ARE  ',E12.5,'  AND  ',E12.5)
        FOLD=FMIN
        IF(NS.LT.NOUT.AND.NOLSS.LT.3)GO TO 5
        IF(NS.EQ.NOUT)NOPT=2
        IF(NOLSS.EQ.3)NOPT=3
70      CONTINUE
        RETURN
        END
C*********************************************************************
C*********************************************************************
        SUBROUTINE DRCHK(DIR,P,NX,RHO,WK,ITST)
        DIMENSION DIR(1),P(1),WK(1)
C*********************************************************************
C  THIS SUBROUTINE ENSURES THAT THE EIGENVALUES FOR THE SEARCH POINTS
C  ARE ALWAYS NEGATIVE.
C*********************************************************************
        NX2=NX+2
        NV=2*NX+1
        ITST=0
        TST=ABS(RHO)
        DO 10 K=NX2,NV
        IF(RHO.GT.0.0.AND.DIR(K).LT.0.0)GO TO 10
        IF(RHO.LT.0.0.AND.DIR(K).GT.0.0)GO TO 10
        TERM=ABS(P(K)/DIR(K))
        IF(TERM.GT.TST)GO TO 10
        TST=TERM
10      CONTINUE
```

```
      TERM=ABS(RHO)
      IF(TST.LT.TERM)ITST=1
      IF(ITST.EQ.1)RHO=RHO*0.7*TST/TERM
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE FUNCT(NO,NX,ND,A,X,Y,P,T,YO,IOPT,IER,FVAL)
      DIMENSION A(ND,ND),Y(1),P(1),T(1),YO(1),X(1)
C*************************************************************************
C  THIS SUBROUTINE CALCULATES THE FUNCCTIONAL TO BE MINIMIZED  WHICH IS
C  THE APPROXIMATE VARIANCE OF THE RESIDUALS.  IF IOPT=0, THE OPTIMUM
C  EXPANSION COEFFICIENTS ARE CALCULATED.  IF IOPT IS NOT ZERO, THE EXPANSION
C  COEFFICIENTS ARE NOT CALCULATED.
C*********************************************************************
C  CALCULATE THE  OPTIMUM EXPANSION COEFFICIENTS IF IOPT=0
      IF(IOPT.NE.0)GO TO 10
      CALL COF(NO,NX,ND,A,X,Y,T,YO,P,IER)
      IF(IER.EQ.0)GO TO 10
      GO TO 30
10    CONTINUE
C*********************************************************************
C  EVALUATE THE FUNCTIONAL
      FVAL=0.0
      DO 20 K=1,NO
      TM=T(K)
      S1=FAPR(P,NX,TM)
20    FVAL=FVAL+(S1-YO(K))**2
      ANO=NO-(2*NX+1)
      FVAL=FVAL/ANO
30    CONTINUE
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE COF(NO,NX,ND,A,X,Y,T,YO,P,IER)
      DIMENSION A(ND,ND),X(1),Y(1),T(1),YO(1),P(1)
C*********************************************************************
C  THIS SUBROUTINE EVALUATES THE OPTIMUM EXPANSION COEFFICIENTS BY A
C  LINEAR LEAST SQUARES ALGOPITHM
C*********************************************************************
C  CALCULATE THE COEFFICIENT MATRIX
      A(1,1)=NO
      NX1=NX+1
      DO 20 I=1,NX1
      DO 20 J=I,NX1
      EX=P(NX+I)*P(NX+J)
      IF(J.EQ.1)GO TO 20
      IF(I.EQ.1)EX=P(NX+J)
      A(I,J)=0.0
      DO 10 K=1,NO
      TM=T(K)
```

```
10      A(I,J)=A(I,J)+EXP(E)*TM)
20      CONTINUE
        DO 25 I=2,NX1
        DO 25 J=1,I
25      A(I,J)=A(J,I)
C  COMPUTE THE INHOMOGENEOUS VECTOR
        DO 30 I=1,NX1
        Y(I)=0.0
        XR=P(NX+I)
        XT=1.0
        DO 30 K=1,ND
        IF(I.GT.1)XT=EXP(T(K)*XR)
30      Y(I)=Y(I)+XT*YO(K)
C  CHECK FOR A ZERO DIAGONAL ELEMENT
        IER=0
        DO 35 K=1,NX1
        IF(ABS(A(K,K)).GT.1.0E-12)GO TO 35
        IER=1
        GO TO 50
35      CONTINUE
C  CALCULATE THE EXPANSION COEFFICIENTS
        CALL GAUS(A,Y,X,ND,NX1)
        DO 40 I=1,NX1
40      P(I)=X(I)
50      CONTINUE
        RETURN
        END
C*****************************************************************************
C*****************************************************************************
        FUNCTION FAPR(P,NX,TM)
        DIMENSION P(1)
C*****************************************************************************
C  THIS FUNCTION SUBROUTINE EVALUATES THE APPROXIMATING FUNCTION AT
C  A SPECIFIED TIME
C*****************************************************************************
        NX1=NX+1
        FX=P(1)
        DO 10 I=2,NX1
10      FX=FX+P(I)*EXP(P(NX+I)*TM)
        FAPR=FX
        RETURN
        END
C*****************************************************************************
C*****************************************************************************
        SUBROUTINE VECDP(K,RHO,X,Y,DIR)
        DIMENSION X(1),Y(1),DIR(1)
C*****************************************************************************
C  THIS SUBROUTINE CALCULATES THE COORDINATES OF A POINT THAT LIES A
C  SPECIFIED DISTANCE IN A SPECIFIED DIRECTION FROM A K-DIMENSIONAL
C  VECTOR
```

```
C*********************************************************************
      DO 10 I=1,K
10    Y(I)=X(I)+RHO*DIR(I)
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE TAUA(TAU,E1,E2)
C*********************************************************************
C  THIS SUBROUTINE CALCULATES THE APPROXIMATE TIME CONSTANT(63%) FROM
C  AN ANALYTICAL EXPRESSION USING TWO EIGENVALUES.
C*********************************************************************
      TAU=(ALOG(1.00-E1/E2)-1.0)/E1
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE TSG(A,ASAV,AIV,X,Y,YO,TIM,P,ND,NO,NX)
      DIMENSION A(ND,1),ASAV(ND,1),AIV(ND,1),X(1),Y(1),YO(1),TIM(1),
     1 P(1)
C*********************************************************************
C  THIS SUBROUTINE CALCULATES THE FOLLOWING:
C     1. STANDARD DEVIATIONS OF THE MODEL PARAMETERS USING THE
C        VARIANCE-COVARIANCE MATRIX
C     2. AN ESTIMATE OF THE TIME CONSTANT USING AN ANALYTICAL
C        EXPRESSION
C     3. AN ESTIMATE OF THE STANDARD DEVIATION OF THE TIME CONSTANT
C        USING THE VARIANCE-COVARIANCE MATRIX AND THE PROPAGATION OF
C        ERROR FORMULA
C*********************************************************************
C  INPUT DATA REQUIRED BY THIS SUBROUTINE ARE AS FOLLOWS:
C     1. THE NUMBER OF OBSERVARIONS
C     2. UNBIASED PARAMETER ESTIMATES FROM THE LCSR MINIMIZATION
C        PROGRAM
C        (ITEMS 1. AND 2. ARE READ UNDER THE FILENAME 'TVDAT')
C     3. OBSERVATION DATA AND TIME VALUES ASSOCIATED WITH THE
C        OBSERVATION DATA
C        (ITEM 3. IS READ UNDER THE FILENAME 'PAD'.  THIS IS THE
C         SAME FILE USED BY THE LCSR MINIMIZATION PROGRAM)
C*********************************************************************
      NV=2*NX+1
C  CALCULATE THE MATRX:  Z TRANSPOSE Z WHICH WILL BE INVERTED TO
C  OBTAIN THE VARIANCE-COVARIANCE MATRIX
      CALL ZTZ(NO,NX,A,P,TIM,ND)
C  INVERT THE Z TRANSPOSE Z MATRIX TO OBTAIN THE VARIANCE-
C  COVARIANCE MATRIX
      IER=0
      N=NV
      CALL INV(A,AIV,ASAV,X,Y,ND,N,IER)
      IF(IER.EQ.0)GO TO 30
      PRINT 1005, IER
```

```
1005   FORMAT(' ','  A ZERO DIAGONAL ELEMENT WAS ENCOUNTERED '/
     1'          IER = ',I5)
30     CONTINUE
C*********************************************************************
C  ESTIMATE THE STANDARD DEVIATIN AND THE EXPECTED VALUE OF THE
C  RESIDUAL ERROR.  ALSO CALCULATE THE NORMALIZED GRADIENT.
       CALL VAR(P,NX,NO,TIM,YO,NV,STD,EXV)
       AR=STD**2
       CALL GRAD(NO,NX,YO,P,TIM,X,GRD)
       GRD=GRD/AR
       PRINT 1006, EXV,STD,AR,GRD
1006   FORMAT('0','  EXPECTED VALUE OF RESIDUAL          = ',E12.5/
     1 '    STANDARD DEVIATION OF THE RESIDUAL = ',E12.5/
     2 '    VARIANCE OF THE RESIDUAL           = ',E12.5/
     3 '    THE NORMALIZED GRADIENT            = ',E12.5)
C*********************************************************************
C  CALCULATE THE STANDARD DEVIATIONS OF THE EXPANSION COEFFICIENTS
C  AND OF THE EIGENVALUES
       DO 40 K=1,NV
       Y(K)=AIV(K,K)*STD**2
       XA=ABS(Y(K))
40     Y(K)=SQRT(XA)
       NX1=NX+1
       PRINT 1007, (P(K),Y(K),K=1,NX1)
1007   FORMAT('0','  EXPANSION COEFFICIENTS ARE: '/
     1 (' ',5X,E12.5,' + OR - ',E12.5))
       PRINT 1008,(P(NX1+K),Y(NX1+K),K=1,NX)
1008   FORMAT('0','  EIGENVALUES ARE: '/
     1 (' ',5X,E12.5,' + OR - ',E12.5))
C*********************************************************************
C  CALCULATE THE TIME CONSTANT AND THE STANDARD DEVIATION OF THE
C  TIME CONSTANT
       SG1=Y(NX1+1)
       SG2=Y(NX1+2)
       CALL TVAR(SG1,SG2,TAU,TSIG,P,NX)
       PRINT 1009, TAU,TSIG
1009   FORMAT('0','  THE TIME CONSTANT = ',E12.5,' + OR - ',E12.5)
       RETURN
       END
C*********************************************************************
C*********************************************************************
       SUBROUTINE GRAD(NO,NX,YO,P,T,G,GRD)
       DIMENSION YO(1),P(1),T(1),G(1)
C*********************************************************************
C  THIS SUBROUTINE EVALUATES THE GRADIENT OF THE FUNCTIONAL WITH
C  RESPECT TO EACH PARAMETER AND EVALUATES THE NORM OF THE GRADIENT.
C*********************************************************************
       NV=2*NX+1
       DO 5 J=1,NV
5      G(J)=0.0
       DO 15 K=1,NO
       TM=T(K)
```

```
      RID=FAPR(P,NX,TM)-YO(K)
      G(1)=G(1)+RID
      DO 10 I=2,NV
      K1=I
      CALL DRV(K1,NX,TM,P,DIV)
10    G(I)=G(I)+RID*DIV
15    CONTINUE
      AT=2.0/FLOAT(NO-NV)
      DO 20 K=1,NV
20    G(K)=G(K)*AT
      GRD=0.0
      DO 30 K=1,NV
30    GRD=GRD+G(K)**2
      RETURN
      END
C*****************************************************************************
C*****************************************************************************
      SUBROUTINE ZTZ(NO,NX,A,P,T,ND)
      DIMENSION A(ND,ND),P(1),T(1)
C*****************************************************************************
C  THIS SUBROUTINE CALCULATES THE SQUARE MATRIX NEEDED TO INVERT
C  TO OBTAIN THE VARIANCE-COVARIANCE MATRIX
C*****************************************************************************
      NV=2*NX+1
      DO 10 I=1,NV
      DO 10 J=I,NV
      A(I,J)=0.0
      DO 10 K=1,NO
      K1=I
      K2=J
      TM=T(K)
      CALL DRV(K1,NX,TM,P,FV1)
      CALL DRV(K2,NX,TM,P,FV2)
      A(I,J)=A(I,J)+FV1*FV2
10    CONTINUE
      DO 20 I=1,NV
      DO 20 J=1,I
      A(I,J)=A(J,I)
20    CONTINUE
      RETURN
      END
C*****************************************************************************
C*****************************************************************************
      SUBROUTINE DRV(K,NX,T,P,FVAL)
      DIMENSION P(1)
C*****************************************************************************
C  THIS SUBROUTINE CALCULATES THE PARTIAL DERIVATIVES OF THE
C  APPROXIMATING FUNCTION WITH RESPECT TO EACH OF THE MODEL PARAMETERS
C  THE EXPANSION COEFFICIENTS ARE IN THE FIRST NX1 ELEMENTS OF THE
C  P VECTOR AND THE EIGENVALUES ARE IN THE LAST NX ELEMENTS OF THE
C  VECTOR.
```

```
C*****************************************************************
      NX1=NX+1
      IF(K.GT.1)GO TO 10
      FVAL=1.0
      GO TO 30
10    CONTINUE
      IF(K.GT.NX1)GO TO 20
      FVAL=EXP(T*P(NX+K))
      GO TO 30
20    CONTINUE
      FVAL=P(K-NX)*T*EXP(T*P(K))
30    CONTINUE
      RETURN
      END
C*****************************************************************
C*****************************************************************
      SUBROUTINE INV(A,AIV,ASAV,X,Y,ND,N,IER)
      DIMENSION A(ND,ND),AIV(ND,ND),ASAV(ND,ND),X(1),Y(1)
C*****************************************************************
C  THIS SUBROUTINE CALCULATES THE INVERSE BY REPEATED CALLS TO A
C  LINEAR EQUATION SOLVER WHHICH USES DIRECT GAUSSIAN ELIMINATION.
C  THUS, NONE OF THE DIAGONAL ELEMENTS CAN BE ZERO.  A CHECK IS
C  MADE FOR ZERO DIAGONAL ELEMENTS EVEN THOUGH NONE SHOULD BE ZERO.
C*****************************************************************
C  TEST FOR ZERO DIAGONAL ELEMENTS
      DO 2 K=1,N
      IF(ABS(A(K,K)).GT.1.0E-12)GO TO 2
      IER=1
      GO TO 30
2     CONTINUE
      NS=0
10    NS=NS+1
C  GENERATE THE APPROPRIATE UNIT VECTOR
      DO 15 K=1,N
      Y(K)=0.0
      IF(K.EQ.NS)Y(K)=1.0
15    CONTINUE
      DO 16 I=1,N
      DO 16 J=1,N
16    ASAV(I,J)=A(I,J)
      CALL GAUS(ASAV,Y,X,ND,N)
      DO 20 I=1,N
20    AIV(I,NS)=X(I)
      IF(NS.LT.N)GO TO 10
30    CONTINUE
      RETURN
      END
C*****************************************************************
C*****************************************************************
      SUBROUTINE GAUS(A,Y,X,ID,N)
      DIMENSION A(ID,ID),Y(1),X(1)
```

```
C*****************************************************************
C  THIS SUBROUTINE SOLVES A SET OF LINEAR EQUATIONS BY DIRECT ELININATION
C*******************************************************************
      M=N-1
      DO 10 I=1,M
      L=I+1
      DO 10 J=L,N
      IF(A(J,I))6,10,6
6     DO 8 K=L,N
8     A(J,K)=A(J,K)-A(I,K)*A(J,I)/A(I,I)
      Y(J)=Y(J)-Y(I)*A(J,I)/A(I,I)
10    CONTINUE
      X(N)=Y(N)/A(N,N)
      DO 30 I=1,M
      K=N-I
      L=K+1
      DO 20 J=L,N
20    Y(K)=Y(K)-X(J)*A(K,J)
      X(K)=Y(K)/A(K,K)
30    CONTINUE
      RETURN
      END
C*****************************************************************
C*******************************************************************
      SUBROUTINE VAR(P,NX,NO,T,YO,NV,SUM,EXVAL)
      DIMENSION P(1),T(1),YO(1)
C*****************************************************************
C  THIS SUBROUTINE ESTIMATES THE VARIANCE OF THE DATA FORM THE RESIDUALS.
C  IT ALSO CALCULATES THE EXPECTED VALUE OF THE RESIDUALS
C*****************************************************************
      NX1=NX+1
      EXVAL=0.0
      SUM=0.0
      DO 10 K=1,NO
      TM=T(K)
      S1=P(1)
      DO 5 J=1,NX
5     S1=S1+P(J+1)*EXP(P(NX1+J)*TM)
      EXVAL=EXVAL+YO(K)-S1
10    SUM=SUM+(YO(K)-S1)**2
      ANO=NO
      EXVAL=EXVAL/ANO
      SUM=SUM/(ANO-NV)
      SUM=SQRT(SUM)
      RETURN
      END
C*****************************************************************
C*******************************************************************
      SUBROUTINE SWAP(NX,WK,P)
C*****************************************************************
C  THIS SUBROUTINE ORDERS THE EIGENVALUES AND ASSOCIATED EXPANSION
C  COEFFICIENTS.  THE TIME CONSTANT CALCULATION AND TIME CONSTANT
C  VARIANCE CALCULATION ASSUMES ORDEREDD EIGENVALUES AND EXPANSION
C  COEFFICIENTS.
```

```
C*********************************************************************
      DIMENSIONWK(1),P(1)
      NX1=NX+1
      NV=2*NX+1
      DO 20 I=1,NX
      ETST=-1.E+12
      DO 10 J=1,NX
      IF(P(NX1+J).LE.ETST)GO TO 10
      ETST=P(NX1+J)
      JSAV=J
10    CONTINUE
      WK(JSAV+NX1)=P(JSAV+NX1)
      P(JSAV+NX1)=-2.0E+12
      WK(JSAV+1)=P(JSAV+1)
20    CONTINUE
      DO 30 I=2,NV
30    P(I)=WK(I)
      RETURN
      END
C*********************************************************************
C*********************************************************************
      SUBROUTINE TVAR(SG1,SG2,TAU,TSIG,P,NX)
      DIMENSION P(1)
C*********************************************************************
C  THIS SUBROUTINE CALCULATES THE TIME CONSTANT USING THE FIRST TWO
C  (ORDERED) EIGENVALUES IN AN ANALYTICAL EXPRESSION.  THE STANDARD
C  DEVIATION IS CALCULATED USING THE PROPAGATION OF ERROR FORMULA
C*********************************************************************
      NX1=NX+1
      X1=P(NX1+2)
      X2=P(NX1+1)
      X3=X1-X2
      X4=X3/X1
      TAU=(ALOG(X4)-1.)/X2
      DV1=-(1./X3+TAU)/X2
      DV2=1./(X1*X3)
      SIG2=SG1*DV1**2+SG2*DV2**2
      TSIG=SQRT(SIG2)
      RETURN
      END
```

INTERNAL DISTRIBUTION

1. B. G. Eads
2-6. L. F. Miller
7. L. C. Oakes
8-9. R. L. Shepard
10. R. S. Wiltshire
11. M. J. Kopp (Advisor)
12. P. F. McCrea (Advisor)
13. H. M. Paynter (Advisor)
14. H. E. Trammell (Advisor)
15. Central Research Library
16. Y-12 Document Reference Library
17-18. Laboratory Records
19. Laboratory Records-RC
20. ORNL Patent Section
21. I&C Publications and Information Processing Center

EXTERNAL DISTRIBUTION

22. Office of Assistant Manager for Energy Research and Development, U.S. Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37831

23-24. Office of Scientific and Technical Information, Oak Ridge, TN 37831