LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

MASTER

ORNL/CSD/TM-189

**SURE: A System of Computer Codes for Performing
Sensitivity/Uncertainty Analyses With the RELAP Code**

M. A. Bjerke

Date Published - February 1983

**COMPUTER SCIENCES
at
Oak Ridge National Laboratory
Post Office Box X
Oak Ridge, Tennessee 37830**

ORNL/CSD/TM-189
7883 000623

**Union Carbide Corporation, Nuclear Division
operating the
Oak Ridge Gaseous Diffusion Plant Paducah Gaseous Diffusion Plant
Oak Ridge Y-12 Plant Oak Ridge National Laboratory
under Contract No. W-7405-eng-26
for the
Department of Energy**

NOTICE

**PORTIONS OF THIS REPORT ARE ILLEGIBLE. It
has been reproduced from the best available
copy to permit the broadest possible avail-
ability.**

TABLE OF CONTENTS

	<u>Page</u>
LIST OF FIGURES	v
ACKNOWLEDGMENTS	vii
ABSTRACT	ix
1. INTRODUCTION	1
2. ANALYSIS METHOD	3
2.1 Response Surface Method	3
2.2 Use of Sensitivity Coefficients in Other Uncertainty Analysis Methods	6
3. ANALYSIS PROCEDURES	9
3.1 Parameter Perturbation and RELAP Runs	11
3.2 Accessing the RELAP Output Files	11
3.3 Conversion of RELAP Responses to Binary Files	15
3.4 Calculation of Sensitivity Coefficients	15
3.5 Uncertainty Analysis	15
3.6 Producing Edits of All Binary Files	16
4. INPUT SPECIFICATIONS	17
4.1 Parameter Perturbation and RELAP Runs	18
4.2 Accessing the RELAP Output Files	21
4.3 Conversion of RELAP Responses to Binary Files	23
4.4 Calculation of Sensitivity Coefficients	26
4.5 Uncertainty Analysis	35
4.6 Editing of Binary Files	38
5. FURTHER SYSTEM DEVELOPMENT	41
5.1 Automated Processing	41
5.2 Conversion to a Different Base Code	42
REFERENCES	47

TABLE OF CONTENTS (Cont.)

	<u>Page</u>
Appendix A - File Specifications	49
A.1 RELAP Responses	50
A.2 Sensitivity Coefficients	51
A.3 Mixed Derivatives	53
A.4 Uncertainty Analysis Results	54
A.5 Response Covariances	55
Microfiche	Back Cover

LIST OF FIGURES

	<u>Page</u>
Figure 1. Computational Procedure Flow Chart	10
Figure 2. PDP-10 File for Running REDSEL and RLPSFLUX	12
Figure 3. PDP-10 File for Running REDPLOT	14
Figure 4. MIC File for Automated Processing	43

ACKNOWLEDGMENTS

This work was funded by the Engineering Technology Division. Thanks are due to W. G. Craddick of that organization for his support. The helpful suggestions of E. Wacholder in the design of the system and the typing of the report by J. Patton are also greatly appreciated.

ABSTRACT

A package of computer codes has been developed to perform a non-linear uncertainty analysis on transient thermal-hydraulic systems which are modeled with the RELAP computer code. Using an uncertainty analysis methodology developed by Ronen, the package has been tailored around the analyses of experiments in the FWR-BDHT Separate Effects Program at Oak Ridge National Laboratory. The use of FORTRAN programs running interactively on the PDP-10 computer has made the system very easy to use and provided great flexibility in the choice of processing paths. Several experiments simulating a loss-of-coolant accident in a nuclear reactor have been successfully analyzed. It has been shown that the system can be automated easily to further simplify its use and that the conversion of the entire system to a base code other than RELAP is possible.

1. INTRODUCTION

The accurate prediction of the transient behavior of a nuclear reactor during a hypothetical loss-of-coolant accident is an important part of reactor safety research. Decisions on the licensing and regulation of nuclear reactors are based partly on computer code simulations of this transient behavior. In recent years, uncertainty analyses have been applied to these code simulations in order to quantify the uncertainties in the calculated responses due to uncertainties in the input parameters.

The RELAP¹ computer code is a major tool for performing thermal-hydraulic reactor calculations. A package of computer codes has been developed to perform uncertainty analyses on transient thermal-hydraulic systems which are modeled with this code. These codes compute first and second order sensitivity coefficients and use them in a nonlinear uncertainty analysis methodology developed by Ronen.² (Alternatively, the coefficients may be passed to other uncertainty analysis codes. See Sect. 2.2 of this report). The package has been tailored around the analyses of experiments in the PWR-BDHT Separate Effects Program³ of the Engineering Technology Division (ETD) at Oak Ridge National Laboratory (ORNL). Much generality has been built into the software to allow for the varied needs of the user. FORTRAN programs running interactively on the PDP-10 computer greatly simplify the use of the codes. Several experiments at the Thermal-Hydraulic Test Facility (THTF) at ORNL have been successfully analyzed.

The package consists of a number of PDP-10 disk files. These files contain (1) the JCL required to run a locally modified version of the RELAP code (RLPSFLUX⁴) on the IBM 370/3033, (2) the JCL and input data for several auxiliary codes^{5,6} that manipulate the input to and output from RLPSFLUX and, (3) a number of interactive programs on the PDP-10 which do the actual sensitivity and uncertainty calculations.

Originally developed in 1980 to perform an uncertainty analysis on an experimental test simulating a LOCA in a nuclear fuel bundle, the codes have been extensively upgraded to accommodate a wider range of computational needs. These improvements include extending the uncertainty analysis to calculate response covariances as well as variances, providing a method for calculating more realistic sensitivity coefficients and allowing more control over the format of the printed output.

Although the system is designed to run interactively on the PDP-10, only minor reprogramming would be required to convert the entire system to a batch mode. The input instructions given in Sect. 4 of this report should be detailed enough to allow a user to create an input stream for each code. However, there are several advantages to running this system interactively. First of all, the user has much more control over the sequence of calculations. He can decide, on the basis of results obtained in one terminal session, which responses to study or which time ranges are important. Secondly, the interactive method is a more efficient use of the user's time. Fewer input errors will occur because a prompting message precedes each required input. The number of re-runs due to inputting the wrong data file is greatly reduced because the codes usually allow the user to verify that he has input the correct files before proceeding. Finally, there is probably little cost incentive for running in a batch mode. In a recent use of this system, about 80% of the computer charges were incurred in running RELAP and the other IBM-based codes, while the interactive PDP-10 programs accounted for about 20%. Since the cost for a given number of calculations is about the same on the two computers, no significant savings would be realized in going to a batch mode.

This software package has been designed around an uncertainty analysis of systems which are modeled with the RELAP code, but could be converted to handle a different base code. Those PDP-10 programs (or JCL for IBM programs) which manipulate the input to or output from RELAP would have to be replaced, but the programs which calculate the sensitivity coefficients and perform the uncertainty analysis would need only minor modifications. The files which contain all the intermediate and final results would also not require many changes. A user could, in principle, adjust this package to fit his special computational needs.

Section 2 of this report describes the nonlinear uncertainty analysis methodology used in these codes. To illustrate the process and describe the various codes, the analysis of a transient film boiling heat transfer experiment is outlined in Sect. 3. In Sect. 4, a detailed description of the input to each code is given. Section 5 contains suggestions for extended uses of the system and comments on the changes necessary to convert the system to a different base code. For the user who is interested in changing the system to meet his own needs, the formats of the files containing the intermediate and final results are given in Appendix A.

2. ANALYSIS METHOD

2.1 Response Surface Method

The method employed in this code package is a nonlinear uncertainty analysis methodology developed by Ronen² and updated by Wacholder.⁷ It is basically a response surface method in which various calculations are performed with perturbed input parameters to estimate the functional dependence of the calculated responses on the input parameters. First and second order sensitivity coefficients are computed for various system responses with respect to the input parameters. These coefficients and a set of parameter uncertainties are then used to calculate the expected value and standard deviation of each response.

To begin with, we use the following second order expansion of a response R about the reference parameters.

$$R(\bar{\sigma}) \approx R(\bar{\sigma}_0) + \sum_{i=1}^N \frac{\partial R}{\partial \sigma_i} \Delta \sigma_i + \frac{1}{2} \sum_{i,j=1}^N \frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} \Delta \sigma_i \Delta \sigma_j \quad (1)$$

$\bar{\sigma}_0$ is the set of N reference parameter values and $\bar{\sigma}$ is the set of parameters when perturbed by the amount $\Delta \sigma_i$. The derivatives $\partial R / \partial \sigma_i$ and $\partial^2 R / \partial \sigma_i \partial \sigma_j$, evaluated at $\bar{\sigma}_0$, are the required first and second order sensitivity coefficients, with the second order coefficient termed a "mixed derivative" when $i \neq j$.

The coefficients $\partial R / \partial \sigma_i$ and $\partial^2 R / \partial \sigma_i^2$ can be calculated by performing two RELAP runs for each input parameter. In the first run the parameter σ_i is perturbed by $+\Delta \sigma_i$ and the resulting response is denoted by R_+ . In the second run the perturbation is $-\Delta \sigma_i$ and the response is denoted by R_- . If the reference response is R_0 , we obtain

$$\frac{\partial R}{\partial \sigma_i} = \frac{R_+ - R_-}{2\Delta \sigma_i} + O(\Delta \sigma_i^2), \quad (2)$$

$$\frac{\partial^2 R}{\partial \sigma_i^2} = \frac{R_+ + R_- - 2R_0}{\Delta \sigma_i^2} + O(\Delta \sigma_i^2). \quad (3)$$

For the calculation of the mixed derivatives, in which two parameters are involved, one should make four RELAP runs for each pair of parameters. The perturbations for the four cases are $(+\Delta \sigma_i, +\Delta \sigma_j)$, $(+\Delta \sigma_i, -\Delta \sigma_j)$, $(-\Delta \sigma_i, +\Delta \sigma_j)$ and $(-\Delta \sigma_i, -\Delta \sigma_j)$, and the responses are denoted by R_{++} , R_{+-} , R_{-+} and R_{--} . The mixed derivative is then calculated as

$$\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} = \frac{R_{++} - R_{+-} - R_{-+} + R_{--}}{4\Delta\sigma_i \Delta\sigma_j} + O(\Delta\sigma_i \Delta\sigma_j). \quad (4)$$

This procedure requires $2N(N-1)$ computer runs for all mixed derivatives, with a total of $2N^2+1$ for all first and second order derivatives.

To reduce the large number of computer runs required to calculate the mixed derivatives, we can approximate $\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j}$ by using only the $(+\Delta\sigma_i, +\Delta\sigma_j)$ perturbed case. The result is

$$\left(\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} \right)_{++} = \frac{R_{++} - R_{+,i} - R_{+,j} + R_0}{\Delta\sigma_i \Delta\sigma_j} \quad (5)$$

where $R_{+,i}$ is the response due to a change of $+\Delta\sigma_i$ in parameter i . Similarly, if we used only the $(-\Delta\sigma_i, -\Delta\sigma_j)$ case, we would obtain

$$\left(\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} \right)_{--} = \frac{R_{--} - R_{-,i} - R_{-,j} + R_0}{\Delta\sigma_i \Delta\sigma_j} \quad (6)$$

Using either one of these cases would reduce the number of computer runs needed to calculate all of the mixed derivatives to $N(N-1)/2$, and the total number of computer runs to $N(N+3)/2+1$.

A somewhat better approximation to the mixed derivative, but still less expensive than using all four perturbed cases, is to use both the $(+\Delta\sigma_i, +\Delta\sigma_j)$ and $(-\Delta\sigma_i, -\Delta\sigma_j)$ cases. The resulting derivative is

$$\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} = \frac{1}{2} \left[\left(\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} \right)_{++} + \left(\frac{\partial^2 R}{\partial \sigma_i \partial \sigma_j} \right)_{--} \right] \quad (7)$$

This procedure requires $N(N-1)$ runs for all of the mixed derivatives and N^2+N+1 for all derivatives.

All of the above approximations to the mixed derivatives are allowed in the code package. Which one is used depends on the number of mixed derivatives that need to be calculated, the cost of each one (and the size of the user's budget) and the size of them relative to the other coefficients.

For the calculation of sensitivity coefficients in Eqs. 2-7 above, the perturbation $\Delta\sigma_i$ is a time-dependent quantity. Although it could, in principle, take on any arbitrary value, the usual practice is to specify the perturbation as constant over fairly long time ranges, with the size of the perturbation determined by the supplied value of the standard deviation of the parameter. We have commonly used a perturbation of from one-fifth to one-half the size of the standard deviation. This fairly small value was chosen to try and keep the system in a range of linear behavior. If the standard deviation was specified as a percentage of the reference parameter, the perturbations were also fractions of the parameter values.

In the nonlinear uncertainty analysis the response R is once again represented as a Taylor series expansion about the reference parameter values $\bar{\sigma}_0$. To second order this expansion is

$$R = R(\bar{\sigma}_0) + \sum_{i=1}^N S_i \delta\sigma_i + \frac{1}{2} \sum_{i,j=1}^N H_{ij} \delta\sigma_i \delta\sigma_j, \quad (8)$$

where S_i and H_{ij} are the first and second order sensitivity coefficients calculated above and $\delta\sigma_i$ is the variation of parameter i . The expected value of R is then

$$\langle R \rangle = R_0 + \frac{1}{2} \sum_{i=1}^N H_{ii} v_i, \quad (9)$$

where v_i is the variance (square of the standard deviation) of parameter i . The covariance between any two responses R^a and R^b is defined as

$$\begin{aligned} \text{Cov}(R^a, R^b) &= \langle (R^a - \langle R^a \rangle) (R^b - \langle R^b \rangle) \rangle \\ &= \sum_{i=1}^N S_i^a S_i^b v_i + \frac{1}{2} \sum_{i=1}^N H_{ii}^a H_{ii}^b v_i^2 \\ &\quad + \sum_{i=1}^N \sum_{j=i+1}^N H_{ij}^a H_{ij}^b v_i v_j \end{aligned} \quad (10)$$

The variance of a single response is a special case of the covariance in which $a=b$.

$$\begin{aligned} \text{Var}(R^a) &= \text{Cov}(R^a, R^a) = \langle (R^a - \langle R^a \rangle)^2 \rangle \\ &= \sum_{i=1}^N S_i^2 v_i + \frac{1}{2} \sum_{i=1}^N H_{ii}^2 v_i^2 + \sum_{i=1}^N \sum_{j=i+1}^N H_{ij}^2 v_i v_j \end{aligned} \quad (11)$$

The standard deviation of the response is then just the square root of the variance.

A related covariance may be obtained for the nominal values of two responses, R_0^a and R_0^b .

$$\begin{aligned} \text{Cov}(R_0^a, R_0^b) &= \langle (R^a - R_0^a) (R^b - R_0^b) \rangle \\ &= \text{Cov}(R^a, R^b) + \frac{1}{4} \left(\sum_{i=1}^N H_{ii}^a v_i \right) \left(\sum_{i=1}^N H_{ii}^b v_i \right) \end{aligned} \quad (12)$$

Again, the variance of a single nominal response is a special case of the covariance in which $a=b$.

$$\begin{aligned} \text{Var}(R_0^a) &= \text{Cov}(R_0^a, R_0^a) = \text{Var}(R^a) + \frac{1}{4} \left(\sum_{i=1}^N H_{ii}^a v_i \right)^2 \\ &= \text{Var}(R^a) + \frac{1}{4} \sum_{i=1}^N H_{ii}^2 v_i^2 + \frac{1}{2} \sum_{i=1}^N \sum_{j=i+1}^N H_{ii} H_{jj} v_i v_j \end{aligned} \quad (13)$$

A final quantity of interest which is calculated by this code package is the correlation coefficient between any two responses. Denoting the standard deviation of a response as $\text{SD}(R)$, this coefficient is given by

$$\rho(R^a, R^b) = \frac{\text{Cov}(R^a, R^b)}{\text{SD}(R^a) \cdot \text{SD}(R^b)} \quad (14)$$

By definition it assumes a value in the range $[-1, 1]$. A value near 1 indicates a strong positive correlation between the two responses. A set of parameter perturbations resulting in a change in R^a will produce a similar change (in the same direction) in R^b . Conversely, a coefficient near -1 indicates a strong negative correlation, with responses R^a and R^b moving in opposite directions to the same perturbation. A value of the coefficient near 0 indicates no correlation between the two responses.

2.2 Use of Sensitivity Coefficients in Other Uncertainty Analysis Methods

The uncertainty analysis methodology described above generally assumes that a parameter is perturbed uniformly in time. The derivatives at a particular time step are then calculated as a difference between a perturbed and a reference response value divided by the amount of the uniform perturbation. Because these perturbed responses

are computed from compounded variations of the parameters in time, the calculated derivatives cannot properly represent the true values of the coefficients in the Taylor's expansion (Eq. 1). If this and other problems associated with the response surface method are judged by the user to be serious enough to warrant a more exact approach, then the methods proposed by Barhen² might be used.

These methods basically require the sensitivities of a response at time step v due to impulse perturbations of the parameters at all previous time steps. If we denote by $\partial R^v / \partial \sigma_i^\mu$ the sensitivity of a response R at time step v due to a perturbation of parameter i at time step μ , then we can calculate the first order sensitivities as

$$\begin{aligned} \frac{\partial R^v}{\partial \sigma_i^\mu} &= 0 && \text{for } \mu > v \\ &= \frac{R_+^{v\mu} - R_-^{v\mu}}{2 \Delta \sigma_i^\mu} + O(\Delta \sigma_i^{\mu^2}) && \text{for } \mu \leq v \end{aligned} \quad (15)$$

As before, $R_+^{v\mu}$ and $R_-^{v\mu}$ are the values of the perturbed response at time step v when the perturbation at time step μ was positive and negative, respectively. These equations would replace Eq. 2 in Sect. 2.1, and similar equations would be used to calculate the higher order derivatives. An option in the codes allows for the calculation of sensitivity coefficients in this manner.

The generation of all of the first and second (but not mixed) derivatives for a response at time step v would require $2(vN+1)$ computer runs. The cost of the RELAP runs for these more exact methods would, of course, be greater by a factor of v than the runs required for the response surface method. The result would be v sets of sensitivity coefficients from which the user could extract all of the derivatives for time steps $\leq v$.

3. ANALYSIS PROCEDURES

An uncertainty analysis of a system modeled with the RELAP code is performed in two stages. In the first stage the experimental group measures various system parameters, under both static and transient conditions, and places them in a data file. For the THTF experiments this data file is called an Engineering Units Data Tape.¹¹ The parameters are the boundary or initial conditions of the system. A RELAP model of the system is devised and the parameters from the Engineering Units Data Tape are converted to RELAP variables and stored on an input data file recognized by RELAP. For the local version of the code this is called an OXREP¹² tape. Uncertainties (1σ standard deviations) are assigned to each parameter. At the end of the first stage the user can run the RELAP code to predict various system responses.

In the second stage of the analysis, the software package described herein is used to compute first and second order sensitivity coefficients for various system responses with respect to the input parameters. These coefficients and the parameter uncertainties supplied by the experimental group are used in the nonlinear uncertainty analysis methodology described in Sect. 2. The expected value and standard deviation of each response are calculated, as well as the covariance between a pair of responses.

The purpose of this section is to describe the sequence of calculations comprising the second stage of the analysis. Each component of this software package will be introduced as it appears in the sequence. In order to illustrate the calculations, an analysis for the PWR-BDHT program will be described. Figure 1 is a flow chart of that analysis. With the exception of the SENSORINC⁹ code, all steps to the right of the dashed line are performed by this software package.

The THTF facility in the PWR-BDHT program is a pressurized test section containing a bundle of electrically heated, water-cooled rods with conditions, both static and transient, simulating those in a nuclear fuel bundle. Single- or double-ended pipe breaks may be approximated. In this analysis, the important parameters were the flow rate and enthalpy at the inlet to the test section, the pressure at the outlet from the test section and the responses of a number of thermocouples located in the heater rods. Using the thermocouple responses and a model of a heater rod, the ORINC¹⁰ code calculated the 20 rod surface heat fluxes used as boundary conditions in the RELAP model of the test facility. Along with the three inlet and outlet parameters, the arithmetic average of the 20 heat fluxes was used as the fourth system parameter. The six basic steps in the calculation of response uncertainties are as follows.

3.1 Parameter Perturbation and RELAP Runs

In order to calculate the sensitivity coefficients needed in the uncertainty analysis, a number of RELAP runs must be made with perturbed parameters. These parameters are located in one of two places. The conditions at the inlet or outlet, or possibly both, are contained in the model input to RELAP. This card-image file, whose main purpose is the specification of system geometry and material properties, has "fill tables" in which are stored the time-dependent mass flow rate and enthalpy. These are specialized sections of the RELAP input set up for the local version and given section designators of 81xxxxx or 83xxxxx. With the reference version of the model input file stored on a PDP-10 file, the interactive program RLPCRE is used to make perturbations to the parameters in these sections and output an updated file. In the flow chart of Fig. 1, the parameters perturbed by this code, inlet mass flow rate and enthalpy, are identified as JW27 and JH27, respectively.

All of the other boundary conditions necessary for running RELAP are stored on the RELAP Input Tape. For the local version of RELAP this is also called an OXREPT¹² tape. Perturbations of parameters on this input file are accomplished with the REDSEL⁵ code, written for ETD and running on the IBM 370/3033. The outlet pressure and surface heat fluxes, identified as AP26 and FR01-FR20, are the parameters that were perturbed in this analysis. The JCL and input data necessary to run REDSEL are contained in the PDP-10 file shown in Fig. 2. This file also contains the JCL required to run RLPSFLUX. The reference model input file, or perturbed file from RLPCRE, and the output from REDSEL are the input files to RLPSFLUX.

3.2 Accessing the RELAP Output Files

Each RELAP run creates an output file which contains the time-dependent values of every response (and parameter) defined in the model. The REDPLOT code, written for ETD and running on the IBM 370/3033, is used to access the data on these files. The JCL for running REDPLOT is contained in the PDP-10 file shown in Fig. 3.

The output from REDPLOT are BCD-formatted tables of RELAP variables versus time on a PDP-10 file. Every response of interest would be included in the tables and, if not the reference case, the parameter that had been perturbed. The user is responsible for specifying the four-character identifiers of the parameters and responses he desires. The rules he must obey in modifying the file shown in Fig. 3 for his purposes are described in Sect. 4.

```

//UIDRELAP JOB (charge #,IO4),'name field',TIME=2
/*
/* BEGINNING OF REDSEL
/* INPUT OXREPT TAPE TO BE MODIFIED IS IN FT03F001 IN STEP GO
/*
//SETUP EXEC PGM=RULOAD,REGION=50K
//STEPLIB DD DSN=A.L.TEA15231.RULOAD,DISP=SHR,UNIT=3330V,
// VOL=SER=VCRH12
//FT05F001 DD *
RE 20.0
AP -1
/*
//FT06F001 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=1100)
//FT08F001 DD DSN=*&RUDATA,UNIT=SYSDA,DISP=(,PASS),
// SPACE=(TRK,(5,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//FT09F001 DD DSN=*&RUREPL,UNIT=SYSDA,DISP=(,PASS),
// SPACE=(TRK,(5,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//COMPL EXEC FORTHCLG,CLSIZE=270K,PARM.LKED='LIST,NCAL',LKSIZE=270K,
// GOSIZE=500K
//PORT.SYSIN DD *
      SUBROUTINE SMOOTH(Y,X,I,J)
      REAL*4 Y(100,800)
      REAL*4 X(800)
      INTEGER*4 I,J
C      Y(L,XXX) NOW CONTAINS DATA ELEMENT L AT TIME STEP XXX.
C      ARRAY X CONTAINS TRANSIENT TIMES.
C      I IS THE NUMBER OF DATA ELEMENTS SELECTED AND J IS THE
C      NUMBER OF TIME STEPS.
C
C
C      INSERT HERE THE CODING TO CHANGE THE PARAMETERS
C      STORED IN THE Y ARRAY
C
C      RETURN
      END
/*
//LKED.OLDOBJ DD DSN=A.L.TEX15231.REDSEL,DISP=SHR,UNIT=3330V,
// VOL=SER=VCRH12
//LKED.SYSIN DD DSN=*&RUREPL,UNIT=SYSDA,DISP=(OLD,DELETE)
//GO.FT01F001 DD DSN=*&INTERMA,UNIT=SYSDA,DISP=(,PASS),
// SPACE=(TRK,(100,100),RLSE),DCB=(RECFM=UT,BLKSIZE=24001)
//GO.FT02F001 DD DSN=*&INTERMB,UNIT=SYSDA,DISP=(,PASS),
// SPACE=(TRK,(100,100),RLSE),DCB=(RECFM=UT,BLKSIZE=24001)
//GO.FT03F001 DD      Input OXREPT tape being updated by REDSEL
//GO.FT04F001 DD UNIT=SYSDA,DISP=(,PASS),SPACE=(TRK,(100,100),RLSE),
// DCB=(RECFM=UT,BLKSIZE=24001),DSN=*&NEWOX
//GO.FT05F001 DD DSN=*&RUDATA,UNIT=SYSDA,DISP=(OLD,DELETE)

```

Figure 2. PDP-10 File for Running REDSEL and RLPSFLUX

```

/*
/*  RUN RLPSFLUX
/*  THE MODIFIED (OR REFERENCE) OXREPT TAPE IS ON FT02F001
/*  THE CARD-IMAGE MODEL INPUT IS ON THE PDP-10 FILE REL.TMP
/*  THE OUTPUT FILE IS ON FT04F001
/*
/*RLP EXEC PGM=RLPSFLUX,REGION=1100K,PARM='10,1600'
/*STEPLIB DD DSN=A.L.TEA15231.RLP4M5U2,DISP=SHR,UNIT=3330V,
//          VOL=SER=VCRH12
/*FT02F001 DD UNIT=SYSBA,DISP=(OLD,DELETE),DSN=*&NEWOX
/*FT03F001 DD DUMMY
/*FT04F001 DD DCB=(RECFM=UT,BLKSIZE=24001),
//          Specify rest of JCL for output file
/*FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1064)
/*FT15F001 DD UNIT=3330V,VOL=SER=VCRH12,DISP=SHR,
//  DSN=A.L.TEA15231.WATRPROP
/*FT16F001 DD UNIT=SYSBA,SPACE=(TRK,(5,5))
/*FT05F001 DD *
=REL.TMP
//

```

Figure 2. Continued

```

//UIDPWC JOB (charge #,I02),'SAVEp,pn',TIME=(,20)
/*JOBPARM CARDS=10000
/*ROUTE PUNCH REMOTE#5
/**
/** START REDPLOT STEP
/** RELAP OUTPUT FILE TO BE ACCESSED IS ON FT01F001
/** OUTPUT FROM REDPLOT WILL BE A FILE ON USER'S PDP-10 DISK SPACE
/** WITH THE NAME XXXXXX.PCH, WHERE XXXXXX IS THE FIRST SIX
/** CHARACTERS OF THE JOBNAME
/**
//S1 EXEC PGM=GENPLT,REGION=400K
//STEPLIB DD DSN=A.TEALF231.SUPERPL3,DISP=SHR,UNIT=3330V
//FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1064)
//FT07F001 DD SYSOUT=B,DCB=(RECFM=F,LRECL=80)
//FT20F001 DD UNIT=SYSDA,SPACE=(32000,(500,4)),
// DCB=(RECFM=FT,BLKSIZE=32000)
//FT40F001 DD DUMMY,DCB=(RECFM=VBS,BLKSIZE=6515)
//FT01F001 DD RELAP output file to be accessed
//FT33F001 DD UNIT=3330V,VOL=SER=VCRH11,DISP=SHR,
// DSN=A.D.EZD15231.UPDRAF1
//COMPOUT DD UNIT=SYSDA,DISP=(,PASS),DSN=*&COMP,SPACE=(TRK,(200,50))
//FT05F001 DD *
0.0 20.0
RELMAC
1 10
JW03
JW04
JW05
JW06
JW07
JW08
JW09
JW10
JW11
JW01
PUNCH 0.05 0.0 20.0 10
(1X,0PF6.2,1P5E13.6)
PUNCH 0.05 0.0 20.0 1 2 3 4 5
(1X,0PF6.2,1P5E13.6)
PUNCH 0.05 0.0 20.0 6 7 8 9
(1X,0PF6.2,1P5E13.6)
/*
//

```

Figure 3. PDP-10 File for Running REDPLOT

3.3 Conversion of RELAP Responses to Binary Files

The BCD-formatted files created by REDPLOT are cumbersome to work with and are an inefficient use of PDP-10 disk storage and computing time. For this reason, all storage of RELAP responses and subsequent use of the data involves the binary (unformatted) file whose contents are described in Sect. A.1. The interactive program RLPRLP is used to create this binary file from the BCD-formatted file created by REDPLOT. A file is created for each RELAP run. In addition to reading and storing the perturbed (or reference) responses, this program also requests the user to specify the amount of perturbation that was used in the RELAP run.

3.4 Calculation of Sensitivity Coefficients

Once RELAP response files have been created for the reference case and for the perturbed cases, the RLPSEN interactive program is used to calculate the sensitivity coefficients and mixed derivatives as per the formulae presented in Sect. 2. The output from this program is a binary file of the first and second order sensitivity coefficients if only a single parameter is perturbed, or of the mixed derivatives if two parameters are perturbed. The contents of these files are described in Sects. A.2 and A.3.

3.5 Uncertainty Analysis

The interactive program UNCOVER performs the uncertainty analysis presented in Sect. 2. It requires as input the binary files of sensitivity coefficients and mixed derivatives for all parameters as created in the previous step. Also required are the uncertainties, supplied by the experimental group, of every parameter in the calculation. These are the variances (square of the standard deviations). (Note that in Fig. 1 the SENSORINC code is not part of this software package, but is a separate code that was used to calculate the uncertainties in the surface heat fluxes from the uncertainties in the thermocouple readings.) They may be input either from the terminal or from a separate file.

The output from UNCOVER is a binary file containing the nominal and expected values of each response and its standard deviation. If response covariances are requested, the covariances and correlation coefficients of pairs of responses are output to a separate binary file. The description of these files is given in Sects. A.4 and A.5.

3.6 Producing Edits of All Binary Files

The final interactive program in this package, EDITOR, produces readable edits of all of the binary files created in Steps 3.3-3.5. As with most of the codes in the package, EDITOR has been made quite general in order to provide the user with flexibility in producing edits.

4. INPUT SPECIFICATIONS

This section describes in detail the input to the interactive programs and to the codes running on the IBM 370/3033. The PDP-10 files which contain the JCL and input for running the latter are reproduced in Figs. 2 and 3, while the source listings for the former are given in the enclosed microfiche. The input descriptions follow the procedure outlined in Sect. 3.

With the exception of the four-character identifiers for the parameters and responses, all data entered into the interactive programs is entered free-form. This means that the user need not worry about the format of the data he is entering from the terminal, as long as there is at least one blank space or comma between entries and the correct number of entries per line. Most of the prompting messages give the number of type of entries required, with I indicating an integer, F a real number and A a literal field. Thus, the response to a prompting message specifying the format (2I,3F,I/2A4) would be two lines of data. On the first line there would be six entries, the first two integers, the next three real numbers and the last one an integer. On the second line the first eight spaces would have two Hollerith fields, each four characters long. A carriage return entered before all of the required entries have been typed will result in the remaining entries being set to zero if they are numeric data, or to blank fields if they are literal data.

Most of the files, whether input or output, used by the interactive programs have a default filename which may be changed by the user. When such a file is opened, a message of the following form will be issued.

Unit=n :filename/ACCESS=SEQINOU/MODE=ASCII or BINARY

Enter new file specs. End with an \$(ALT)

*

Following the asterisk, the user should type in the name of the file he wishes to use, followed by an altmode (ALT or ESC). If the name should not be changed, the user should just type an altmode. If any of the interactive programs are being run in a batch mode by setting up an input stream according to the instructions given in this section, care must be taken when opening the files. Since the filenames are terminated with an altmode, which generates an automatic carriage return, a carriage return should not be typed before entering the next line of data. For example, in the RLPCRE program the first three required entries are the filenames for input and output and the two-digit

indicator of the section to be changed. If the input filename is to be changed to RELAP.IN, the output filename is to be left at the default of REL.TMP and Sect. 81 is to be accessed, these should all be specified on the following first line of data.

RELAP.IN\$\$81

The remaining data would start on the next line.

4.1 Parameter Perturbation and RELAP Runs

The interactive program RLPCRE is used to make changes to the input parameters which exist in the fill tables of the RELAP card-image model input. The default filenames of the reference input and updated output files are RELAL.REF and REL.TMP. Since REL.TMP is the default filename of the card-image input file used in the job step that runs RLPSFLUX, (see Fig. 2, line 78), this name is usually not changed. In order for RLPCRE to correctly read the fill tables in the input file, the user must ensure that each line in the fill table, except possibly the last one, has the same number of time steps. This may require some reformatting of the input file.

Changes to the mass flow rate or enthalpy in a fill table may be accomplished by either entering the amount of perturbation from the terminal or by reading a separate file that contains the new values of the parameter. If the perturbation amounts are being supplied at the terminal, these are given as either absolute amounts or as relative changes (in percents), held constant over specified time ranges. Any number of time ranges may be used; but if the number becomes too large, it may be easier to specify the new parameter in the separate PDP-10 file with the default filename ADJ.PAR. This should be a binary (unformatted) file with the following contents.

Record 1 : NEWP,NEWT

NEWP = # of new parameters

NEWT = Number of time steps at which the new parameters are evaluated

Record 2 : (A(I),I=1,NEWP)

A(I) = Four-character identifier of new parameter I

Record 3 : T,(X(I),I=1,NEWP)

X(I) = New value of parameter I at time T

Repeat record 3 NEWT times.

Only one file of new parameters may be used in a single run of RLPCRE. The user must make sure that all of the new parameters he needs are on the same file.

Any combination of parameter perturbations can be accomplished by making multiple passes through the input and output files. After the first pass through the original input file, with the perturbed parameters placed on the output file, the unit numbers of the input and output files will be switched. The newly updated output file would be used as the input file for the second pass and the original input file would be written over with the perturbed amounts from the second pass. This switching will continue until all of the desired changes have been made. The unit number of the current output file will be printed at the terminal before any switching occurs. It is up to the user to ensure that a permanent reference version of the input does not get destroyed by multiple passes.

Input to RLPCRE

1. Open input file. Default filename is RELAL.REF.
2. Open output file. Default filename is REL.TMP.
3. DC (I2)

DC = Two-digit section indicator. Must be 81, 83 or 99.

= 81 or 83 to make changes in one of these sections. Go to record #4.

= 99 to terminate this pass through the input file. Go to record #9.

4. LINE (I)

LINE = Number of time steps per line in section identified by DC.

5. ISUB (I)

ISUB < 0 to stop processing this section. Go to record #3.

= 0 to accept perturbation information from the terminal.
Go to record #6.

> 0 to read new parameters from a separate file. Go to
record #7.

6. MORA,MSEN,T1,T2,ADJ (2I,3F)

MORA = 0 for addition, 1 for multiplication.

MSEN = 0 to change the mass flow rate, 1 to change the enthalpy.

T1,T2 = First and last times at which to apply the perturbation.

ADJ = Constant addition if MORA=0, constant percentage change if
MORA=1.

Go to record #5 to specify the next perturbation.

7. If it has not already been used in this RLPCRE run, open the file which contains the perturbed parameters. Default filename is ADJ.PAR.

8. NEN,(NORD(J),J=1,NEN),(NORDC(J),J=1,NEN) (5I)

NEN = Number of parameters to be changed with this file. Since only
the mass flow rate and enthalpy are in the fill table, NEN
must be either 1 or 2.

NORD(J) = 0 if the Jth parameter to be changed is the mass flow
rate, 1 if it is the enthalpy.

NORDC(J) = Order number in the file of the Jth parameter to be
changed.

Go to record #5 to specify the next perturbation.

9. NSW (I)

NSW = 1 to switch unit numbers and make another pass through the
files.

Go to record #3.

= anything else to terminate the job

To make changes to the parameters stored on the OXREPT tape, the REDSEL code is used. The PDP-10 file shown in Fig. 2 contains the JCL and input data for running both REDSEL and RLPSFLUX. To set up REDSEL for the perturbation of a particular parameter or set of parameters, one must know the four-character identifiers of the parameters and the length of the transient being studied. Line 21 in the file of Fig. 2 contains length of the transient (in seconds) formatted as (2HRE,F). The parameter identifiers are entered beginning in line 22 with the format (A2,17I4/(18I4)). Each line, or lines, of input specifies parameters of the same type in the A2 section of the format. In the example shown in Fig. 2, AP denotes the average pressure in a volume. The volume or junction numbers to be retrieved are read in the integer portions of the format, with a zero entry signaling the end of the list. If the first numeric entry is negative, the only volume or junction retrieved will be the absolute value of the entered number. This is the case in the example shown, in which only the outlet pressure, AP02, was desired. If the first numeric entry is zero, all volumes or junctions associated with the specified parameter type will be retrieved.

To make changes to the parameters specified on line 22, the user must supply the appropriate FORTRAN coding in the SMOOTH subroutine. The parameters will be located in array Y(I,J), with I the number of parameters selected and J the number of time steps. The array X contains the transient times. The reference OXREPT tape should be identified in the DD card for FT03F001 at line 56.

The RLPSFLUX code is set up to run in a job step immediately following the REDSEL execution. If no changes are being made to the parameters on the OXREPT tape, then any RELAP variable may be specified in the REDSEL input. No coding would be supplied in the SMOOTH subroutine to perturb the parameter, so that the OXREPT tape passed to the RLPSFLUX step would be the reference. The command at line 78 of Fig. 2 specifies the filename of either the reference model input or the updated file produced by RLPCRE. The output data set should be identified in the DD card for FT04F001 at line 72.

4.2 Accessing the RELAP Output Files

In this software package the RELAP output file is assumed to be stored on some device, such as disk or magnetic tape, accessible to the IBM 370/3033. The REDPLOT code, whose JCL and input are located on a PDP-10 file such as shown in Fig. 3, is used to retrieve selected portions of this output and transfer it to a PDP-10 file on the user's disk area. The output of REDPLOT is a BCD-formatted file containing tables of RELAP variables versus time.

To run REDPLOT, one must first specify the location of the RELAP output file in the DD card for FT01F001 on line 18 of Fig. 3. Then, the time range of interest should be entered on line 23 in free-form format. On line 25, the second number, again in free-form, should be the number of parameters and responses to be accessed. This should be immediately followed by the four-character identifiers, one per line in the first four spaces of each line, of the required parameters and responses.

Following the last identifier are the PUNCH and format statements for outputting the desired RELAP variables. Each PUNCH statement creates a table of from one to five of the variables according to the format given on the next line. The PUNCH statement has the form

PUNCH , DT, T1, T2, order numbers in variables list

using the format (8HPUNCH ,3F,5I). T1 and T2 are the beginning and end of the time range to be output and DT is the time step size. The format statement on the line after the PUNCH statement should have at least one more field than the number of variables being output because the time is punched as the first field on every line. Any real number format may be used. The parameters and responses must be output to the PDP-10 file in a particular manner in order to be correctly read by the next program in the sequence. If this is a perturbed parameter case, the parameter only must be output by the first PUNCH statement. If this is a two-parameter perturbation, the second parameter should be output by the second PUNCH statement. If any parameter is a function of other parameters, such as the average heat flux mentioned in Sect. 3, each of its components must be output in a separate PUNCH statement. Once the parameters have been output, the responses may be output in any order with as many PUNCH statements as needed, as long as no more than five responses are specified in any PUNCH statement. The format statements must be the same for all parameters. Likewise, the responses must all be output with the same format, although this format may be different than that used for the parameters. If a reference case is being accessed, only the responses should be output to the BCD file. In the example shown in Fig. 3, JW01 is the parameter and JW03-JW11 are the nine responses. Note that the parameter need not be the first variable in the list; it only needs to be the first one output by a PUNCH statement.

It is imperative that the user keep track of the order in which the parameters and responses were output because the identifiers are not output along with the tables. These tables will be read in the next program of the sequence and the user will be required to assign an identifier to each column of the table. The output file from the REDPLOT code will be stored on the user's disk area as defined by his project, programmer number given on the job card. This should be entered in the field labeled #####,### immediately following the word "SAVE" on line 1 of Fig. 3. The file created by REDPLOT will have the filename xxxxxx.PCH, where xxxxxx is the first six characters of the jobname.

4.3 Conversion of RELAP Responses to Binary Files

For each BCD-formatted file of RELAP responses created by the REDPLOT code, the interactive program RLPRLP produces a binary (unformatted) file, the use of which saves both computing time and disk space. Any number of the BCD files may be converted to binary files in a single interactive session. Associated with each case is a parameter number which identifies it as a reference or perturbed case.

As described in the previous section, the BCD file created from the reference case contains only the reference values of the responses; the parameters are not included. In the perturbed cases, the perturbed values of the parameters are output before the perturbed values of the responses. Since the reference values of the parameters are required in the calculation of the relative sensitivity coefficients, the perturbation amounts are input to RLPRLP and the reference parameters are created from the perturbed parameters. The reference parameter values are then carried along in the binary response files of the perturbed cases. An alternate, and easier, method would have been to output the reference parameters to the same file as the reference responses. This would have eliminated the need to specify (again) the perturbation amounts, but would have reduced the generality of the system. If, after creation of the reference file, the user decided to add more parameters to his study, the reference file would have to be recreated. The current system allows for more generality and more straightforward programming.

In the input described below, it should be noted that the variable IP is intended to order the parameters of interest and convey some information about how many parameters were perturbed and how they were perturbed. The values used, however, do not have to be in any specified order or be in any way dependent on the type of parameter. The rules given below for choosing the allowed range of IP values must be conformed to, but within that range all values are arbitrary. The IP values for various parameters need not even be unique.

Input to RLPRLP

1. IC (I)

IC = 1 to continue with another case

= anything else to terminate the job

2. Open input BCD file. Default filename is RELAP.PCH.

3. Open output binary file. Default filename is BINARY.DAT.

4. IFMT (I)

IFMT = 0 to keep the current format for reading the parameters.

Default format is (7X,1PE13.6). This format skips over the field containing the time. Go to record #6.

= anything else to read in a new format. Go to record #5.

5. FMTPAR (80A1)

FMTPAR = 80-character format, including outer parentheses, for reading parameters from input file.

6. IFMT (I)

IFMT = 0 to keep the current format for reading the responses.

Default format is (7X,1P5E13.6). Go to record #8.

= anything else to read in a new format. Go to record #7.

7. FMTRES (80A1)

FMTRES = 80-character format for reading responses from input file.

8. TITLE (80A1)

TITLE = 80-character title for this case.

9. NR,NT,DT,TZ,IP (2I,2F,I)

NR = Total number of responses on input file.

NT = Number of time steps.

DT = Time step size (seconds).

TZ = Beginning time (seconds).

IP = Parameter number; must conform to one of the following options

- i) IP = -1 for the reference case.
- ii) $0 < IP < 999$ for perturbation of the single parameter IP. Sensitivities are calculated at each time step using the perturbation supplied for that time step.
- iii) IP = 0 for the perturbation of two parameters. Sensitivities calculated as in option ii.

iv) $IP < -1$ for a single parameter $ABS(IP+1)$ changed at a single time T . Sensitivities are calculated at times greater than T using the perturbation at time T .

v) $IP = 999$ for two parameters changed at a single time T and sensitivities calculated as in option iv.

10. HPAR1,HPAR2 (2A4)

HPAR1,HPAR2 = Four-character identifiers of the perturbed parameters. If this is the reference case, leave both fields blank. If this is a single-parameter case, leave HPAR2 blank. For a two-parameter case, use both fields.

If this is the reference case, skip to record #16.

11. NCOM (I)

NCOM = Number of components of perturbed parameter. If < 1 , go to record #13.

12. IPSC (I)

IPSC = 0 if the perturbations are the same for all components.

= 1 if they are not.

13. ND (I)

ND = Number of time ranges needed to specify the perturbation.

14. MORA,BT,ET,DEL

MORA = 0 if the perturbation for a time range is a constant addition.

= 1 if it is a constant percentage change.

BT,ET = First and last times at which to apply this perturbation.

DEL = Constant addition if MORA = 0, constant percentage change if MORA = 1.

Repeat record #14 for each of the ND time ranges.

15. FRAC (F)

FRAC = Fraction of total parameter coming from this component.

This is entered only if there is more than one component.

Repeat the sequence of records 13-15 for each of the NCOM components. If NCOM > 1 and IPSC = 0, only record #15 should be entered for components past the first. Repeat the entire sequence of records 11-15 for the second parameter if this is a two-parameter perturbation.

16. NRS,(HR(I),I=1,NRS) (I1,5A4)

NRS = Number of responses in this section of the input table.
Must be < 6.

HR(I) = Four-character identifier to be assigned to the Ith response in this section of the input table.

Repeat record #16 as many times as necessary to read all of the input file. Terminate with NRS = 0 and go to record #1 to start a new case.

4.4 Calculation of Sensitivity Coefficients

Given the reference and perturbed binary response files, the RLPSEN interactive code calculates the sensitivity coefficients and outputs them to a binary file. In principle, all of the data required by the equations given in Sect. 2 is contained in the response files, so that the user need only specify which responses should be included. However, it is possible for the reference values for a given parameter to be incorrect on the perturbed response files. This could occur if, when running REDPLOT to punch the perturbed parameter and responses, the wrong identifier was supplied for the parameter. It could also occur if the desired parameter was defined only in the fill tables, not in the RELAP output file, and could not be accessed with REDPLOT. In the latter case, the user would have to specify some other parameter in REDPLOT and go through the motions in RLPRLP of forming the reference parameter values.

In either case, a separate PDP-10 file would have to be created which contained the correct parameter values. This file would be read by RLPSEN and the user would supply, at the terminal, the amounts of the perturbation. This should be a binary file with the following contents.

Record 1: NEWP, NEWT = Number of new parameters, number of time steps.

Record 2: (A(I), I=1, NEWP) = Four-character identifiers of the new parameters.

Record 3: T, (X(I), I=1, NEWP)

X(I) = New value of parameter I at time T.

Repeat record #3 for each of the NEWT time steps.

The use of this file should be considered only a temporary solution. The user should see to it that his perturbed response file is modified to include the correct parameter values.

In addition to calculating the sensitivity coefficients, KLPSEN has several options for making changes to a file of coefficients. One of these options allows the user to update the absolute sensitivities if the reference value of the parameter should change. This would be the case for a parameter which is a function of others if the function definition of the parameter changed after the sensitivities were calculated. The absolute derivative is recalculated from the relative derivative by multiplying by the old reference parameter and dividing by the new. The new parameter values should be given in a binary file with the same contents as just described.

Another option allows one to produce a short-form file of sensitivity coefficients from the default long-form file. The long form of the file contains the reference and perturbed values of the response as well as the absolute and relative first and second order sensitivity coefficients. The short form of the file contains the minimum information needed for the uncertainty analysis - the reference value of the response and the absolute first and second order sensitivity coefficients - and requires only about one-third as much disk storage as the long form. Since one sensitivity coefficient file is created for each parameter and all must be available at the same time in the program that performs the uncertainty analysis, the space savings realized by using the short form of the files may be significant (and necessary) if there are quite a few parameters and a large number of responses.

For the options labeled "S", "B" and "F" below, the user may either specify a list of responses for which to calculate sensitivities or have the code use every response found on the reference binary response file. In either case, sensitivities will only be calculated if the response is found on every input file. If a response cannot be found on a particular file, a warning message is issued and the code proceeds to the next requested response. For the case in which a response list has been specified, the sensitivities are written to a scratch file as they are calculated and a counter, IRES, keeps track of the number of responses that were found on all of the files. After each response in the list has been processed, the total number that were found, IRES, is written out to record #3 of the output file (see Sects. A.2 and A.3), and the sensitivities on the scratch file are copied to the output file.

A somewhat different situation occurs, however, if every response on the reference file is to be processed. Since the output and scratch files can become quite large if there are many responses, the scratch file is not used. Sensitivities are written directly to the output file as they are calculated. This requires that the number of responses on the reference file be written to record #3 of the output file before any

calculations have been made. If a response on the reference file cannot be found on one of the perturbed files, the code will once again issue a warning message and go on to the next response. This leads to a situation in which the number of responses actually on the output file is less than what is specified in record #3. Since ample warning is given when this occurs, the user should be able to correct the problem.

Input to RLPSEN

1. HOPT (A1)

HOPT = Option selector.

- = "S" to calculate single-parameter sensitivities.
- = "B" to calculate first approximation (++) or (--) mixed derivatives.
- = "M" to calculate second approximation (++) and (--) mixed derivatives.
- = "F" to calculate mixed derivatives using all four perturbed cases.
- = "U" to update a sensitivity file.
- = "L" to convert a long-form sensitivity file to a short-form.
- = anything else to terminate the job.

Option "S"

2. IALL (I)

IALL = 0 if a list of responses will be given.

- = 1 to use all responses on the reference response file.
- = -1 to use all responses on the reference response file and also give no printout of the sensitivities.

3. Open reference response file. Default filename is RELAP.REF.

Open positively perturbed response file. Default filename is PLUS.DAT.

Open negatively perturbed response file. Default filename is MINUS.DAT.

Open output sensitivity file. Default filename is SENSIT.DAT.

4. IFILL (I)

IFILL = 1 if the parameter on the perturbed response files is incorrect.

= anything else if it is correct.

5. If parameter is incorrect (IFILL=1), open file which has the correct parameter. Default filename is CORECT.DAT.

6. IVER (I)

IVER = 1 if the title of the positively perturbed response file, which is printed at the terminal, is correct. This is the only case checking which is done. Go to record #7.

7. IVER (I)

Same function as IVER in record #6, but using the negatively perturbed response file.

If the parameter is correct (IFILL=1), go to record # 11.

8. NP1 (I)

NP1 = Order number of desired parameter in the array of new parameters on the file.

9. ND (I)

ND = Number of time ranges needed to specify the perturbation for the new parameter.

10. MORA,BT,ET,DEL (1,3F)

Same definitions as in input to RLPRLP, record #14.

Repeat record #10 for each of the ND time ranges.

If all responses are being processed (IAL=0), go to record #14.

11. LIST (I)

LIST = 0 to input a new list of response.

= 1 to keep the previous list. Go to record #14.

12. NRES (I)

NRES = Number of responses in new list.

13. (HRES(I), I=1, NRES) (20A4)

HRES(I) = Four-character identifier of the Ith response in the list.

14. MORE (I)

If no printout of sensitivities is desired (IALL=-1), go to record #1. If printouts are desired, first 10 time steps of each response will always be edited. To get succeeding 10 time step sections, enter MORE=1. Enter MORE=0 to go to the next response. After the last response, go to record #1.

Option "B"

2. IALL (I)

IALL = 0 if a list of responses will be given.

= 1 to use all responses on the reference response file.

= -1 to use all responses on the reference response file and also give no printout of the mixed derivatives.

3. Open reference response file. Default filename is RELAP.REF.

Open positive (or negative) perturbation case for first parameter. Default filename is PAR1.DAT.

Open positive (or negative) perturbation case for second parameter. Default filename is PAR2.DAT.

Open case with both parameters positively (or negatively) perturbed. Default filename is PARBTH.DAT.

Open mixed derivative output file. Default filename is MIXD.DAT.

4. IFILL (I)

IFILL = 1 if either or both of the parameters on the perturbed response files are incorrect.

= anything else if they are both correct.

5. If incorrect (IFILL=1), open file which contains correct parameters.
Default filename is CORECT.DAT.
6. IVER (I)
IVER = 1 if the title of the parameter 1 perturbed case is correct.
7. IVER (I)
IVER = 1 if the title of the parameter 2 perturbed case is correct.
8. IVER (I)
IVER = 1 if the title of the two-parameter perturbed case is correct.
If both of the parameters are correct, go to record #12.
9. NP1,NP2 (2I)
NP1 = Order number of parameter 1 in the array of new parameters.
Enter 0 if this parameter is correct on the response files.
NP2 = Order number of parameter 2 in the array of new parameters.
Enter 0 if this parameter is correct on the response files.
10. ND (I)
ND = Number of time ranges needed to specify the perturbation for the new parameter.
11. MORA,BT,ET,DEL (I,3F)
All definitions as in RLPRLP, record # 14.
Repeat record #11 for each of the ND time ranges. Give records 10 and 11 for parameter 1 first, then for parameter 2. For either parameter enter ND=0 and skip record #11 if the parameter is already correct.
If all responses are being processed (IALL=0), go to record #15.
12. LIST (I)
LIST = 0 to input a new list of responses.
= 1 to keep the previous list. Go to record # 15.
13. NRES (I)
NRES = Number of responses in new list

14. (HRES(I),I=1,NRES) (20A4)

HRES(I) = Four-character identifier of the Ith response.

15. MORE (I)

If no printout of mixed derivatives is desired (IALI=-1), go to record #1. If printouts are desired, first 10 time steps for each response will always be edited. To get succeeding 10 time step sections, enter MORE=1. Enter MORE=0 to go on to the next response. After the last response, go to record #1.

Option "M"

2. Open mixed derivative file with both parameters positively perturbed. Default filename is MIXP.DAT.

Open mixed derivative file with both parameters negatively perturbed. Default filename is MIXM.DAT.

Open output second approximation mixed derivative file. Default filename is MIX2D.DAT.

3. TITLE (80A1)

TITLE = 80-character title for the output file.

Go to record #1.

Option "F"

2. IALL (I)

IALL = 0 if a list of responses will be given.

= 1 to use all responses on the reference response file.

= -1 to use all responses on the reference response file and also give no printout of the mixed derivatives.

3. Open reference response file. Default filename is RELAP.REF.

Open case with parameter 1 perturbed positively, parameter 2 positively. Default filename is PP.DAT.

Open case with parameter 1 perturbed positively, parameter 2 negatively. Default filename is PM.DAT.

Open case with parameter 1 perturbed negatively, parameter 2 positively. Default filename is MP.DAT.

Open case with parameter 1 perturbed negatively, parameter 2 negatively. Default filename is MM.DAT.

Open output mixed derivative file. Default filename is MIXD.DAT.

4. IFILL (I)

IFILL = 1 if either or both of the parameters on the perturbed response files are incorrect.

= anything else if they are both correct.

5. If incorrect (IFILL=1), open file which contains correct parameters. Default filename is CORECT.DAT.

6. IVER (I)

IVER = 1 if the title of the ++ perturbed case is correct.

7. IVER (I)

IVER = 1 if the title of the +- perturbed case is correct.

8. IVER (I)

IVER = 1 if the title of the -+ perturbed case is correct.

9. IVER (I)

IVER = 1 if the title of the -- perturbed case is correct.

If the titles of any of the cases are incorrect, go to record #1.
If both of the parameters are correct, go to record #13.

10. NP1,NP2 (2I)

NP1 = Order number of parameter 1 in the array of new parameters.
Enter 0 if this parameter is correct on the response files.

NP2 = Order number of parameter 2 in the array of new parameters.
Enter 0 if this parameter is correct on the response files.

11. ND (I)

ND = Number of time ranges needed to specify the perturbation for the new parameter.

12. MORA,BT,ET,DEL (I,3F)

All definitions as in RLPRLP, record #14.

Repeat record #12 for each of the ND time ranges. Give records 11 and 12 for parameter 1 first, then for parameter 2. For either parameter, enter ND=0 and skip record #12 if the parameter is already correct.

If all responses are being processed (IALI=0), go to record #16.

13. LIST (I)

LIST = 0 to input a new list of responses.

= 1 to keep the previous list. Go to record #16.

14. NRES (I)

NRES = Number of responses in new list.

15. (HRES(I),I=1,NRES) (20A4)

HRES(I) = Four-character identifier of the Ith response.

16. MORE (I)

If no printout of mixed derivatives is desired (IALI=-1), go to record #1. If printouts are desired, first 10 time steps for each response will always be edited. To get succeeding 10 time step sections, enter MORE=1. Enter MORE=0 to go on to the next response. After the last response, go to record #1.

Option "U"

2. STYPE (A1)

STYPE = One-character identifier of type of file to update. Use "S" to update a long-form sensitivity file, "B" to update a mixed derivative file.

3. Open input file. Default filename is SENSIT.IN if STYPE=S, MIXD.IN if STYPE=B.

Open output file. Default filename is SENSIT.OUT if STYPE=S, MIXD.OUT if STYPE=B.

Open file which contains the new parameters. Default filename is NEWREF.DAT.

4. NP1,NP2 (2I)

NP1 = Order number of parameter 1 in the array of new parameters.
Enter 0 if this parameter is correct on the sensitivity file.

NP2 = Order number of parameter 2 in the array of new parameters.
Enter 0 if this parameter is correct on the sensitivity file.
NP2 need not be entered if this is a single-parameter file.

Go to record #1.

Option "L"

2. Open long-form input file. Default filename is LONG.SEN.

Open short-form output file. Default filename is SHORT.SEN.

Go to record #1.

4.5 Uncertainty Analysis

The sensitivity coefficient files created in the previous step are used by the interactive program UNCOVER in the uncertainty analysis. Either the short-form or long-form sensitivity files may be used, but the type of file must be the same for all parameters. Any or all of the responses may be included in the analysis and any pair of responses may be specified for the response covariance calculations. However, the responses used in the response covariance calculations must be among those used in the variance calculations.

The uncertainties of the parameters of interest are the 1σ variances (square of the standard deviation) supplied by the experimental group. These may be input to the program either through the terminal or from a separate PDP-10 file. If entered through the terminal, the variance must be specified as constant values over a set of time ranges, in the same way that the perturbations are supplied to the RLPRLP

program. If, however, the variance is given as a percentage of the reference value of the parameter, then it is easier to set up a binary file that contains the variance at every time step. This file should consist of a single record containing the array of variances for the parameter. If more than one parameter will have its variance specified in this manner, each must be given in a separate PDP-10 file.

Input to UNCOVER

1. NP,NT,MIX,NHRC (4I)

NP = Number of parameters.

NT = Number of time steps.

MIX = Number of mixed derivatives.

NHRC = Number of responses involved in response covariance calculations.

2. Open output uncertainty file. Default filename is VAR.BIN.

3. IALL (I)

IALL = 0 to process all responses found on the sensitivity file for the first parameter. Go to record #6.

> 0 to specify a list of responses. Go to record #4.

4. NRES (I)

NRES = Number of responses to process.

5. (HRES(I),I=1,NRES) (20A4)

HRES(I) = Four-character identifier of Ith response.

6. MESS (I)

MESS = 1 to print the number of times that the first order contribution to the variance was less than 90% of the total variance.

= anything else to give no printout.

7. IS (I)

IS = 0 if the long form of the sensitivity files will be used.

= 1 if the short form will be used.

If NHRC = 0, go to record # 11.

8. NHRCP (I)

NHRCP = Number of response pairs for which to calculate covariances.

9. HRCP(1),HRCP(2) (2A4)

HRCP(1),HRCP(2) = Four-character identifiers of the two responses in the first pair. These may be the same, in which case variances will be calculated.

Repeat record #9 for each of the NHRCP pairs. The code will count up the number of unique identifiers in the NHRCP pairs and compare it with the input value of NHRC. If the two agree, go to record #11.

10. DISGST (A1)

DISGST = "1" to try a new set of response pairs. Go to record #8.

= anything else to change your mind and skip the covariance calculations. Go to record #11.

11. Open input sensitivity coefficient file for first parameter.
Default filename is SENS.DAT.

12. IFV (I)

IFV = 1 to read time-dependent variances from a separate file.
Go to record #13.

= anything else to enter the variances at the terminal.
Go to record #14.

13. Open file with variances. Default filename is VARTBL. Go to record #11.

14. ND (I)

ND = Number of time ranges needed to specify the variances.

15. BT,ET,V (3F)

BT,ET = Beginning and end time of first range.

V = Constant variance for the first range.

Repeat record #15 for each of the ND time ranges.

Repeat sequence of records 11-15 for each of the NP parameters.

If no mixed derivatives are being used, terminate the job; otherwise go to record #16.

16. Open mixed derivative file for the first mixed derivative. Default filename is MIXD.DAT.

Repeat record #16 for each of the MIX mixed derivatives.

4.6 Editing of Binary Files

The EDITOR interactive program is available for the production of readable edits of every binary file described in Appendix A. The user has the option of either editing every response on the file or of specifying a list of responses to be edited. For a response covariance file, edits are produced for every pair of responses in which at least one of the responses is among the input list. Similarly, the user may choose either the entire time range on the file, or up to 10 segments of it.

For all binary files except the uncertainty file containing the expected response values and standard deviations (see Sect. A.4), the output edit file contains the information just as it existed on the binary input file. For the uncertainty file, however, the user is given the additional option of applying constraints to the expected value of a response and to the standard deviations of the reference and expected values of the response. If a response exhibits a highly nonlinear behavior with respect to an input parameter, the equations for expected value and standard deviation (Eqs. 10, 12 and 14 in Sect. 2) may yield a physically impossible or absurd value which the user does not wish to accept. For example, the water quality in a RELAP model volume must lie in the range [0,1]. If the user specifies such a constraint on all of the water quality responses, any values lying outside of this range will be reset to the appropriate bound and an error flag will be set in the printout. This does not affect the data in the binary file, only its representation on the edit.

Input to EDITOR

1. HOPT (A1)

- HOPT = "R" to edit a RELAP response file. (See Sect. A.1.)
- = "L" to edit a single-parameter sensitivity file, long form. (See Sect. A.2.)
- = "S" to edit a single-parameter sensitivity file, short form. (See Sect. A.2.)
- = "B" to edit a mixed derivative file. (See Sect. A.3.)
- = "U" to edit the uncertainty file with expected values and variances. (See Sect. A.4.)
- = "C" to edit the uncertainty file with response covariances. (See Sect. A.5.)
- = Anything else to terminate the job.

Input records 2 through 6 for any of the options.

2. Open binary input file. Default filename is INPUT.BIN.
Open BCD output file. Default filename is OUTPUT.BCD.

3. IALL (I)

- IALL = 0 to perform edit for every response on input file.
- > 0 to perform edit for a new input list of IALL responses.
- < 0 to perform edit for the current list of -IALL responses.

If IALL $\neq$ 0, go to record #5.

4. (HIN(I), I=1, IALL) (20A4)

HIN(I) = Four-character identifier of response I.

Note: If a response covariance file is being edited, the covariances for a pair of responses will be edited if at least one of them is in the list of IALL responses.

5. NTT (I)

NTT = Number of time ranges to be edited. Maximum is 10.

6. BT,ET,NEWP (2F,I)

BT,ET = First and last time in a range. Input ET=0.0 to get all times from ET on.

NEWP = New page parameter.

= 1 to start this range on a new page.

= Anything else to just skip a line after the previous range.

Repeat record #6 NTT times.

If any option but "U" has been specified, go to record #1. If option "U", input records 7 through 10.

7. MODEST (I)

MODEST = 0 to get seven-digit printout of both responses and standard deviations.

= 1 to get four digits for responses, two digits for standard deviations.

8. NCON (I)

NCON = Number of constraints to be applied on the printed output. Maximum is 10.

If NCON=0, go to record #1.

9. (HCON(I),I=1,NCON) (10A2)

HCON(I) = Two-character identifier of response type I for which constraints will be applied. This is just the first two characters of the appropriate RELAP variables.

10. (CONSTR(J,I),J=1,2) (2F)

CONSTR(1,I) = Lower bound on expected value of response, standard deviation of nominal response and standard deviation of expected response.

CONSTR(2,I) = Upper bound on same quantities.

Go to record #1.

5. FURTHER SYSTEM DEVELOPMENT

The software package described in Sects. 3 and 4 can be used in a straightforward manner to perform an uncertainty analysis on a thermal-hydraulic system modeled with the RELAP code. The user need only follow the detailed input instructions described in Sects. 4.1 to 4.6 to proceed step by step from parameter perturbation to editing of final results. However, there are several important situations which call for a somewhat different use of the system or for a change to the components of the system. These are the automation of the normal process to handle repetitive calculations and the conversion of the entire package to a different base code.

5.1 Automated Processing

In the normal use of this software package, the user would run each component separately, repeating components as necessary. For example, the file shown in Fig. 2 would be set up and submitted twice for each parameter, as would the execution of RLPRLP. RLPSEN would be run once for each parameter and UNCOVER once for the entire system. Unless these are being run in a batch mode, all input would be entered from the terminal as it was needed.

There are many cases, however, in which certain components may have to be run many times, with each run only slightly (and systematically) different from the previous. In these cases, it may be more efficient to use a PDP-10 MIC file to create the input to the appropriate components and then run them. A MIC file can contain the programming for running the repetitive calculations. An example will be given to illustrate the use of MIC files in these situations. A general knowledge of MIC file formats is assumed.

Assume that a system is being modeled which has as one of its boundary conditions the flow rate through junction 5, JW05. This is the main parameter of interest and is located in the fill table of Section 81 in the input card-image file. The modeler wishes to calculate the sensitivities of various responses with respect to JW05. As a test of the linearity of the responses, he intends to calculate the sensitivities due to a range of perturbations of JW05, starting from very small and going to several times the standard deviation of JW05. For every perturbation, he must run RLPCRE to make the perturbation (once positive and once negative), submit the file in Fig. 2 to run RELAP, submit the file in Fig. 3 to access the desired responses from the RELAP output file, run RLPRLP to create the binary response files and, finally, run RLPSEN to calculate the sensitivities. To illustrate the use of a MIC file in automating this procedure, we will show how to combine the first two steps.

Given that the reference RELAP card-image file is on PDP-10 file RELAP.IN and has two time steps per line in the fill table of section 81, the MIC file listed in Fig. 4 will execute RLPCRE, run REDSEL with dummy arguments just to get the reference OXREPT tape to the RLPSFLUX job step and run RLPSFLUX. The PDP-10 file in Fig. 2 contains the REDSEL and RLPSFLUX job steps and is assumed in the MIC file of Fig. 4 to have the filename RELRDX. The perturbations are in units of 10% of the standard deviation. If the name of the file in Fig. 4 was AUTO.RLP, then the PDP-10 command for executing this procedure would be

```
.DO AUTO.RLP A,B,C
```

where A = Standard deviation of JW05

B = Starting point for perturbation loop, in units of 10% of the standard deviation

C = Number of perturbations to make

Thus, if the command .DO AUTO.RLP 3.6,3,12 was entered, 12 different perturbations would be used, ranging from 30% to 140% of the standard deviation of 3.6. Both the positive and negative perturbations would be made, and the RELAP output files would be named E.UID####.AUTOPnn and E.UID####.AUTONnn, respectively. The two-digit counter, nn, would range from 03 to 14 to identify the various perturbations.

The use of a MIC file such as listed in Fig. 4 could greatly reduce both the amount of time needed to run each of the components and the number of input errors which are likely to occur in repetitive situations. In principle, a standard automated procedure could be set up to perform every function of the software package, from perturbation of the parameters to editing of the final results. This would require at least two automated MIC files: one for all functions up through the IBM steps (RELRDX and PNCK10) and another for the succeeding steps. To ensure reasonable quality assurance, however, one would have to split the automated procedure into enough steps to adequately check intermediate results before proceeding.

5.2 Conversion to a Different Base Code

Although this software package has been designed to perform sensitivity and uncertainty analyses with the version of the RELAP code in use at ORNL, it could, at least in principle, be adapted to handle any base code. The only requirements are that the user be able to identify and designate the input parameters and output responses of interest and that the code be able to output a data file of calculated responses. Conversion to a different base code is accomplished in two stages:

```

.SILENCE
.LET D=B+C-1
.LET E=B-1
; Loop over perturbation steps
LOOP::
.LET E=E+1
.IF(E>D) .GOTO STOPIT
; Multiply st. dev. in parameter A by perturbation index in
; parameter E and divide by 10. St. dev. must be in the form x.x.
; Parameter I will hold the integer part of the product, parameter
; R the fractional part.
.LET I=E*A
.LET R=$A.[".",1]
.LET R=E*R
.IF(R<10) .GOTO CONTINUE
.LET J=$R.[1]
.LET I=I+J
.LET R=$R.[2]
CONTINUE::
; Recombine the integer and fractional parts of the product and
; append "E-1" to facilitate division by 10. The result is placed
; in parameter P.
.LET P=$I+"."+ $R+"E-1"
!
! Step 'E
! Perturbation = 'P
;
; First do the positive perturbation
; Run RLPCRE to make the changes to the card-image file
.EX RLPCRE
RELAP.IN$$81
2
0
0 0 0. 20. 'P
-1
99
0
; Now submit file RELRDX to run RELAP
.TECO RELRDX
SRLP EXEC$6LKI// UNIT=3330V,VOL=SER=VMASTR,DSN=E.RELAPOUT.P'E
$EX$$
!
! RELAP output file will be on E.RELAPOUT.P'E
!
.IBM RELRDX
; Now do the negative perturbation
! Perturbation = -'P
!
.EX RLPCRE
RELAP.IN$$81
2
0
0 0 0. 20. -'P
-1
99
0

```

Figure 4. MIC File for Automated Processing

```
.TECO RELRDX
SRLP EXEC$6LKI// UNIT=3330V,VOL=SER=VMASTR,DSN=E.RELAPOUT.M'E
$EX$$
! RELAP output file will be on E.RELAPOUT.M'E
!
.IBM RELRDX
.BACKTO LOOP
STOPIT::
.MIC EXIT
```

Figure 4. Continued

- i) Rewrite the routines which manage the input to and output from the base code.
- ii) Adjust the interactive programs and binary file specifications to handle any changes required by step i.

The first stage in the conversion is by far the more difficult. It covers the replacement of routines which perturb the input to the code (the interactive program RLPCRE and the job submission file shown in Fig. 2), which run the code (Fig. 2) and which access the output from the code (Fig. 3). The user must precisely define his parameters and responses and provide some means for changing them. For the RELAP code, an auxiliary code, REDSEL, already existed for changing the input on the OXREPT data file. For the parameters in the card-image model input, however, a routine (RLPCRE) had to be written to make the changes.

Since the RELAP code outputs a data file which contains every quantity defined in the model and another auxiliary code, REDPLOT, already existed to access this data file, it was quite easy to prepare the output data for the sensitivity analysis. If the new base code does not automatically create an output file, but only provides printed output, then the user will have to design an output file and provide a code to read it. The first step in this process, however, is to assign a naming convention for all the calculated or input quantities which will later be used in the analysis. The RELAP code has its own four-character identification scheme for all model quantities, and this scheme was carried through the entire sequence of calculations. Changing to a different type of identification will be discussed below in the second stage of the conversion.

The output of the code which accesses the data on the base code output file must be a BCD-formatted file which can be routed to the user's PDP-10 space. The RLPRLP interactive program reads this file and creates the binary response file described in Sect. A.1. To minimize the changes necessary to the RLPRLP code, the user's new code should, if possible, put the parameters and responses out in the form expected by RLPRLP. This form is described in Sect. 4.2 and Fig. 3. If it is impossible or cumbersome to provide this form of the output, then RLPRLP can be adjusted to accept the new form.

In the second stage of the conversion, the interactive programs and the file specifications of the binary PDP-10 files are modified as necessary. Unless there is some pressing need, the only changes made to the files described in Appendix A should be in the type of identifier used for the parameters and responses. If the user desires an identifier that is a number or a literal with the number of characters different than four, he can easily change the file specifications that define the parameters or responses.

Once the file specifications have been modified, each of the programs RLPRLP, RLPSEN, UNCOVER and EDITOR must be modified to accept the new naming convention. Identifiers currently are always input in A4 fields. If the new convention requires a literal with fewer than four characters, then no changes are required; the input identifiers would be padded (by the user) with blanks to fill the A4 field. If a number or a literal with more than four characters is required, then every FORMAT statement in the programs that inputs or outputs the identifiers must be changed. Since a word on the PDP-10 is five bytes long, a maximum of five characters can be stored in the single-precision variables currently containing the identifiers. Up to 10 characters may be obtained by declaring these variables to be double-precision. Identifiers longer than 10 characters would require replacing each single variable by an array of the appropriate length.

With one exception, the only inspection or checking of the parameter and response identifiers is to determine a possible match with an input list of other identifiers. For example, the user might supply to RLPSEN a list of responses for which to calculate sensitivities. The code would then check the response identifiers on the various binary response files against this list. Thus, once the user has modified the codes to input and output the new identifiers, he need not worry about how they are used. They will simply appear in statements such as

```
IF(HR.EQ.HRES(I)) - - -
```

where HR and HRES(I) are identifiers, and the particular interpretation of the identifiers will not matter.

The exception to this rule is in the editing of the uncertainty file created by the UNCOVER program. If constraints are to be applied to certain types of responses, these are denoted by the first two characters of the identifier. For example, if the user wanted to constrain the water quality in any volume to be in the range 0.0 to 1.0, he would input (see Sect. 4.6) HCON = "AX" and the EDITOR program would search for identifiers which started with the letters AX. This function, which is obviously tailored around the RELAP naming convention, would have to be modified somewhat to work with the user's new naming convention.

REFERENCES

1. "RELAP4/MOD5 - A Computer Program for Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Related Systems: User's Manual," ANCR/NUREG-1335, September 1975.
2. Y. Ronen, M. A. Bjerke, D.G. Cacuci, J. Barhen, "A Nonlinear Sensitivity and Uncertainty Analysis in Support of the Blowdown Heat Transfer Program," ORNL/NUREG/TM-412, November 1980.
3. "Project Description, ORNL PWR Blowdown Heat Transfer Separate-Effects Program - Thermal-Hydraulic Test Facility (THTF)," ORNL/NUREG/TM-212, November 1978.
4. S. B. Cliff, et al., unpublished data, January 1978.
5. J. W. Teague, II, et al., unpublished data, April 1979.
6. L. J. Schlemper, internal memorandum, September 16, 1977.
7. E. Wacholder, M. A. Bjerke, J. Barhen, C. F. Weber, "Nonlinear Time-Dependent Sensitivity and Uncertainty Analysis for Post-CHF Heat Transfer to High Pressure Water Upflow," ORNL/TM-7543, to be published, 1982.
8. J. Barhen, D. G. Cacuci, J. J. Wagschal, M. A. Bjerke, C. B. Mullins, "Uncertainty Analysis of Time-Dependent Nonlinear Systems: Theory and Application to Blowdown Thermal-Hydraulics," Nucl. Sci. Eng., 81, 23-44 (1982).
9. C. F. Weber, personal communication to E. Wacholder, July 1981.
10. L. J. Ott, R. A. Hedrick, "ORINC - A One-Dimensional Implicit Approach to the Inverse Heat Conduction Problem," ORNL/NUREG-23, November 1977.
11. S. B. Cliff, L. I. Schlemper, R. M. Flanders, A. F. Johnson, et al., unpublished data, January 10, 1978.
12. S. B. Cliff, "OXREPT - ORINC and Experimental Data to RELAP Plot Tape Interface," ORNL/NUREG/CSD/TM-3, May 1978.

Appendix A

File Specifications

For the storage of intermediate and final results in the uncertainty analysis described in this report, five different data files were defined. These are binary (unformatted) files which save both disk space and computing time over similar BCD-formatted files. The EDITOR program described above produces readable edits of the data in these files. These files contain i) the responses from the RELAP runs, both reference and perturbed, ii) the first and second order sensitivity coefficients, iii) the mixed derivatives, iv) the uncertainty analysis results and v) the response covariances. The contents of each of these five files will be described below so that the user may access the data for special purposes, such as different edited formats or extensions to the current software.

Before describing the files, several precautions are in order which apply to all of the files. On the files which contain the RELAP responses, sensitivity coefficients and mixed derivatives, a parameter number and four-character identifier are included to indicate which parameter was perturbed. Although the numbering and naming of a parameter is arbitrary and two perturbed cases are never checked to ensure that their parameter numbers and identifiers match, it is advisable to assign a unique number and name to each parameter being studied. This will help avoid any possible confusion when compiling or discussing results.

Included in each file is a record containing the number of responses (or response pairs if on the response covariance file) contained in the file. As mentioned in Sect. 4, it is possible for this number to be greater than what is actually on the file. For some of the codes, which simply read through the file until an end-of-file is reached, this is not a problem. Other codes, however, read the file inside a DO loop over the specified number of responses. For those cases, the job might be aborted due to an attempt to read past the end of the file. Since there is always some indication that this error has occurred, the user should be able to quickly identify the problem. Given the description of the contents of each file, he can then write a very simple FORTRAN program to correct the problem.

A.1 RELAP Responses

This file contains the time-dependent values of the responses of interest. A parameter number indicates whether the file contains values from the reference case or from one of the perturbed cases. If the latter, the file also contains the reference values of the parameter and the amount of the perturbation.

* * * * *

Record 1: 80-character title. Access as A(80)

Record 2: NR,NT,DT,TZ,IP,HPAR1,HPAR2

where NR = Number of responses

NT = Number of time steps

DT = Time step size (seconds)

TZ = Beginning time (seconds)

IP = parameter number, which must conform to one of the following options

- i) IP = -1 for the reference case
- ii) $0 < IP < 999$ for perturbation of the single parameter IP. Sensitivities are calculated at each time step using the perturbation supplied for that time step.
- iii) IP = 0 for the perturbation of two parameters. Sensitivities calculated as in option i).
- iv) $IP < -1$ for a single parameter $ABS(IP+1)$ changed at a single time T. Sensitivities calculated at times greater than T using the perturbation at time T.
- v) IP = 999 for two parameters changed at the single time T and sensitivities calculated as in option iv).

HPAR1,HPAR2 = Four-character identifiers of the perturbed parameters. For the reference case, both should fill the blanks. For single-parameter perturbations, HPAR2 should be left blank, while for two-parameter perturbations both are used.

Record 3: (DELTA(1,K),K=1,NT)

where DELTA(1,K) = Perturbation of the first parameter at time step K.

Record 4: (PARAM(1,K),K=1,NT)

where PARAM(1,K) = Reference value of the first parameter at time step K.

Record 5: (DELTA(2,K),K=1,NT)

where DELTA(2,K) = Perturbation of the second parameter at time step K.

Record 6: (PARAM(2,K),K=1,NT)

where PARAM(2,K) = Reference value of the second parameter at time step K.

Omit records 3-6 for the reference case or records 5 and 6 for a single-parameter perturbation. Use each of records 3-6 for a two-parameter perturbation.

Record 7: HRES = Four-character response identifier.

Record 8: (RES(K),K=1,NT)

where RES(K) = Value of the response identified as HRES at time step K.

Repeat records 7 and 8 for each of the NR responses.

* * * * *

A.2 Sensitivity Coefficients

This file contains the first and second order sensitivity coefficients for a single parameter, as well as the reference value and perturbation amounts for that parameter. Two forms of this file are defined. The long form contains, for each response, the nominal and perturbed values of the response and both the absolute and relative first and second order sensitivity coefficients. This is a lot of information, and for a case with a large number of responses and time steps requires a large amount of storage. In the uncertainty analysis performed by the UNCOVER program, the sensitivity files for all parameters must be available at the same time. Thus, an analysis that uses several parameters may require more disk storage than the user has available. In this case, the short form of the sensitivity coefficient file could be useful. For each response, this form of the file carries only the reference value of the response and the absolute first and second order sensitivity coefficients - the minimum required in the uncertainty analysis. Only about one-third as much disk storage is required for the short form of the file compared to the long form.

* * * * *

Record 1: 80-character title of RELAP response file with positive perturbation -- A(80).

Record 2: 80-character title of RELAP response file with negative perturbation -- A(80).

Record 3: NT,DT,TZ,NR,IP,HPAR

where NT = Number of time steps.

DT = Time step size (seconds).

TZ = Beginning time (seconds).

NR = Number of responses.

IP = Parameter number; same definitions as described for the RELAP response file.

HPAR = Four-character identifier for the parameter.

Record 4: (DELTA(K),K=1,NT)

where DELTA(K) = Amount of perturbation of this parameter in the positive perturbation case, at time step K. A check in the code RLPSEN ensures that the perturbation in the negative case is equal and opposite that in the positive case.

Record 5: (PARAM(K),K=1,NT)

where PARAM(K) = Reference value of the parameter at time step K.

Record 6: HRES = Four-character response identifier.

For the long-form sensitivity coefficient file, the following record is used.

Record 7: RZ,RP,RM,S,H,RELS,RELH

where RZ = Response from reference case.

RP = Response from positive perturbation case.

RM = Response from negative perturbation case.

S = Absolute first order sensitivity coefficient.

H = Absolute second order sensitivity coefficient.

RELS = Relative first order sensitivity coefficient.

RELH = Relative second order sensitivity coefficient.

Record 7 is repeated for each of the NT time steps.

For the short-form file, record 7 should be as follows.

Record 7: (RZ(K),S(K),H(K),K=1,NT)

RZ, S and H have the same definitions as in the long-form file. For both forms, repeat the sequence of records 6 and 7 for each of the NR responses.

* * * * *

A.3 Mixed Derivatives

This file contains the derivatives (sensitivity coefficients) calculated for each response when two parameters are perturbed. As explained in Sect. 2, these derivatives can be calculated, to a first approximation, using either the case in which both parameters are perturbed in a positive sense, or the case in which they are both perturbed in a negative sense. A somewhat better approximation is obtained by using both cases. Whichever is used, only the title records can be used to indicate the method by which the file was created. Thus, one must ensure that the descriptive titles being used, which are taken from the appropriate RELAP response files, are sufficient documentation.

* * * * *

Record 1: 80-character title of RELAP response file with the two-parameter perturbation -- A(80).

Record 2: NR,NT,DT,TZ,HPAR1,HPAR2

where NR = Number of responses.

NT = Number of time steps.

DT = Time step size (seconds).

TZ = Beginning time (seconds).

HPAR1,HPAR2 = Four-character identifiers of the two parameters.

Record 3: (PARAM(1,K),K=1,NT)

where PARAM(1,K) = Reference value of first parameter (HPAR1) at time step K.

Record 4: (PARAM(2,K),K=1,NT)

where PARAM(2,K) = Reference value of second parameter (HPAR@) at time step K.

Record 5: HRES = Four-character response identifier.

Record 6: (AMD(K),RMD(K),RZ(K),K=1,NT)

where AMD(K) = Absolute mixed derivative at time step K.

RMD(K) = Relative mixed derivative at time step K.

RZ(K) = Reference value of response at time step K.

Repeat records 5 and 6 for each of the NR responses.

* * * * *

A.4 Uncertainty Analysis Results

The uncertainty analysis program, UNCOVER, calculates the expected value of every response and the standard deviations of the nominal and expected values of every response. For presenting the results, however, a more useful set is the expected value of the response and its standard deviation, and the difference between the nominal and expected values of the response and standard deviation. Since the difference between nominal and expected is often quite small, more accuracy is retained by storing the differences and recreating the nominal values in the editing program. For those users who wish to know the effect of the second order terms in the calculation of the standard deviation, the first order contribution to that quantity is also stored.

* * * * *

Record 1: NP,NT,DT,TZ,NR

where NP = Number of parameters.

NT = Number of time steps.

DT = Time step size (seconds).

TZ = Beginning time (seconds).

NR = Number of responses.

Record 2: 80-character title for first parameter - - A(80).

Repeat record 2 for each of the NP parameters.

Record 3: HRES = Four-character response identifier.

Record 4: (R_{BAR}(K), DSV(K), SDR_{BAR}(K), DVAR(K), F2V(K), K=1, NT)

where R_{BAR}(K) = Expected value of response HRES at time step K.

DSV(K) = Fractional difference between the expected and nominal values of the response. If R is the nominal value and R the expected, then $DSV = 1 - R / R$.

SDR_{BAR}(K) = Standard deviation of the expected value of the response.

DVAR(K) = Fractional difference between the expected and nominal values of the variance. If VAR is used to denote variance, then $DVAR = 1 - VAR(R) / VAR(R)$.

F2V(K) = First order contribution to the variance.

Repeat records 3 and 4 for each of the NR responses.

Note that we can recreate the nominal values of the response and its standard deviation from the above quantities.

$$R = R_{BAR} * (1 - DSV)$$

$$\text{St. Dev. of } R = SDR_{BAR} * \sqrt{1 - DVAR}$$

* * * * *

A.5 Response Covariances

This file contains the covariances and correlation coefficients between pairs of responses and is created by the UNCOVER program along with the uncertainty analysis file described in Sect. A.4.

* * * * *

Record 1: NRP, NT, DT, TZ, NP

where NRP = Number of pairs of responses.

NT = Number of time steps.

DT = Time step size (seconds).

TZ = Beginning time (seconds).

NP = Number of parameters.

Record 2: 80-character title for the first parameter -- A(80).

Repeat record 2 for each of the NP parameters.

Record 3: HA,HB = Four-character identifiers for the two responses.

Record 4: (COV(K),CORCOF(K),K=1,NT)

where COV(K) = Covariance of responses HA and HB at time step K.

CORCOF(K) = Correlation coefficient of responses HA and HB at time step K.

Records 3 and 4 are repeated for each of the NRP response pairs.