A GENERAL STUDY OF UNDERSAMPLING PROBLEMS IN MONTE CARLO CALCULATIONS

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Wilson José Vieira

December, 1989

A GENERAL STUDY OF UNDERSAMPLING PROBLEMS IN MONTE CARLO CALCULATIONS

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Wilson José Vieira

December 1989

To the Graduate Council:

I am submitting herewith a dissertation written by Wilson José Vieira entitled " A General Study of Undersampling Problems in Monte Carlo Calculations ." I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfilment of the requirements for the degree of Doctor of Philosophy, with a major in Nuclear Engineering.

Paul N Stevens

Paul N. Stevens, Major Professor

We have read this dissertation
and recommend its acceptance:

Bennie F.L. Ward

H.L. Dodds, Jr.

L F Miller

Accepted for the Council:

Vice Provost
and Dean of The Graduate School

DEDICATION

to Ligia and Daniel.

ACKNOWLEDGEMENTS

I would like to thank all the people that contributed in some way for the realization of this work. In particular, the Conselho Nacional de Desenvolvimento Científico e Tecnológico of Brazil, which solely provided financial funds.

The members of the committee to whom I am particular indebted for the experience and knowledge that each one provided for my education.

To my advisor Prof. Paul N. Stevens who actually suggested the theme and with his encouragement and expertise provided important guidelines throughout this work. However, whatever errors remain in my work, are solely my responsibility. Also, I would like to thank Mrs. Barbara Stevens for her continuous encouragement.

ABSTRACT

Various techniques devised to flag undersampling conditions were investigated. Undersampling conditions can often lead to underestimation and a solution which can be significantly smaller than the true solution but the estimate of the standard deviation may seem acceptably small. In an attempt to identify undersampling in Monte Carlo calculations, the estimation of F values, the coefficient of variation of the standard deviation, the figure of merit, and a particle contribution distribution histogram were incorporated into the MORSE code.

It was found for the problems considered that the F tests were not conclusive because the distribution of contributions was not normally distributed. The calculation of the coefficient of variation turned out to require significantly more computational effort than that necessary to directly achieve a very small standard deviation because of its dependency upon the kurtosis of the distribution which takes much longer than the variance to achieve a stable value. If the kurtosis is not too large, this coefficient can be used in problems which demand high degrees of precision such as criticality calculations.

The figure of merit $FOM = 1/\sigma^2 t$ is a function of the variance of the population which becomes stable faster than its coefficient of variation. Therefore, the FOM provides a more reliable guarantee of a stable solution – also, because it tends to a constant value, it is more easily analyzed. However, in severe undersampling conditions the FOM may become apparently constant over a large range of sample sizes and then abruptly changing with the sampling of rare particles. Also, a sudden increase in the variance may not have an accompanying significant change in the mean while the figure of merit experiences a jump. Under this condition, the solution may still be a perfectly acceptable estimate.

The creation of the particle contribution distribution which is output at the end of each

batch provides a very effective way of detecting undersampling. If only a few particles account for a large fraction of the response, the estimates of both mean and standard deviation should be regarded as unreliable and, therefore, the sample size should be increased.

Although not thoroughly investigated, the utilization of the statistical tools implemented into the MORSE code was demonstrated to be useful in the study of the behavior of particle distributions when subjected to various biasing and/or estimation procedures.

Contents

APPENDICES 45

List of Figures

List of Tables

Chapter 1

Introduction

Monte Carlo calculations are possible in almost every field which involve mathematical modeling. All such Monte Carlo analyses comprise of the generation of sequences of random variables and have as solutions estimates of means and variances. The problems can range from the estimation of the mean of a small set of Monte Carlo generated data to the estimation of detailed radiation particle flux distributions which represent the solutions of the familiar integro-differential Boltzmann transport equation.

The quality of a Monte Carlo calculation involves two basic concerns: accuracy and precision. Accuracy is a measure of how close the Monte Carlo estimate is from the true value and is related with the amount of *bias*. Precision is a measure of the statistical uncertainty associated with the estimate and is usually expressed in terms of the standard deviation. Many factors can introduce bias into the Monte Carlo solution of a problem which will affect accuracy as well as the behavior of the estimator of precision. Bias can be caused by inadequacies of the model such as may occur in geometry or cross section descriptions and through the use of some sampling schemes such as point-detector estimators whose expected values are not the true solutions. One interesting way to understand the role of the parameters involved in a statistical estimation is to address the following question: *How can the quality of an estimate be guaranteed*? This raises another question: *Does the population sample represent the true population*? The answer to these questions is the major objective of this work which is to analyze the statistical features of the mean, variance, and population sample and to devise schemes to utilize more of the information generated during the calculation.

Undersampling occurs in problems where the effect of interest is determined primarily by very rare events and a very large population sample is required to achieve a good estimate. In a completely analog Monte Carlo particle transport calculation, the distribution of the scores is binomial and the effect of interest is simply the probability of scoring a success. Therefore the scores of a problem with a small scoring probability would be mostly zeroes and would require on the order of 400 successes (rare events) to achieve a 5% fractional standard deviation – approximately $1/\sqrt{n}$ where n is the number of successes. Undersampling has been recognized as a major source of concern. The inexperienced user may not recognize this condition and accept a solution that can be orders of magnitude too low even though the standard deviation indicates good precision. This problem was pointed out by Gelbard [1] and Cramer et al. [2], both of whom considered that many of the problems associated with statistical uncertainty were still unresolved. Dubi et al. [3] showed that the one-particle method yields more reliable estimates of the variance, but also recognized the usefulness of the batch method. Lux et al. [4] devised a correction scheme for the estimates of the mean and variance and emphasized the importance of distinguishing between rare events and the unimportant background.

The problem of *undersampling* is particularly important when point-detector estimators are used because the contributions to the outer detectors from collisions near the source region are by their nature very small and can be essentially of the same magnitude. If a sufficient number of particles are not sampled to include enough particles which experience rare events, the final results will be unrealistically small while their standard deviations may indicate seemingly acceptable results. In this work, this condition will be called *underestimation*. The computational evolution of a statistically acceptable estimate of an effect of interest, in a deep penetration problem using a point-detector estimator, presents three different stages. The first stage is when undersampling is so severe that no major-contributing particle is sampled. A major-contributing particle is not only a particle that yields a high contribution relative to background values but also the value of its contribution has to be sufficiently high to significantly move the estimated solution up to values around the true mean. One way of detecting the undersampled condition during the first stage would be to analyze the statistics throughout the region between the source and the point of interest. This procedure is based on the fact that the standard deviation should increase with the distance from the source. If it decreases, this condition indicates that the contributions are due to collisions far from the detector and that there were no important collisions (rare events) near the detector. The second stage is when a few major-contributing particles are sampled. The results as they evolve during this stage will experience jumps in the estimated fluxes and also in their variances. Finally, the third stage is reached when a sufficient number of major-contributing particles are sampled and unbiased estimates are achieved. This behavior is more or less obvious and depends upon factors such as the distance in mean free paths between the source and the detector and the utilization of the various variance reduction techniques.

In Chapter 2 some theoretical considerations about the interpretation of Monte Carlo results are presented. Chapter 3 describes the modifications made to a standard version of the MORSE code [5] to accomplish the calculation and output of the additional information. This modified version of MORSE will be designated as the MORSE/STAT package.

Chapter 4 describes the problems studied and the techniques used in their solution, such as type of estimators and variance reduction techniques used. The results for each problem are analyzed in Chapter 4 with respect to the benefits realized by the user through the proper interpretation of the additional information compiled by the bookkeeping procedures implemented into the MORSE code.

Chapter 2

Theoretical Background

The mathematical and statistical bases which underlie this study are presented in this chapter.

2.1 Measures of Error

The variance and mean square error are central to the characterization of the error associated with a sampling distribution. The population variance of an estimator is a measure of the dispersion of the distribution around the mean, and the mean square error which is a measure of the dispersion around the true value of the parameter. The mean square error can be defined as the sum of the variance plus the square bias i.e.

$$MSE = E(x - \overline{x})^2 + E(\overline{x} - \mu)^2. \tag{2.1}$$

Bias is another concept of error and is defined as the difference between the estimated mean and the true value of the parameter. If the bias is equal to zero, then the mean square error is given by the variance alone.

2.2 Estimators for the Mean and Variance

The statistical behavior of estimators commonly used in Monte Carlo calculations can be better understood with the consideration of some *Analysis of Variance* theory. Consider a

Table 2.1: A Standard Analysis-of-Variance Table.

			Group			
1	2	…	i	…	I	
x_{11}	x_{21}	…	x_{i1}	…	x_{I1}	
x_{12}	x_{22}	…	x_{i2}	…	x_{I2}	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
x_{1j}	x_{2j}	…	x_{ij}	…	x_{Ij}	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
x_{1J}	x_{2J}	…	x_{iJ}	…	x_{IJ}	
$\overline{x}_1$	$\overline{x}_2$	…	$\overline{x}_i$	…	$\overline{x}_I$	$\overline{x}$

population sample of I groups each with J elements as shown in Table 2.1. Summing the x_{ij} over j, the mean corresponding to the ith group is given by

$$\overline{x}_i = \frac{1}{J}\sum_{j=1}^{J} x_{ij}. \tag{2.2}$$

An estimator for the total mean μ is the average of the I $\overline{x}_i$ estimates

$$\overline{x} = \frac{1}{I}\sum_{i=1}^{I} \overline{x}_i, \tag{2.3}$$

which is equivalent to the estimator

$$\overline{x} = \frac{1}{N}\sum_{n=1}^{N} x_n, \tag{2.4}$$

where $N = I \times J$. Equation 2.4 effectively considers all elements to belong in just one group, i.e.

$$\overline{x} = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij}. \tag{2.5}$$

It is possible to derive various estimators for the population variance σ^2. The first one S_w^2 is called the variance of *mean square within*, which can be calculated using

$$S_i^2 = \frac{1}{J-1}\sum_{j=1}^{J}(x_{ij} - \overline{x}_i)^2 \tag{2.6}$$

so that

$$S_w^2 = \frac{1}{I}\sum_{i=1}^{I} S_i^2. \tag{2.7}$$

This estimator is used when results from several runs are combined and represents the population variance of the grand mean.

Another estimator for the population variance S_b^2 is called variance of the *mean square between*, which is based on the fact that for a sufficient number of elements, the group means are normally distributed with a variance equal to the population variance divided by J

$$\frac{\sigma^2}{J} = \frac{1}{I-1}\sum_{i=1}^{I}(\overline{x}_i - \overline{\overline{x}})^2, \tag{2.8}$$

and

$$S_b^2 = \frac{J}{I-1}\sum_{i=1}^{I}(\overline{x}_i - \overline{\overline{x}})^2. \tag{2.9}$$

This is the *batch* estimate of the population variance and is the procedure used in standard versions of the MORSE Monte Carlo code.

Finally, a third estimator for the population variance is designated as S_t^2 and is called the variance of the *mean square total*,

$$S_t^2 = \frac{1}{N-1}\sum_{i=1}^{I}\sum_{j=1}^{J}(x_{ij} - \overline{\overline{x}})^2. \tag{2.10}$$

This is the *one particle* estimate of the population variance and is the procedure used in most Monte Carlo programs.

Any of the equations 2.11, 2.12, or 2.13 can be used to calculate an estimate for the population variance S^2 and estimates for the variance of *mean* $S_{\overline{\overline{x}}}^2$ are given by:

$$S_{\overline{\overline{x}}}^2 = \frac{1}{N}S_w^2, \tag{2.11}$$

$$S_{\overline{\overline{x}}}^2 = \frac{1}{N}S_b^2, \tag{2.12}$$

and

$$S_{\overline{\overline{x}}}^2 = \frac{1}{N}S_t^2. \tag{2.13}$$

Equations 2.12 and 2.13 represent the *batch* estimator and the *one particle* estimator of the variance of the mean, respectively.

The fractional standard deviation (FSD) is the estimate of precision calculated in the MORSE code and is given by:

$$FSD = \frac{\sqrt{S_{\bar{\bar{x}}}^2}}{\bar{\bar{x}}}. \tag{2.14}$$

Equations 2.7, 2.9, and 2.10 can be written as

$$S_w^2 = \frac{SSW}{I(J-1)}, \tag{2.15}$$

$$S_b^2 = \frac{SSB}{I-1}, \tag{2.16}$$

$$S_t^2 = \frac{SST}{N-1}. \tag{2.17}$$

Remembering the *sum-of-squares* identity

$$\sum_{i=1}^{I}\sum_{j=1}^{J}(x_{ij}-\bar{\bar{x}})^2 = J\sum_{i=1}^{I}(\bar{x}_i-\bar{\bar{x}})^2 + \sum_{i=1}^{I}\sum_{j=1}^{J}(x_{ij}-\bar{x}_i)^2, \tag{2.18}$$

the following relationship follows

$$SST = SSB + SSW. \tag{2.19}$$

Therefore, just two of the three sums have to be calculated and the other can be obtained from Equation 2.19.

2.3 An F-test for the Equality of the Group Means

One way to guarantee sufficient accuracy in the Monte Carlo estimate is to observe the individual solutions of the groups which comprise the total solution. It is expected that the group means are normally distributed, but suppose that one of the groups yielded a much larger value for its mean. This would strongly suggest the existence of undersampling in all the batches — since the batches were drawn from the same population, individual groups having equal numbers of samples should have essentially the same estimates for their means and variances. Therefore, it would be necessary to increase the group size so that the distribution of the group means becomes more normal and also the variance of the grand mean becomes smaller. Before proceeding further, expressions for the expected values of S_b^2 and S_w^2 are derived. From a linear model, each element x_{ij} from the i-th group is written as

$$x_{ij} = \mu_i + \epsilon_{ij}, \qquad (2.20)$$

where μ_i is the mean of the i-th group and ϵ_{ij} is the random error of the individual samples which is assumed to have a mean zero and a variance σ^2. The x_{ij} can be expressed in terms of the the group mean μ_i and the grand mean μ

$$x_{ij} = \mu + \alpha_i + \epsilon_{ij}, \qquad (2.21)$$

where

$$\mu_i = \mu + \alpha_i. \qquad (2.22)$$

Therefore, it will be possible to distinguish between the variance production due to the variation of the group means μ_i and that due to the variation of the elements of the population ϵ_{ij}. The *fixed effect* model is characterized by $\sum_{i=1}^{I} \alpha_i = 0$, otherwise the *random effect* model [6] applies. However, since both models utilize the same F-test, the fixed effect model will be used — which should be appropriate for the present analysis.

The expected value of S_b^2 can be calculated by substituting

$$\overline{x}_i = \mu + \alpha_i + \overline{\epsilon}_i \qquad (2.23)$$

and

$$\overline{x} = \mu + \overline{\epsilon} \qquad (2.24)$$

into Equation 2.9, which yields

$$E(S_b^2) = \frac{J}{I-1} \sum_{i=1}^{I} E(\alpha_i + \overline{\epsilon}_i - \overline{\epsilon})^2. \qquad (2.25)$$

Since $\sum_{i=1}^{I} \alpha_i = 0$ and $E(\epsilon_{ij}) = 0$, Equation 2.25 can be rewritten as

$$E(S_b^2) = \frac{J}{I-1} \sum_{i=1}^{I} E(\alpha_i^2 + \overline{\epsilon}_i^2 + \overline{\epsilon}^2). \qquad (2.26)$$

With the fact that

$$E(\bar{\epsilon}_i^2) = \sigma^2/J, \tag{2.27}$$

and

$$E(\bar{\epsilon}^2) = \sigma^2/IJ, \tag{2.28}$$

Equation 2.26 becomes

$$E(S_b^2) = \sigma^2 + \frac{J}{I-1}\sum_{i=1}^{I}\alpha_i^2, \tag{2.29}$$

which can be written as

$$E(S_b^2) = \sigma^2 + J\overline{\alpha}^2. \tag{2.30}$$

The expected value of S_w^2 can be derived by substituting $E(\bar{x}_i) = \mu$ into Equation 2.7. That is

$$E(S_w^2) = E(\frac{1}{I}\sum_{i=1}^{I}\frac{1}{J-1}\sum_{j=1}^{J}(x_{ij}-\mu)^2), \tag{2.31}$$

which yields

$$E(S_w^2) = E(\frac{1}{I}\sum_{i=1}^{I}\sigma^2) = \sigma^2. \tag{2.32}$$

A statement of the F-test for the equality of the group means would be:

$$\begin{array}{l} H_0 : \mu_1 = \mu_2 = \ldots = \mu_I \\ H_1 : \text{at least one is different,} \end{array} \tag{2.33}$$

and it is performed by calculating

$$F_{(I-1)(N-I)} = \frac{S_b^2}{S_w^2}, \tag{2.34}$$

which has as expected value

$$\frac{\sigma^2 + J\overline{\alpha}^2}{\sigma^2}. \tag{2.35}$$

When the null hypothesis is true, the test yields a value of one. The ability to perform such tests is one of the most important features of the batch method.

From the above discussion there are some important observations. First, S_b^2 is greater than or at least equal to S_w^2. Therefore, S_w^2 and S_t^2 are more reliable because they provide tighter bounds for the statistical error. Second, as demonstrated above, S_t^2 can be used together with S_w^2 to calculate F values that may detect anomalous differences between the group means — which can be an effect associated with undersampling. Another interesting fact is that because both S_w^2 and S_t^2 take in account the variations within the elements of the population they are approximately equal — which also can be implied from the fact that they have the same expected value σ^2. This means that S_w^2 can be used instead of S_t^2 which is a good way to avoid computational round-off errors.

2.4 Coefficients of Variation

Another procedure that provides measures of reliability of the various estimators is the concept of relative error. The coefficient of variation V is defined as the square root of the relative variance of the population V^2, which is given by:

$$V^2 = \frac{S^2}{\bar{x}^2} = \frac{1}{N-1}\sum_{i=1}^{N}\frac{(x_i - \bar{x})^2}{\bar{x}^2}. \tag{2.36}$$

Analogously, the relative variance of the variance is given by:

$$V_{S^2}^2 = \frac{\sigma_{\sigma^2}^2}{(\sigma^2)^2}. \tag{2.37}$$

Hansen, Hurwitz, and Madow [7] (see Appendix D) demonstrated that Equation 2.37 can be expressed as

$$V_{S^2}^2 = \frac{1}{N}\left(\beta - \frac{N-3}{N-1}\right), \tag{2.38}$$

where

$$\beta = \frac{\mu_4}{(\sigma^2)^2}, \tag{2.39}$$

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2, \tag{2.40}$$

$$\mu_4 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^4 \tag{2.41}$$

$$= \bar{x}_4 - 4\bar{x}\bar{x}_3 + 6\bar{x}^2\bar{x}_2 - 3\bar{x}^4, \tag{2.42}$$

and

$$\overline{x}_r = \frac{1}{N}\sum_{i=1}^{N} x_i^r. \tag{2.43}$$

For sufficiently large N, Equation 2.38 becomes:

$$V_{S^2}^2 = \frac{\beta - 1}{N}, \tag{2.44}$$

and for sufficiently large β

$$V_{S^2}^2 = \frac{\beta}{N}. \tag{2.45}$$

For I random groups of J elements, it can be shown that

$$V_{S^2}^2 = \frac{1}{I}\left(\beta_J - \frac{I-3}{I-1}\right), \tag{2.46}$$

where

$$\beta_J = \frac{\beta}{J} + 3\frac{J-1}{J}. \tag{2.47}$$

The derivations of Equations 2.46 and 2.47 can be found in reference [7] (see Appendix D).

The coefficient of variation of the standard deviation V_S is related with the coefficient of variation of the variance V_{S^2} by

$$V_S = \frac{V_{S^2}}{2}. \tag{2.48}$$

The proof of Equation 2.48 can also be found in reference [7] (see also Appendix D). A reasonable value for V_S is a subject for concern. The confidence limits for the standard deviation do not need to be the same as the confidence limits for the mean. Because V_S depends upon the kurtosis of the distribution, the significance of the confidence limits of the standard deviation varies for different distributions. A distribution with a large kurtosis is characterized by a large value for V_S which can be more reliable than an undersampled distribution with a small kurtosis and a much smaller V_S .

According to the central limit theorem, as J increases the distribution of the group means becomes more normal and β_J approaches the value of 3. If a normal distribution for the group means is assumed and if a 0.1 value for V_S is desired, at least 51 groups would be necessary as indicated by Equation 2.46. Therefore, to achieve a coefficient of variation of the standard deviation on the order of 0.1, a reasonable number of groups ($I > 50$) and a sufficiently large value of J are required so that the group means would be normally distributed. However, the results obtained with Equation 2.38 were consistently smaller than those obtained with Equation 2.46, which may be explained by the same argument that the expected value of the mean square within is smaller than the expected value of the mean square between. Because of this reasoning, Equation 2.38 is used instead of Equation 2.46. Also, it is shown in Chapter 4 that to achieve a stable value of β, a sample size larger than the one necessary to achieve a sufficiently reliable estimate of S^2 is required.

2.5 The Figure of Merit

The figure of merit $\sigma_{\bar{x}}^2 T$ is widely accepted as a measure of the calculational efficiency and also as a indicator that the solution has achieved the asymptotic $1/\sqrt{N}$ behavior as predicted by the central limit theorem. The figure of merit can also be expressed as

$$FOM = \frac{1}{\sigma_{\bar{x}}^2 T}, \tag{2.49}$$

where $\sigma_{\bar{x}}^2$ is the variance of the mean and T is the total computation time. A larger FOM indicates a more efficient calculation.

Considering t as the average computation time per particle, Equation 2.49 can be rewritten as

$$FOM = \frac{N}{\sigma^2 N t} = \frac{1}{\sigma^2 t}. \tag{2.50}$$

Therefore, when σ^2 becomes constant, i.e. a sufficient number of particles have been sampled, the figure of merit also becomes constant. However, for highly skewed distributions with a small proportion of very large contributions, the behavior of $\sigma_{\bar{x}}^2$ is typically not $1/\sqrt{N}$ and may assume either faster or slower rates of convergence. The figure of merit will become stable only after a sufficient number of particles have been sampled. Also, if undersampling is severe, the figure of merit will appear essentially constant over a wide range of sample sizes thus giving false indications about the solution.

2.6 Sampling of Rare Events

In deep penetration particle transport problems, the utilization of some form of importance sampling is necessary. For example, an analog solution of a deep penetration problem with a transmission factor of 10^{-12} will require an average of 10^{12} particles to score a success. It is obvious that if a reasonable number of successes are not scored, the results will be worthless. If the parameter being estimated is already known, the sampling scheme can be modified so that every sample yields the same contribution – which results in what is called a zero-variance calculation. It is possible to use any previously known information about the population distribution to enhance the probability of scoring. This procedure is commonly referred to as importance sampling.

Importance sampling may be accomplished by variance reduction techniques such as Russian-roullete, splitting, survival biasing, stratified sampling, weight cut-offs, exponential transform etc. The variance can also be reduced by using estimators or samplers such as the last-flight and the next-event estimators. The utilization of importance sampling requires the use of weight corrections so that the bias introduced by the sampling scheme is eliminated thereby preserving the fair game. However, some techniques such as the point-detector estimator introduce some bias that is not properly corrected by weight correction alone [8]. Importance sampling schemes in general will result in reliable (unbiased) estimates if a sufficiently large number of particles are sampled and provided of course that all sources of bias are corrected. It is also true that improper utilization of importance sampling techniques can result in a calculation that is less efficient.

A counter was introduced into the MORSE code that records the number of particles that have the value of their contributions within a given range or channel. Therefore, it is possible to know how many particles account for specific fractions of the total response. This counter is collapsed from 188 intervals to 10 percentage intervals which would indicate if a small number of particles are responsible for a large percentage of the total value of the response. This provision turn out to be very useful in the analysis and identification of the undersampled condition because of its ability to isolate the contribution background.

Chapter 3

Modifications Introduced into the MORSE Code

Modifications were introduced into the MORSE code in order to calculate the statistics on a per particle basis according to equations 2.4 and 2.13. The original MORSE input was maintained and the output of the fractional standard deviations for the total responses calculated on a batch basis with Equation 2.12 was also retained. Another feature introduced was a table that shows in terms of percentage bins the distribution of all particle contributions according to their increasing values. This table is created from a 188 multichannel-type histogram as shown in Figure 3.1. This histogram is shown in a scale which enhances the importance of particles in the higher channels. Figure 3.2 shows this table in the output of the same batch that generated the histogram in Figure 3.1. Analyzing Detector 2 in Figure 3.2, 69700 particles account for the first 20% of the total response, 14634 particles account for the next 10% fraction of the response, and so on until the 5 particles of highest weight which account for the last 10% of the total response.

Besides the addition of the histogram mentioned above, this new version of the MORSE code also includes the output of the total responses and their standard deviations for each batch and for the accumulated estimates. The output also presents the estimates of the coefficient of variation and figure of merit, based on the accumulated statistics after each batch is completed.

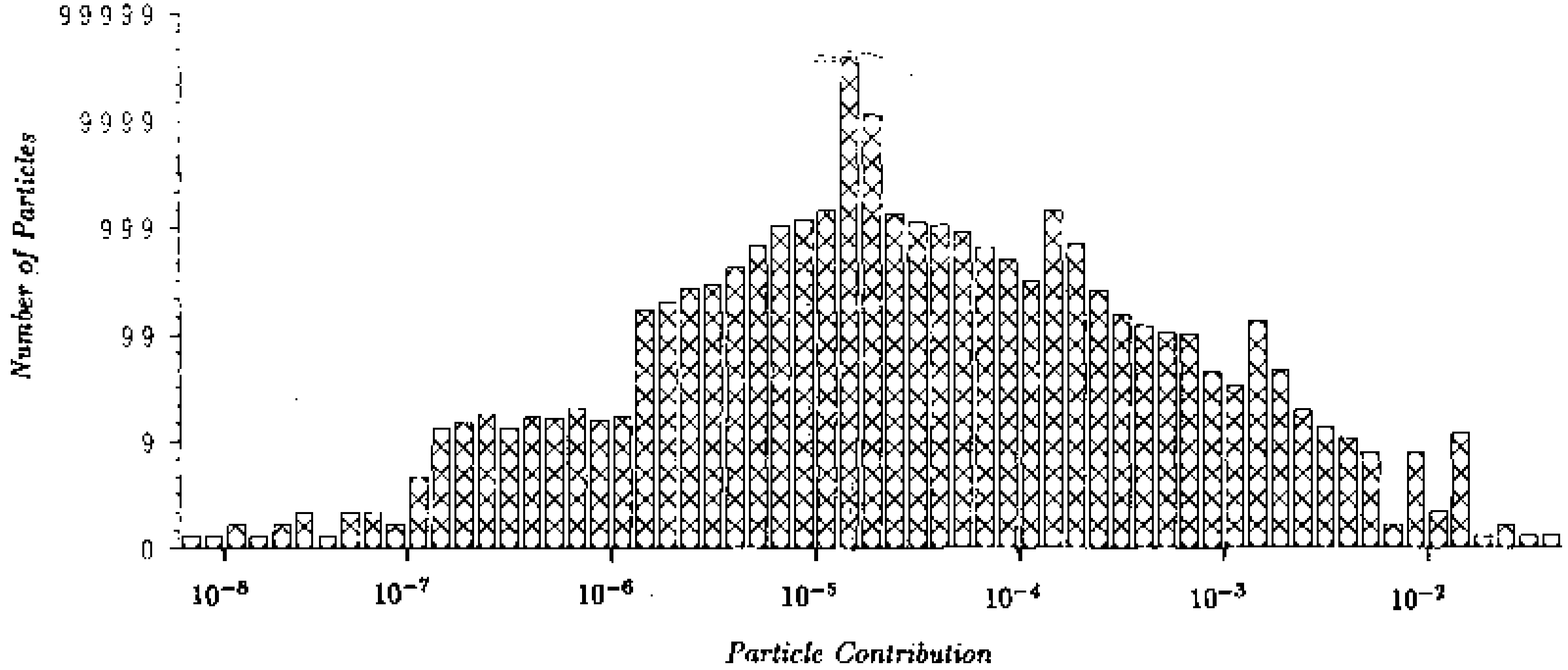

Figure 3.1: Particle Contribution Histogram for Detector 2 - Batch 1.

```
***START BATCH    1                                   RANDOM-13579BDFDB97

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF SOURCE WHICH SETS WATE TO DDF AND PROVIDES AN ENERGY IG.
YOU ARE USING THE DEFAULT VERSION OF GTMED WHICH ASSUMES GEOMETRY AND XSECT MEDIA ARE IDENTICAL.
  WTAVE        IAVE    UAVE    VAVE     WAVE        XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05      1.00  -0.0022 -0.0005  0.0018    0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                 RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR            BATCH            FSD          ACCUMULATED          FSD          CV(SD)          FOM
                      RESPONSE                         RESPONSE
       1             3.8195D-04       0.02420        3.8195D-04        0.02420      0.18380      9.6140D+01
       2             4.0657D-05       0.03220        4.0657D-05        0.03220      0.17042      5.4278D+01
       3             1.0318D-05       0.18977        1.0318D-05        0.18977      0.45750      1.5631D+00
       4             2.2010D-06       0.10484        2.2010D-06        0.10484      0.24872      5.1215D+00
       5             5.8018D-07       0.16399        5.8018D-07        0.16399      0.24418      2.0932D+00
       6             1.3670D-07       0.13992        1.3670D-07        0.13992      0.20058      2.8754D+00
       7             1.0456D-07       0.65115        1.0456D-07        0.65115      0.49116      1.3276D-01
       8             3.6075D-08       0.39992        3.6075D-08        0.39992      0.31527      3.5196D-01
       9             4.4715D-09       0.28794        4.4715D-09        0.28794      0.26414      6.7894D-01
      10             8.1909D-10       0.22073        8.1909D-10        0.22073      0.24052      1.1553D+00

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0     100.0

       1       82140         0         0         0      6767      7358       717       735       251        49
       2       69700         0     14634      7215      3000      1316       971        68       103         5
       3       78956     13447      1962      1753       289       144        23        13         1         0
       4       81208     12320      2213       405       198        43        21         6         5         0
       5       90177      4818      1100       165        76        18         7         3         1         1
       6       88529      6765       885       246        37        34         2         4         2         0
       7       96335       288        15         3         1         0         0         0         0         0
       8       96700        30         0         1         1         1         0         0         0         0
       9       96124       619        63        10         0         1         2         0         1         0
      10       96140       703        86        21         0         3         2         0         1         0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)    FISHN   GAMGEN REALCOLL    ALBEDO    BDRYX   ESCAPE    E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000       0         0        0   188986         0    98169       48    99952        0        0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 45 SECONDS.
```

Figure 3.2: Present MORSE Output after a Batch is Completed.

LOCSD	1	SD	Standard MORSE per batch statistics
	2	SSD	
	3	SSD2	
	4	SD	New MORSE per particle batch statistics
	5	SSD	
	6	SSD2	
	7	SSD	Accumulated statistics
	8	SSD2	
	9	SSD	Working area
	10	SSD2	
	11	SSD3	Statistics used in the calculation of the kurtosis
	12	SSD4	
	13	SSD5	
	14	SSD6	
LOCQE	15	SUD and SSD units	

Figure 3.3: Layout of the New Storage Areas in the Blank Common.

To perform the necessary additional bookkeeping the *user* part of the *blank common* was separated, augmented, and put in double precision (COMMON /SSS/). The particle bank now can accept only one source particle and any number of secondaries particles (NMOST). Therefore, the memory requirements are substantially reduced and the utilization of double precision arithmetic was required to work with numbers of particles on the order of several millions. Figure 3.3 shows the lay out of the new storage areas.

In Appendix C, a listing of all added and modified subroutines is included. All major characteristics of these subroutines are presented in the following list:

1. FLUXST – modified to perform additional bookkeeping in COMMON /SSS/.

2. FTEST – calculates the F values.

3. GSTORE – modified to include only one source particle in the particle bank and at most (NMOST) secondary particles.

4. INPUT1 – minor modification in the input format of CARD B.

5. MAIN – modified to include COMMON /SSS/.

6. MORSE – modified to perform additional calculations for the per particle analysis.

7. MSOUR – modified to help the interim batch information output.

8. NBATCH – now performs only the total response in the per batch method.

9. NPART – called at the end of each history to do the sums needed in the per particle analysis.

10. NRNPRI – perform all calculations in the per particle basis and outputs the new batch information.

11. NRUN – modified to output the differential responses in the per particle basis.

12. OUTPT – modified to accomplish the interim batch information output.

13. RELCOL – this subroutine was modified according to Appendix B and it is shown only because of this purpose.

14. SCORIN – modified to create the new bookkeeping area in COMMON /SSS/.

15. STBTCH – now performs only the zeroing of the per batch bookkeeping areas.

16. STPART – provides the zeroing of the proper bookkeeping areas of the per particle analysis.

17. TESTW – modified in order to handle the the new allocation of secondary particles.

18. VAR2 – modified to include the per particle estimate of the FSD.

19. VAR3 – modified to perform only estimates of the FSD in the per particle basis.

20. VAR4 – provides the estimates of the FSD and of the $CV(SD)$.

Chapter 4

Analysis of Results

At the beginning of this work it was necessary to identify a problem which has a solution with an underestimated mean and a small standard deviation. The search for such a problem yielded what became the main objective of this work — which was to provide the user with more information, and thereby to better understand the behavior of the solution and the associated sampling distribution. This information will be the key for a successful calculation.

Because of the extensive amounts of output generated, the solutions to most of the sample problems were put in Appendix A. The reader can more easily refer to these data for more refined comparisons.

4.1 Sample Problem 1

Sample problem 1 consists of a 1-meter radius concrete sphere with 10 detectors positioned at 10-cm intervals from the center of the sphere. An isotropic monoenergetic (14-Mev) neutron point source is located at the center of the sphere. All responses are for the first group only. Table 4.1 shows the group structure for the cross sections.

This problem offers several degrees of difficulty for the effects of interest calculated. For example, the response of Detector 1 is very easy to calculate as compared with that of Detector 10. Also, the utilization of point detectors makes the flux estimation of Detector 10 very difficult to accomplish because of the three-dimensional nature of point detectors.

Table 4.1: Energy Structure for Sample Problem 1.

Group	Upper Energy	Group	Upper Energy	Group	Upper Energy
1	1.50E+07	2	1.22E+07	3	1.00E+07
4	8.18E+06	5	6.36E+06	6	4.96E+06
7	4.06E+06	8	3.01E+06	9	2.46E+06
10	2.35E+06	11	1.83E+06	12	1.11E+06
13	5.50E+05	14	1.11E+05	15	3.35E+03
16	5.83E+02	17	1.01E+02	18	2.90E+01
19	1.07E+01	20	3.06E+00	21	1.12E+00
22	4.14E-01				

4.1.1 Next-Event Surface Crossing Estimator Solution

The utilization of a next-event surface crossing estimator (NESXE) posed no problems in the solution of sample problem 1. Figures 4.1, 4.2, and 4.3 show the evolution of the flux, FSD, and FOM with increasing number of particles for detectors 2, 6, and 10 respectively. It is useful to observe the asymtoptic behavior of these statistics over a wide range of sample sizes. In this problem the solutions are free from undersampling because it was possible to process a sufficient number of particles and the statistical quality of these solutions are described by the central limit theorem. There is no reason to question the statistics calculated in this problem. Even FSD's much higher than 0.5 are likely to yield fluxes within 3σ of the expected values most of the time.

An important fact suggested from Figure 4.1 and from the data in Appendix A.1.1 is that the coefficient of variation of the standard deviation $CV(SD)$ generally follows but is almost always higher than the standard deviation. Also, the amount by which the $CV(SD)$ changes is often much more peaked than that experienced by the standard deviation because of the changes in the kurtosis of the distribution. An increase in the $CV(SD)$ may reflect either a small increase or even a decrease in the standard deviation. Another interesting aspect of these results is that the FOM accepts the solution of Detector 10 much earlier than the solution of Detector 2 which indicates that this parameter may cause the rejection of perfectly acceptable solutions.

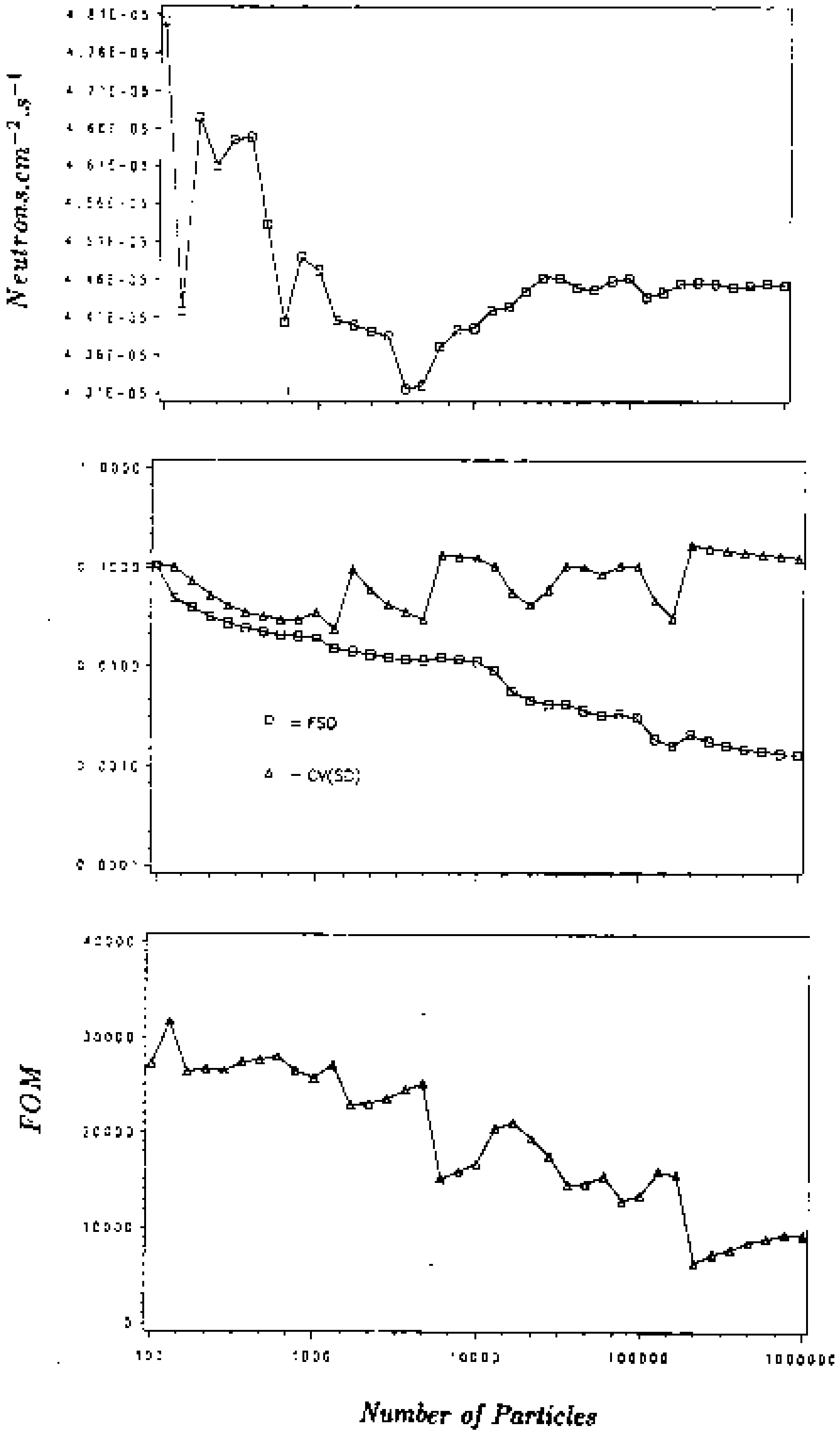

Figure 4.1: Neutron Flux, FSD, $CV(SD)$, and FOM Behavior for Detector 2 – NESXE.

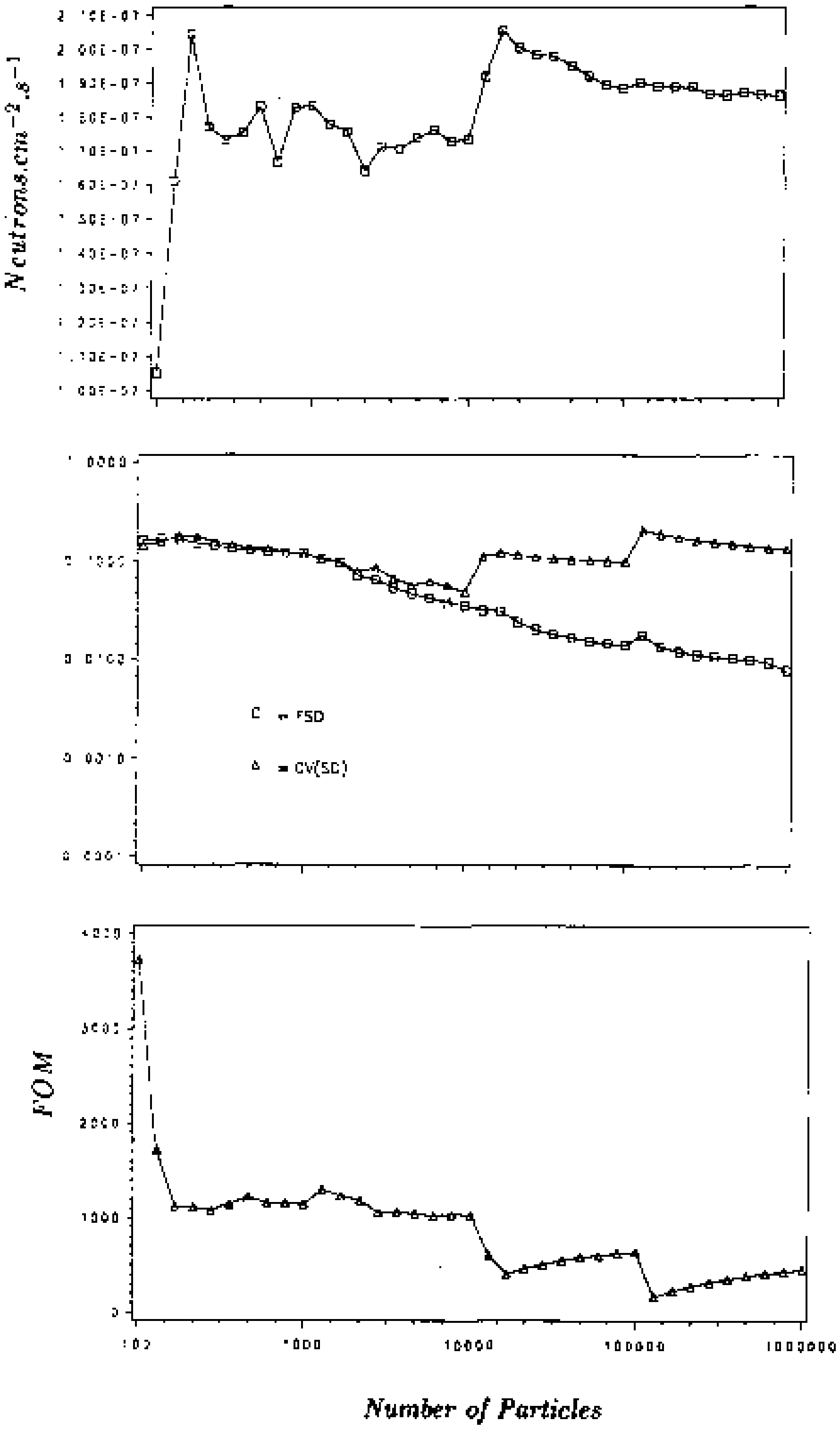

Figure 4.2: Neutron Flux, FSD, $CV(SD)$, and FOM Behavior for Detector 6 – NESXE.

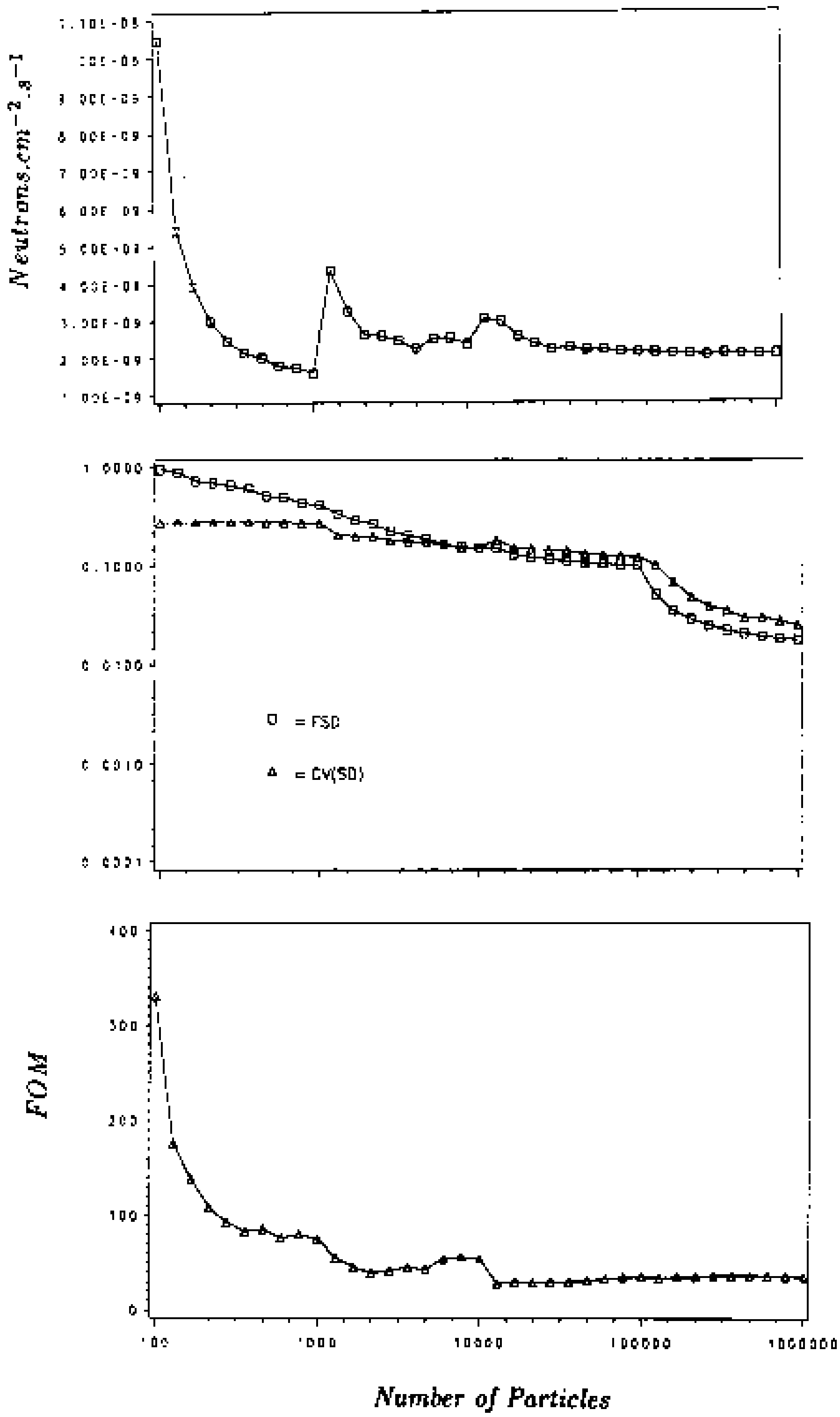

Figure 4.3: Neutron Flux,FSD, $CV(SD)$, and FOM Behavior for Detector 10 – NESXE.

Table 4.2 shows the results for three phases of the calculation. The three phases consisted of 10 batches each with 10^3, 10^4, and 10^5 particles per batch respectively. The F tests present some high values that indicate major differences between the group means. For example, the F value is 1.99945 for Detector 6 in the range of 10^4 to 10^5 particles. It is possible to see in Figure 4.2 that effectively there are large differences between the group means. However, it should be noted that the solution has converged to the true solution and the standard deviation is also sufficiently small. In the case of Detector 7 in the range of 10^5 to 10^6 particles the standard deviation is definitely small enough to guarantee the results — which means that this parameter can also reject perfectly acceptable solutions.

4.1.2 Point-Detector Estimator Solution

The utilization of point-detector estimators (PDE) significantly increases the degree of difficulty in the solution of this problem. In fact, the effects of undersampling can be easily seen when the point detector solutions are compared with the solutions described in Section 4.1.1.

Figures 4.4, 4.5, and 4.6 show the behavior of the neutron flux, FSD and $CV(SD)$, and the figure of merit with increasing values for the sample size for detectors 2, 6, and 10 respectively. In the case of Detector 2 the undersampling condition yielded underestimation for sample sizes up to 200,000 particles and for Detector 6 for sample sizes in the range of 2,000 to 500,000 particles. For both detectors the FOM exhibits stable plateaus in the undersampled region. In the case of Detector 10, the simulation is so undersampled that is possible to see the peaks in Figure 4.6 which are caused by contributions from individual particles. It is interesting to note from Appendix A.1.2 that the particle contribution distribution for Detector 6 contained more particles in the higher channels in batch 5 than in batch 10. This is an effect of the sampling of a few important particles that shifted the distribution and minimized the undersampling condition.

Table 4.2: F Values for Sample Problem 1 with Next-Event Estimator.

Number of Particles	Detector	Total Response	*FSD* Batch	*FSD* Accum.	F Value
$10^3 - 10^4$	1	3.9502D-04	0.01518	0.02069	0.53761
	2	4.3966D-05	0.01400	0.01480	0.89466
	3	8.6957D-06	0.01311	0.02216	0.34984
	4	2.0963D-06	0.03622	0.03055	1.40545
	5	6.2572D-07	0.02918	0.05013	0.33860
	6	1.7343D-07	0.04160	0.05967	0.48560
	7	5.7541D-08	0.11905	0.09939	1.43532
	8	2.0712D-08	0.12643	0.13236	0.91234
	9	8.9238D-09	0.26682	0.23243	1.31807
	10	2.3464D-09	0.27192	0.25963	1.09688
$10^4 - 10^5$	1	3.8398D-04	0.00394	0.00327	1.45084
	2	4.4619D-05	0.00422	0.00526	0.64334
	3	8.9053D-06	0.00626	0.00679	0.84945
	4	2.2075D-06	0.01124	0.01134	0.98125
	5	6.2544D-07	0.00921	0.01497	0.37831
	6	1.8826D-07	0.03401	0.02405	1.99945
	7	6.0974D-08	0.02165	0.03847	0.31681
	8	2.0435D-08	0.04259	0.04535	0.88189
	9	6.3437D-09	0.05786	0.06413	0.81423
	10	2.1429D-09	0.11009	0.10191	1.16697
$10^5 - 10^6$	1	3.8319D-04	0.00079	0.00122	0.42173
	2	4.4523D-05	0.00168	0.00200	0.69934
	3	8.9761D-06	0.00289	0.00343	0.70744
	4	2.2398D-06	0.00626	0.00637	0.96701
	5	6.2468D-07	0.00535	0.00572	0.87482
	6	1.8613D-07	0.00883	0.00907	0.94718
	7	5.8928D-08	0.01453	0.01125	1.66750
	8	1.9024D-08	0.01464	0.01860	0.62002
	9	6.0442D-09	0.01903	0.02137	0.79305
	10	2.0664D-09	0.02113	0.03236	0.42638

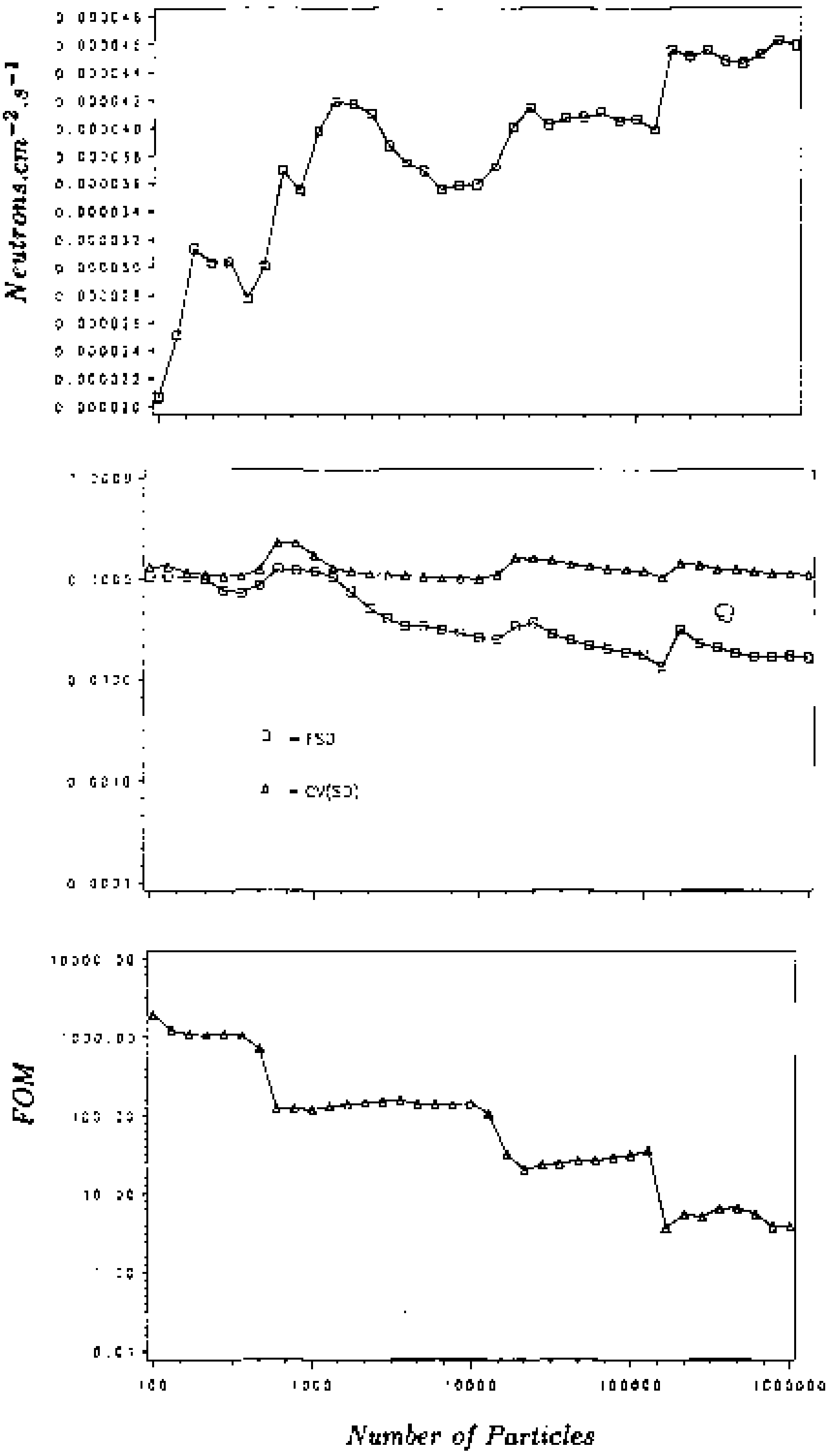

Figure 4.4: Neutron Flux, FSD, $CV(SD)$, and FOM Behavior for Detector 2 – PDE.

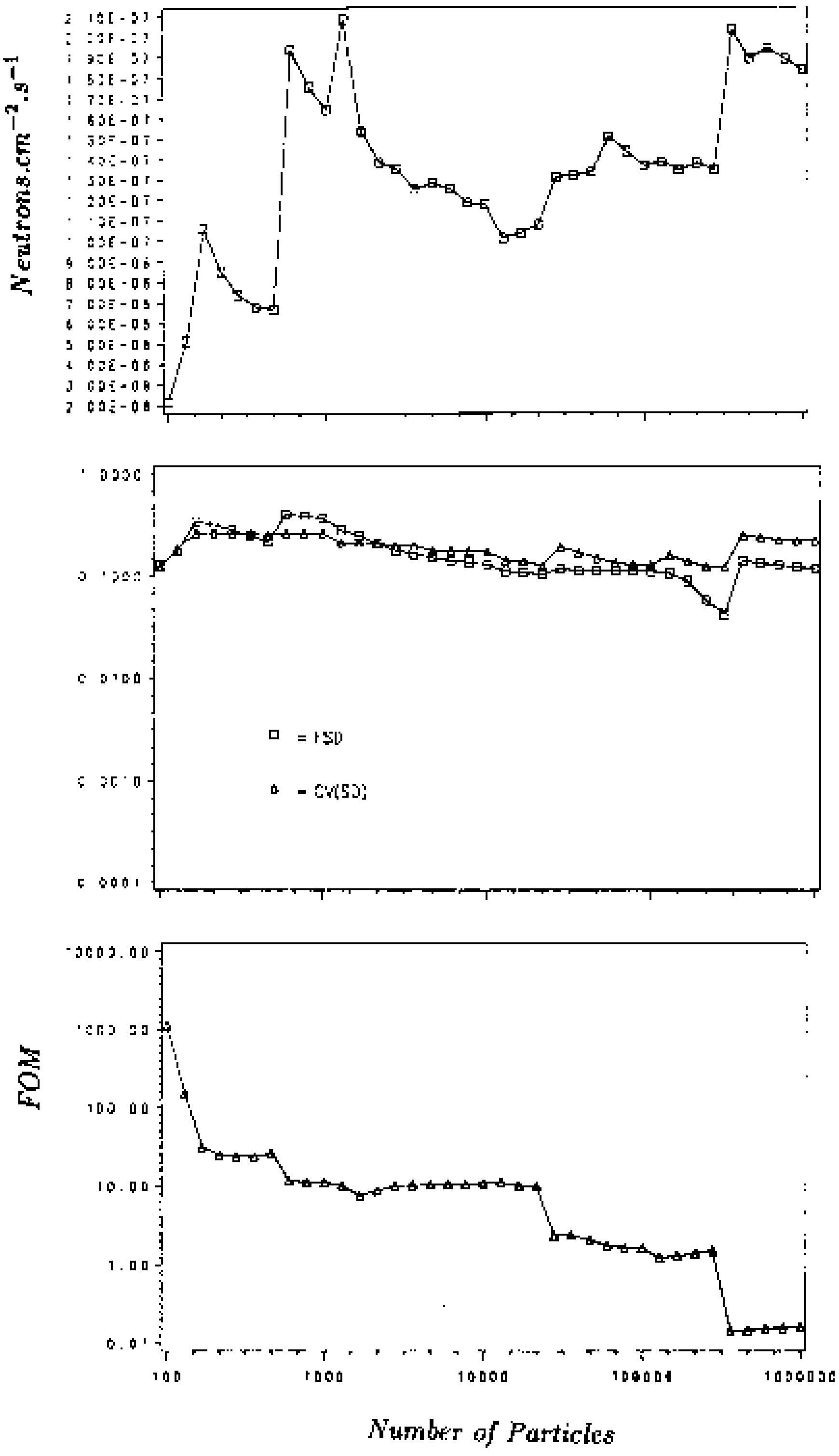

Figure 4.5: Neutron Flux, FSD, $CV(SD)$, and FOM Behavior for Detector 6 – PDE.

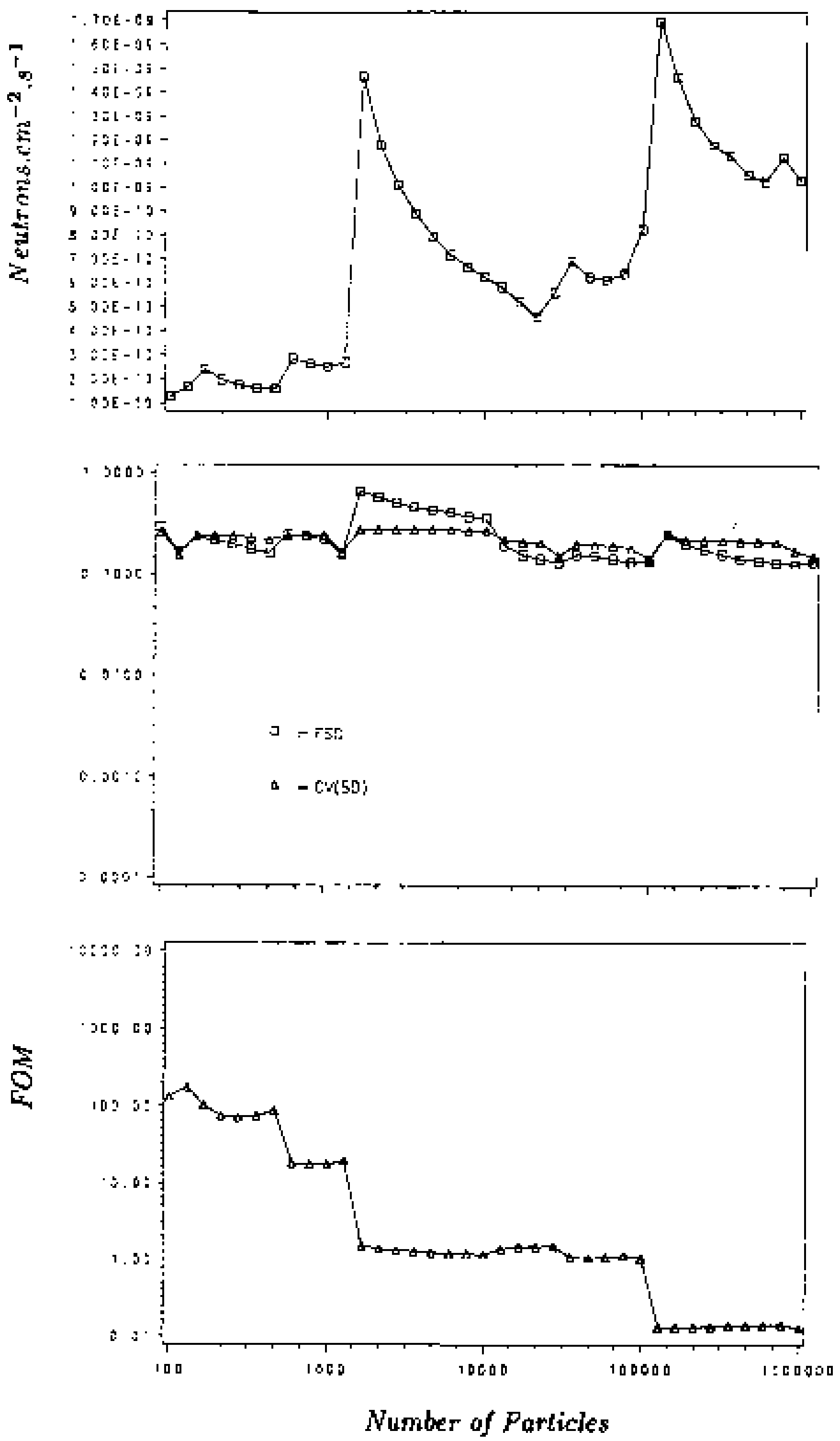

Figure 4.6: Neutron Flux, FSD, $CV(SD)$, and FOM Behavior for Detector 10 - PDE.

Appendix A.1.2 shows the output for 10 batches of 10^5 particles each. For the first batch, detectors 1 and 2 are clearly acceptable solutions on the basis of the behaviors of their standard deviations alone. Detectors 3 and 4 present peaks in their standard deviation and $CV(SD)$ and are very close to the true solutions[1] although they exhibit somewhat large standard deviations. The distribution of particle contributions for all detectors beyond Detector 2, shows that the number of particles accounting for large fractions of the solutions is becoming too small.

In Batch 4, the peaks in the standard deviation and $CV(SD)$ shifted to Detector 5. The distribution of particle contributions for this detector shows a relatively small number of particles accounting for a large fraction of the response which usually characterizes the undersampling condition. Finally, in Batch 10 the peak shifted to Detector 6 — again, a relatively small number of particles account for a large fraction of the response.

Based on these results and the discussion above, the solutions for detectors 1, 2, 3, 4 and 5 can be considered reliable. Detector 6 represents the beginning of what will be called the *breakdown region*, which is associated with a sample size that provides either only a few or no important rare event making the standard deviations – which can be very small - experience large increases. Obviously, all detectors beyond this point will be undersampled unless additional particles are processed. It is interesting to observe the behavior of the solution at Detector 7 which initially overestimates the true solution because of a highly contributing particle that was sampled early in the calculation.

An important fact, as seen in Appendix A.1.2, is that the particle contribution histogram always included a small number of particles which accounted for large fractions of the solutions for the undersampled detectors. Therefore, the histogram is able to detect the undersampling condition for both the breakdown region and the region of severe undersampling. In this problem, the explanation for this behavior is that even in the severe undersampling condition, there are some particles that make contributions only large

---

[1] True solutions are considered the solutions obtained in Section 4.1.1 with 10^6 particles.

enough to make them stand out from the background contributions, and as the sample size increases these contributions will eventually move to the intermediary channels.

Figure 4.7 shows a three-dimensional view of the standard deviation and of the coefficient of variation of the standard deviation. It can be seen that the plot of the coefficient of variation is much more structured than that of the standard deviation providing a qualitative rather than quantitative indicator of the behavior of the solution.

Figure 4.8 shows the same calculation as in Figure 4.7 but with the last random number of that calculation used as the initial one in this calculation. A FSD comparison shows good agreement up to Detector 6 and a $CV(SD)$ comparison shows good agreement only up to Detector 3 which demonstrates that this indicator is too sensitive to be taken as a quantitative measure of precision. Also, from figures 4.7 and 4.8, it can be assumed that Detector 10 represents the end of the breakdown region or the beginning of the region of severe undersampling. Figure 4.9 shows the calculations using the exponential transform technique with PATH = 0.5 and Figure 4.10 shows the results using source angular biasing, Russian roulette, splitting, and exponential transform. It is possible to see that the variance reduction techniques significantly reduce the variability of $CV(SD)$ making it less structured. Because this behavior may indicate the effectiveness of a given variance reduction technique and also that the sampling distribution has become sufficiently stable, it was concluded that the use of this coefficient needs further study.

Table 4.3 shows the calculated F values after 10 batches are completed. The two values that flag greater differences in the batch means are for Detector 2 and Detector 6. These results illustrate the weakness of the test in discerning acceptable from unacceptable solutions. The test detected that there are significant differences in the group means of Detector 6. Therefore, the results should be checked with other indicators to determine if they are acceptable. Otherwise it is necessary to increase the batch size.

In the case of Detector 2 the explanation can be based on the fact that although it is unlikely, there are values in the tails of the normal distribution. And the null hypothesis

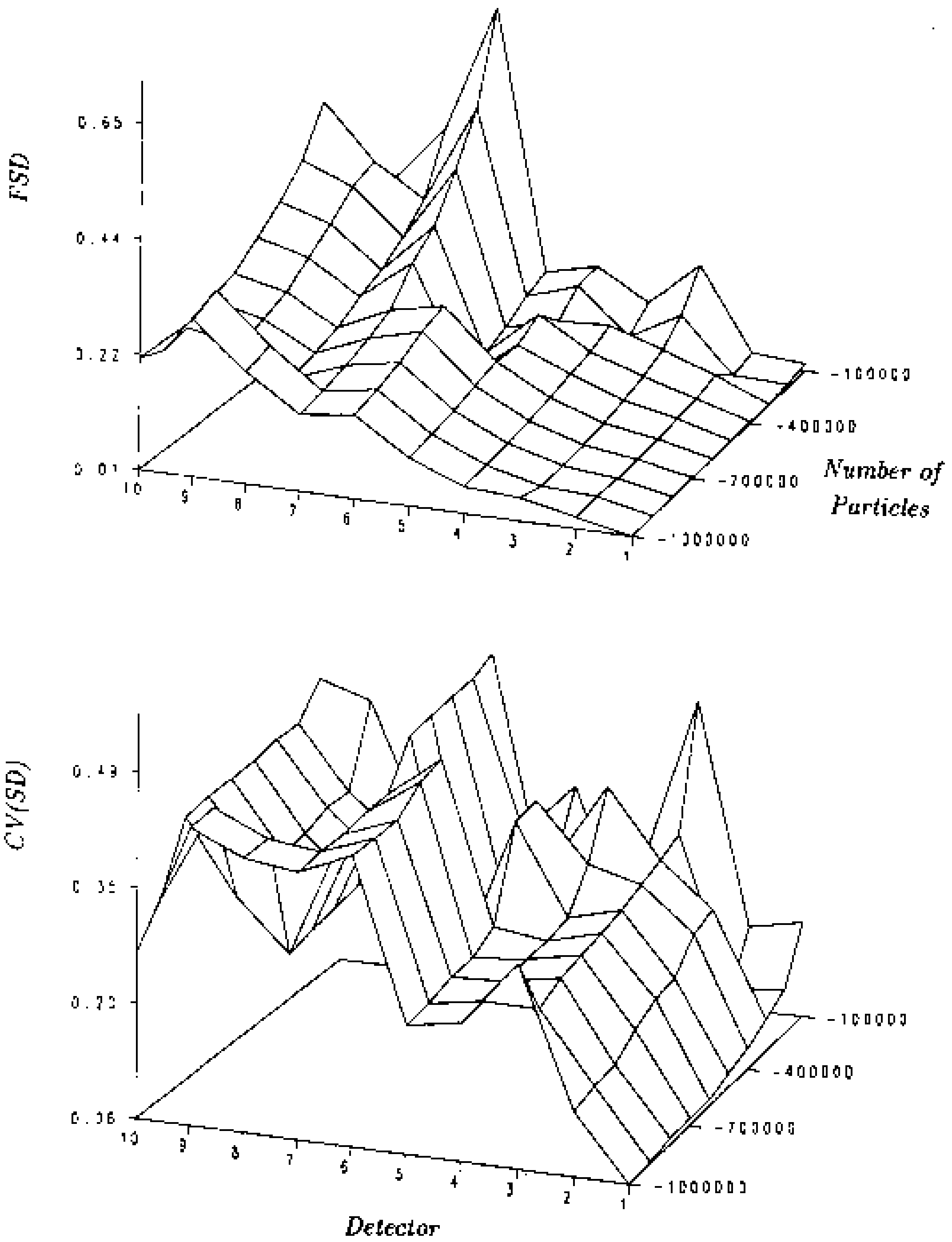

Figure 4.7: Three-Dimensional Behavior of FSD and $CV(SD)$ for Sample Problem 1 – PDE.

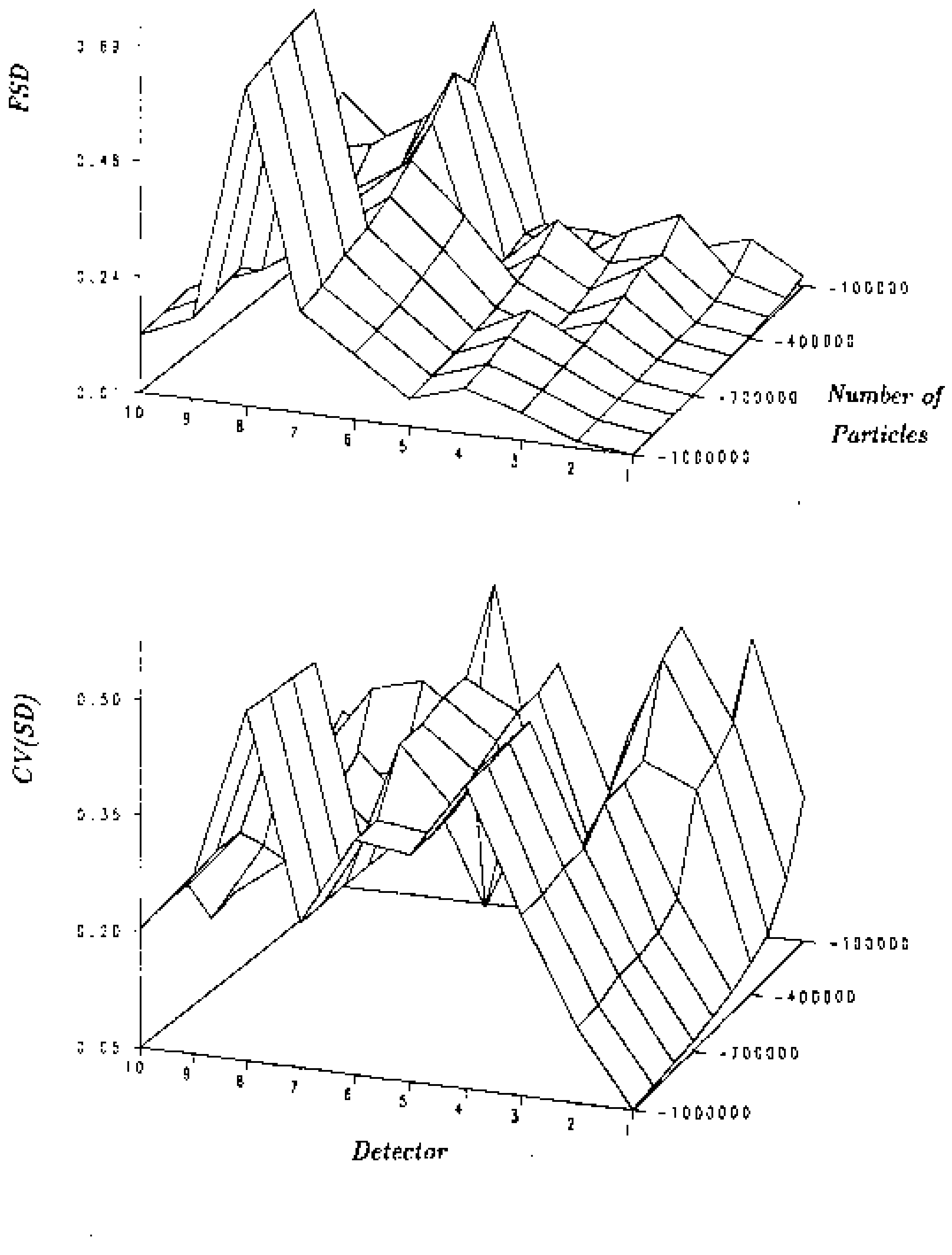

Figure 4.8: Behavior of FSD and $CV(SD)$ for Sample Problem 1 -- PDE with New Starting Random Number.

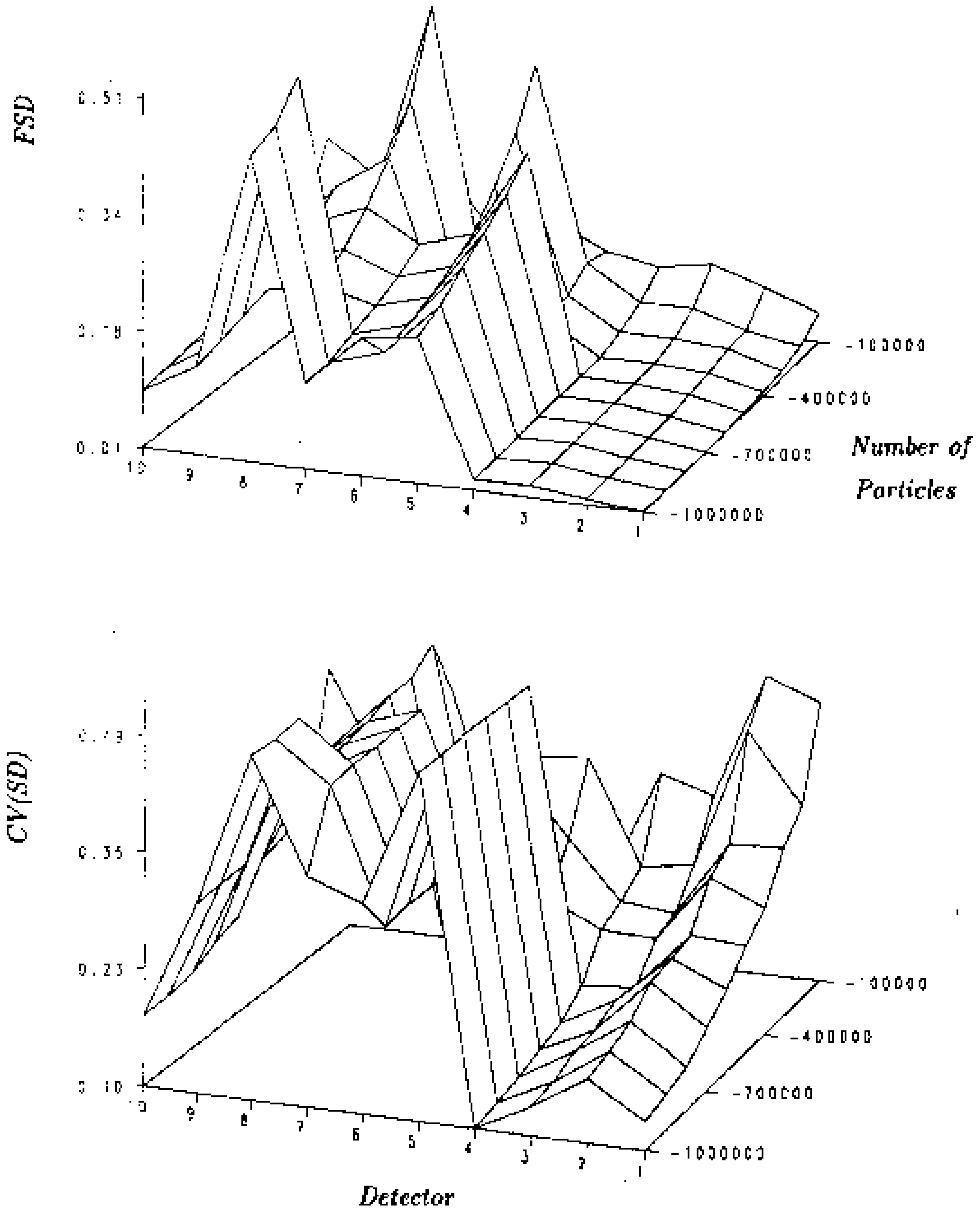

Figure 4.9: Behavior of FSD and $CV(SD)$ for Sample Problem 1 – with PATH = 0.5.

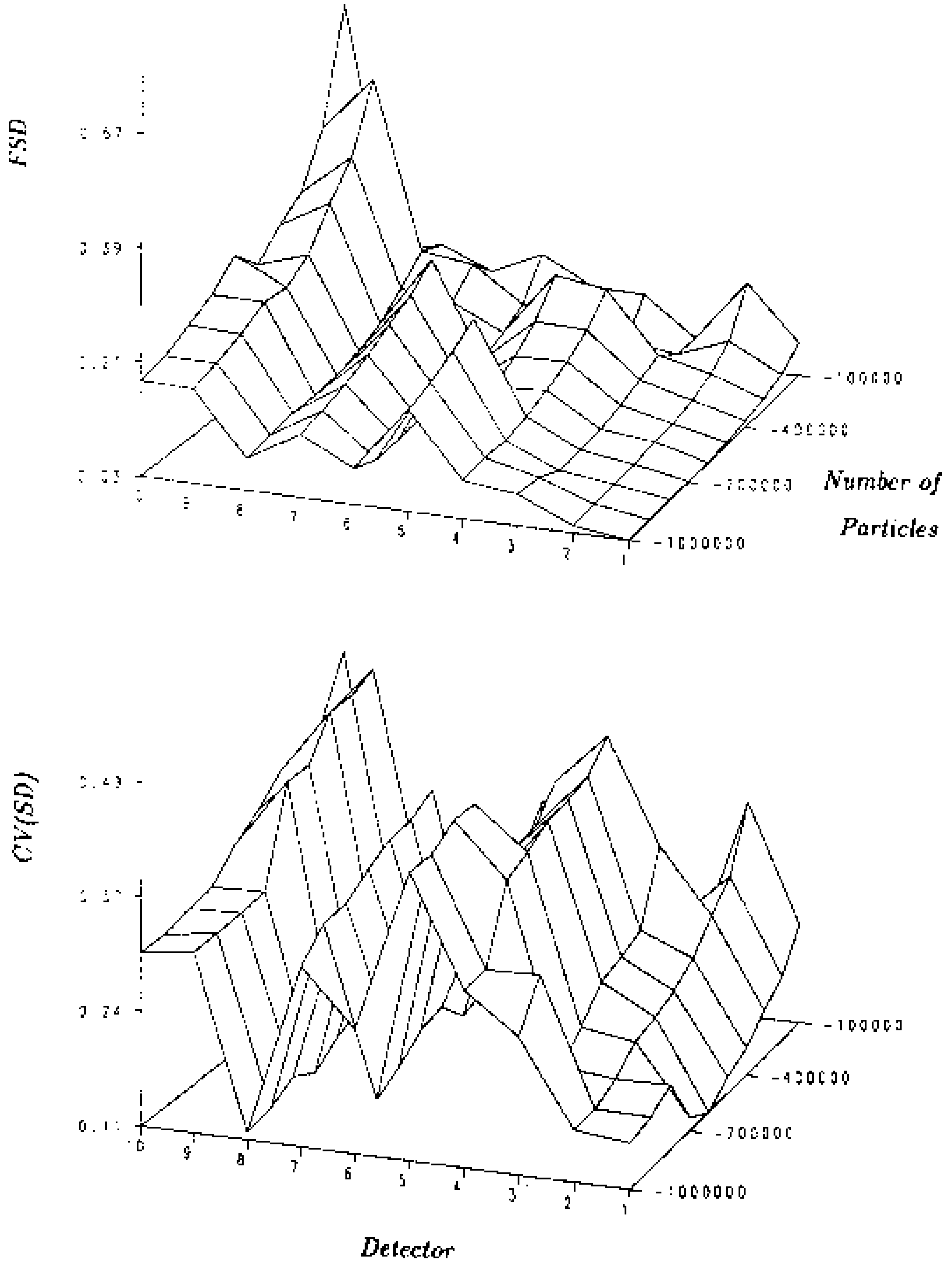

Figure 4.10: Behavior of FSD and $CV(SD)$ for Sample Problem 1 – using Russian Roulette, Splitting, and PATH = 0.5.

Table 4.3: F Values for Sample Problem 1 with Point-Detector Estimator.

Number of Particles	Detector	Total Response	*FSD* Batch	*FSD* Accum.	F Value
$10^3 - 10^4$	1	3.5122D-04	0.02451	0.03081	0.63240
	2	3.5908D-05	0.05013	0.04783	1.09872
	3	1.4941D-05	0.30124	0.30203	0.99466
	4	3.4210D-06	0.38488	0.39837	0.93329
	5	1.0802D-06	0.56001	0.57999	0.93216
	6	1.1766D-07	0.16116	0.20402	0.62366
	7	3.4278D-08	0.34404	0.38522	0.79743
	8	1.5081D-08	0.68582	0.69850	0.96390
	9	2.6313D-09	0.59333	0.60736	0.95422
	10	6.2342D-10	0.57851	0.59306	0.95140
$10^4 - 10^5$	1	3.8195D-04	0.01798	0.02420	0.55218
	2	4.0657D-05	0.03121	0.03220	0.93898
	3	1.0318D-05	0.19179	0.18977	1.02138
	4	2.2010D-06	0.12374	0.10484	1.39315
	5	5.8018D-07	0.16108	0.16399	0.96486
	6	1.3670D-07	0.14838	0.13992	1.12463
	7	1.0456D-07	0.71689	0.65115	1.21213
	8	3.6075D-08	0.48367	0.39992	1.46273
	9	4.4715D-09	0.30049	0.28794	1.08911
	10	8.1909D-10	0.26011	0.22073	1.38867
$10^5 - 10^6$	1	3.9006D-04	0.00957	0.00912	1.10126
	2	4.5916D-05	0.04192	0.02979	1.97938
	3	9.0757D-06	0.04628	0.05148	0.80815
	4	2.1204D-06	0.05785	0.05826	0.98588
	5	7.0483D-07	0.09354	0.09736	0.92293
	6	1.8347D-07	0.22707	0.16347	1.92943
	7	5.0838D-08	0.16119	0.15096	1.14010
	8	1.7408D-08	0.20189	0.21587	0.87468
	9	5.5855D-09	0.26034	0.29830	0.76171
	10	1.0233D-09	0.22022	0.21594	1.04000

actually claims a uniform distribution for the group means. For the last four detectors which are undersampled and exhibit some underestimation, the test fails to show any differences between the means because in the undersampling regime the group means are likely to be very close to each other. As the sample size increases and some heavy contributing particles are sampled larger differences in the group means are observed. Also, the application of F tests as mentioned in Chapter 2 requires that the sampling distribution should be normally distributed. If the kurtosis is too high the tests will tend to be too small [6] and the hypothesis of equality of the means can not be tested.

4.2 Sample Problem 2

This problem belongs to a set of sample problems that comes with the MORSE code package. It comprises of a point fission source in air and uses a surface crossing estimator in the analysis. Appendix A.2 shows the output for the problem using variance reduction techniques with the same parameters as in the MORSE manual. It is interesting to note that most of the $CV(SD)$ values are well within commonly acceptable values. Also, the distribution of particle contributions exhibits a more uniform dispersion of particles in the percentage bins.

Figure 4.11 shows the behavior of both FSD and $CV(SD)$. This figure also show that although the kurtosis of the distribution eventually becomes constant, that happens long after an acceptable solution has been achieved.

4.3 Sample Problem 3

This problem was solved to demonstrate the behavior of the solutions in a more complex geometry. The configuration consists of a concrete cylinder of 150.2-cm height and a 150.0-cm radius with a cylindrical duct of 7.62-cm radius placed along the main axis. A 14-Mev neutron source is positioned along the bottom side of the cylinder and emits neutrons with

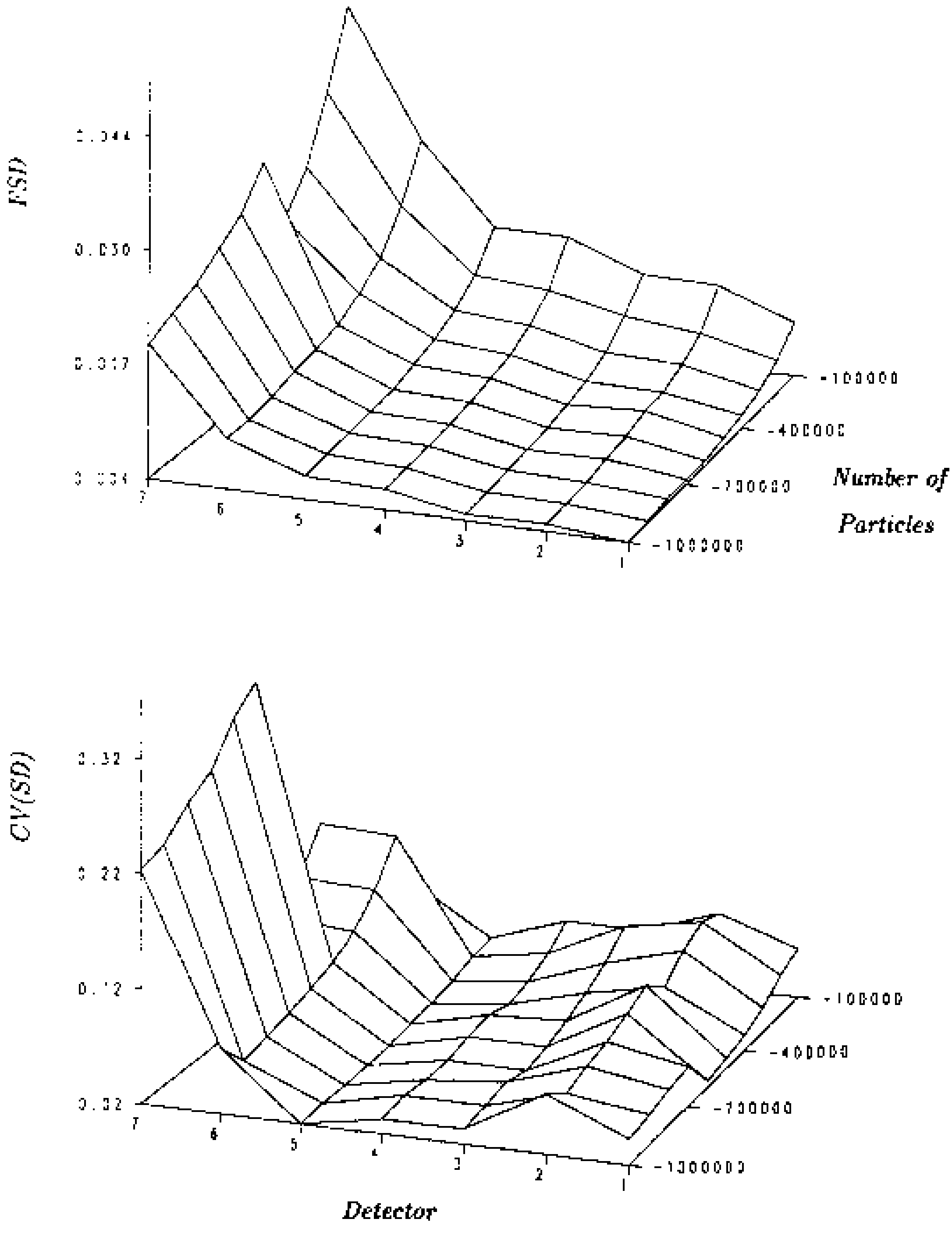

Figure 4.11: Three-Dimensional Behavior of FSD and $CV(SD)$ for Sample Problem 2.

directions uniformly distributed in the hemisphere facing the cylinder. Also, only the top fourteen groups are analyzed.

Because of the characteristics of the problem, source biasing provides large variance reductions which is accomplished using a step importance function for the position of emission which samples particles within a 10-cm radius 1000 times more often than outside this radius.

Appendix A.3.1 shows the data without source biasing and Appendix A.3.2 with source biasing. The effect of the source biasing is to decrease the size of the background contributions. It is interesting to see the behavior of the FOM in this problem. Figure 4.12 shows that although the step biasing procedure increased the efficiency of the calculation, the figure of merit of the problem without step biasing has a much more constant structure.

Figure 4.12 also shows that an increase in efficiency in the calculation may not reflect a smaller $CV(SD)$ which again presents very little information about the behavior of the solution.

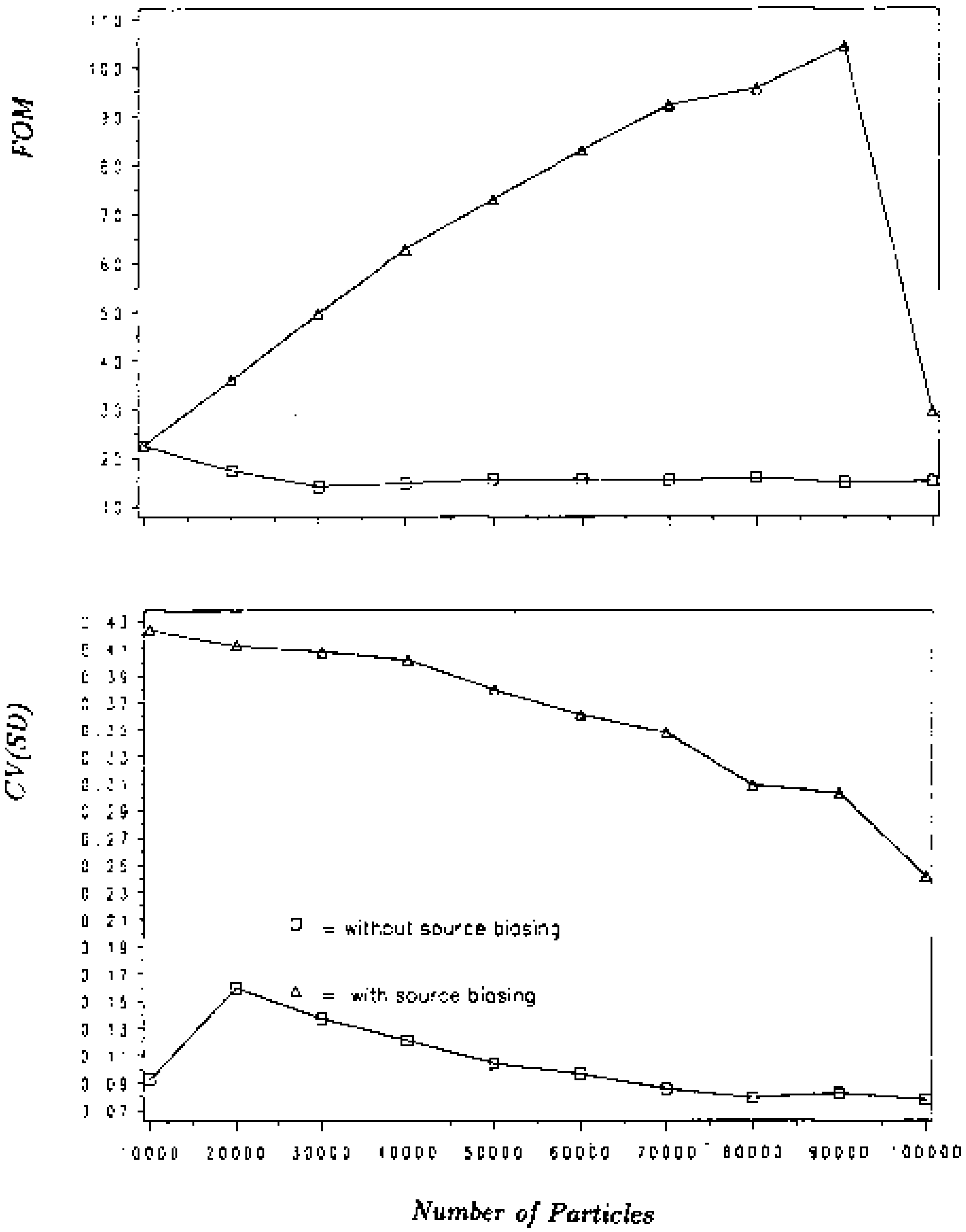

Figure 4.12: FOM and $CV(SD)$ Behavior for Sample Problem 3.

Chapter 5

Conclusions

In this work several potential techniques to detect undersampling conditions were investigated.

The utilization of F tests failed to be conclusive because the contributions were not normally distributed. However, as a recommendation for future work, it could be used if the samples were drawn from a population of group means which have a more normal distribution Also, problems that can be solved with a purely analog Monte Carlo calculation would have binomial distribution of the scores. In this case, F tests could be applied for other purposes rather than to detect undersampling conditions since this condition rarely occurs in such problems. For the reasons explained above, the utilization of F tests is not recommended without more study.

The calculation of the coefficient of variation of the standard deviation which provides confidence limits for the standard deviation estimate was found to be too expensive for deep penetration calculations because of its strong dependence upon the kurtosis of the contribution distribution. It also yields smaller values in the presence of undersampling which eventually increase with the sampling of important particles. Therefore, its utilization was found to be not too productive.

The benefits of calculating the figure of merit are twofold, first it allows an easy comparison of the efficiencies of different schemes and also is an indicator that the variance of the population has become stable. Although the *FOM* may stay essentially constant in severe undersampling conditions, once some important particles are processed it becomes an effective means to verify the stability of the estimates. Therefore, its usage is recommended.

The particle contribution distribution histogram was found to be the most effective way of detecting undersampling. In the breakdown region and more important — under severe undersampling conditions — the number of particles accounting for large fractions of the response is very small. This condition is easily identified from the histogram. This analysis does not heavily depend upon previous observations and is much faster and more reliable than any other means. It also provides a measure of the effectiveness of variance reduction techniques — which generally have the net effect of spreading out the values of the contributions producing a smoother distribution. Its utilization is highly recommended.

From the discussion above, it became evident that the most effective means to guarantee the reliability of an estimate in problems subject to undersampling is to know and characterize the population distribution. Therefore, the ability of the Monte Carlo method to simulate detailed descriptions of the population distribution must be exploited by the user — this is greatly facilitated by the new version of the MORSE code, MORSE/STAT — developed during the course of this research. This version of the code provides much more insight into the statistical quality of the solution obtained. The observation of the behavior of the new parameters as the solution evolves is a powerful means of determining if the undersampling condition exists. Because the standard version of the code provides only the standard deviation at the end of a complete run, the user looses all the interim information that the new version now provides.

The use of graphics for the presentation of the evolution of the statistics being estimated would enormously facilitate the observation of the behavior of the solution of the problem. Therefore, for future work, the coupling of a graphics package is highly recommended.

BIBLIOGRAPHY

Bibliography

[1] E. L. Gelbard. Unfinished Monte Carlo business. In *Proceedings ANS/ENS Topical Meeting of Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems*, 1981.

[2] S. N. Cramer, J. Gonnord, and J. S. Hendricks. Monte Carlo techniques for analyzing deep-penetration problems. *Nuclear Science and Engineering*, 92, 1986.

[3] A. Dubi. On analysis of the variance in Monte Carlo calculations. *Nuclear Science and Engineering*, 72, 1979.

[4] I. Lux and Z. Zatmary. Combined estimation of a common mean from few sample sets and from sets of rare events. *Nuclear Science and Engineering*, 89, 1985.

[5] RSIC Computer Code Collection CCC–474. *MORSE–CGA - A General Purpose Monte Carlo Multigroup Neutron and Gamma–Ray Transport Code System with Array Geometry Capability*. Oak Ridge National Laboratory, October 1987.

[6] H. R. Lindman. *Analysis of Variance in Complex Experimental Designs*. W. H. Freeman and Company, San Francisco, 1974.

[7] M. H. Hansen, W. N. Hurwitz, and W. G. Madow. *Sample Survey Methods and Theory*. John Wiley and Sons, Inc, New York, 1953.

[8] A. Dubi, T. Elperin, and H. Rief. On confidence limits and statistical convergence of Monte Carlo point-flux estimators with unbounded variance. *Annals of Nuclear Energy*, 9, 1982.

APPENDICES

Appendix A

Batch Output

A.1 Sample Problem 1

A.1.1 Next Event Surface Crossing Estimator

```
***START BATCH    1                             RANDOM-13579BDFDB97

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF SOURCE WHICH SETS WATE TO DDF AND PROVIDES AN ENERGY IG.
YOU ARE USING THE DEFAULT VERSION OF GTMED WHICH ASSUMES GEOMETRY AND XSECT MEDIA ARE IDENTICAL.
  WTAVE       TAVE    UAVE    VAVE    WAVE      XAVE        YAVE        ZAVE         AGEAVE
 1.000E+05    1.00   0.0009  0.0004  0.0013   0.0         0.0         0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                  RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR          BATCH              FSD           ACCUMULATED            FSD            CV(SD)            FOM
                   RESPONSE                             RESPONSE
      1           3.8398D-04          0.00327          3.8398D-04          0.00327          0.17907         3.5055D+04
      2           4.4619D-05          0.00526          4.4619D-05          0.00526          0.10571         1.3518D+04
      3           8.9053D 06          0.00679          8.9053D-06          0.00679          0.04823         8.1330D+03
      4           2.2075D-06          0.01134          2.2075D 06          0.01134          0.12860         2.9112D+03
      5           6.2544D-07          0.01497          6.2544D-07          0.01497          0.05511         1.6710D+03
      6           1.8826D-07          0.02405          1.8826D-07          0.02405          0.10014         6.4760D+02
      7           6.0974D-08          0.03847          6.0974D-08          0.03847          0.21179         2.5318D+02
      8           2.0435D-08          0.04535          2.0435D-08          0.04535          0.06699         1.8214D+02
      9           6.3437D-09          0.06413          6.3437D-09          0.06413          0.09174         9.1108D+01
     10           2.1429D-09          0.10191          2.1429D-09          0.10191          0.17910         3.6075D+01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0      100.0

      1        66060         0         0         0     10600     10902      3821      5249      3182       186
      2        56420         0     18868      7335      6219      9127         0         0      1586       445
      3        53727     16856     18827         0      3611      2010      1460      1905      1544        60
      4        65088     17609      7038      4878      3029         0      1162       642       297       257
      5        79683     11774      4679      1275      1088      1026         0       313       102        60
      6        83737     11062      3458       453       612       286       337         0         0        55
      7        93476      4674      1255       156       103       170        76        81         0         9
      8        96354      2586       602       225        83        47        48        26        28         1
      9        98235      1351       188       100        62        26        15         8        12         3
     10        99334       521        36        52        38         0         6         7         5         1

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)    FISHN   GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE    E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0        0        0   189483         0     98727        46    99954        0         0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 40 SECONDS.
```

```
***START BATCH   2                              RANDOM=24E04D19FDAB

SOURCE DATA
  WTAVE        JAVE    UAVE     VAVE    WAVE        XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05     1.00  -0.0003  0.0039  0.0000    0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                              RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR         BATCH           FSD           ACCUMULATED         FSD          CV(SD)          FOM
                    RESPONSE                        RESPONSE
        1          3.8273D-04       0.00311         3.8335D-04       0.00226       0.11038       3.6803D+04
        2          4.4120D-05       0.00439         4.4370D-05       0.00343       0.06929       1.5921D+04
        3          8.8814D-06       0.00746         8.8933D-06       0.00504       0.06061       7.3797D+03
        4          2.2191D-06       0.01225         2.2133D-06       0.00835       0.08322       2.6908D+03
        5          6.0900D-07       0.01549         6.1722D-07       0.01077       0.06376       1.6179D+03
        6          1.9154D-07       0.06106         1.8990D-07       0.03302       0.38768       1.7202D+02
        7          5.7283D-08       0.03704         5.9128D-08       0.02675       0.14283       2.6218D+02
        8          1.0250D-08       0.04934         1.0343D-08       0.03340       0.05806       1.6809D+02
        9          5.6615D-09       0.06491         6.0026D-09       0.04563       0.06415       9.0066D+01
       10          2.1078D-09       0.10720         2.1253D-09       0.07393       0.10690       3.4318D+01

    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

        1       132189          0          0          0      21311      12256      17050      10461       6386        346
        2       112869          0      26190      26281      12466      18104          0          0       3205        885
        3       107469      33677      37648          0       7256       4132       2832       3807       3068        111
        4       130292      35140      14137       9807       5953          0       2273       1274        589        535
        5       159478      23730       9132       2533       2137       2073          0        614          0        303
        6       170370      20745       5437        913       1206        650        575          0         68         36
        7       185365      10233       3218          0        551        246        192        177          0         18
        8       192784       5137       1154        502        152         88         86         42         51          4
        9       196529       2694        269        334        156          0         49         41         22          6
       10       198697       1028         67        106         58          0         16         13         12          3

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN   GAMGEN REALCOLL    ALBEDO      BDRYX    ESCAPB     E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
  100000        0         0        0   189052         0      98516        53     99947        0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 39 SECONDS.
```

```
***START BATCH    0                              RANDOM-3F3275B6B73B

SOURCE DATA
  WTAVE        JAVE     UAVE     VAVE     WAVE        XAVE        YAVE        ZAVE        AGEAVE
 1.000E+05     1.00  -0.0005 -0.0004   0.0013    0.0         0.0         0.0         0.0

 PARTICLES PER CM**2 PER SECOND

                                    RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR           BATCH              FSD          ACCUMULATED          FSD           CV(SD)            FOM
                     RESPONSE                          RESPONSE
       1           3.8373D-04         0.00305         3.8317D 04         0.00141        0.16056        2.3771D+04
       2           4.4566D-05         0.00523         4.4517D-05         0.00230        0.20914        8.9237D+03
       3           9.0082D-06         0.00783         8.9607D-06         0.00282        0.05729        5.9278D+03
       4           2.2733D-06         0.01736         2.2431D-06         0.00770        0.35579        7.9330D+02
       5           6.2939D-07         0.01752         6.2615D-07         0.00657        0.11378        1.0905D+03
       6           1.9336D-07         0.02525         1.8713D 07         0.01066        0.24013        4.1411D+02
       7           6.4693D-08         0.03323         5.3507D-08         0.01278        0.06307        2.8782D+02
       8           1.9984D-08         0.04316         1.9247D-08         0.02064        0.16855        1.1039D+02
       9           6.4810D-09         0.06625         6.1403D-09         0.02373        0.03558        8.3492D+01
      10           2.0340D-09         0.10733         2.0756D-09         0.03584        0.05307        3.6615D+01

   DETECTOR            PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1        529853'         0          0          0      84241      87223      30172      41805      25352       1353
       2        452633          0     150167      58858      49638      72213          0          0      12810       3681
       3        430779     133766     150331          0      28621      16754      20115      11242       7919        472
       4        521317     140381      55955      39103      24188          0       9209       5238       3340       1269
      .5        638110      93931      37016      10163       8619       8565          0       2442        707        447
       6        681110      76996      27975       3767       4739       2127       2837          0          0        449
       7        740819      41476      12958          0       2239        958        769        701          0         80
       8        770882      20727       4371       2370        552        335        390        177        183         13
       9        786102      10870       1101        960        445        208        102         75        104         33
      10        794864       3347        937        398        277          0         85         43         31         18

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO      BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000         0         0        0   109577         0      99451        47     99953        0         0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 39 SECONDS.
```

```
***START BATCH    0                              RANDOM-23BFA3F2639F

SOURCE DATA
  WTAVE        JAVE     UAVE     VAVE     WAVE        XAVE         YAVE         ZAVE         AGEAVE
 2.000E+05     1.00   0.0022 -0.0000   0.0022    0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR          BATCH            FSD          ACCUMULATED          FSD            CV(SD)             FOM
                    RESPONSE                         RESPONSE
       1           3.8472D-04       0.00348        3.8334D-04         0.00131        0.14715         2.4404D+04
       2           4.4803D-05       0.00469        4.4549D-05         0.00211        0.19622         9.4277D+03
       3           8.9128D-06       0.00690        8.9554D-06         0.00262        0.05259         6.0939D+03
       4           2.2128D-06       0.00949        2.2398D-06         0.00693        0.34775         8.6942D+02
       5           6.1276D 07       0.01534        6.2466D-07         0.00609        0.10566         1.1285D+03
       6           1.7981D-07       0.02118        1.8632D-07         0.00978        0.22722         4.3681D+02
       7           5.5462D-08       0.03434        5.9057D-08         0.01200        0.05833         2.9038D+02
       8           1.7885D-08       0.05078        1.9095D 08         0.01923        0.15596         1.1298D+02
       9           5.4812D-09       0.07087        6.0670D-09         0.02250        0.03368         8.2513D+01
      10           1.8839D-09       0.10804        2.0543D-09         0.03402        0.04968         3.6114D+01

   DETECTOR            PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

               0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1     595879          0          0          0      94801      98081      33904      47112      28712       1510
       2     509025          0     168798      66221      56059      81330          0          0      14422       4145
       3     484478     150339     169496          0      32261      18786      22543      12673       8895        528
       4     586235     158142      63076      43886      27236          0      10347       5890       3764       1424
       5     717738     105839      41604      11408       9752       9643          0       2725        803        488
       6     754266      98561      31573       4224       4062       3646       3163          0          0        505
       7     833392      46723      11032       3572       2491       1078        849        777          0         86
       8     867238      23382       4922       2639        600        378        422        198        204         17
       9     884445      12186       1241        957        593        231        113         84        113         37
      10     894297       3734       1022        446        306          0         90         49         35         21

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL    ALBEDO      BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000         0          0         0   189798         0      98936        55     99945        0         0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 39 SECONDS.
```

```
***START BATCH  10                                    RANDOM=DCC13A2747OF

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE        XAVE        YAVE        ZAVE        ACEAVE
 1.000E+05     1.00   0.0008   0.0011   0.0010    0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                            RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR          BATCH          FSD          ACCUMULATED          FSD          CV(SD)          FOM
                    RESPONSE                       RESPONSE
       1          3.8186D-04      0.00322        3.8319D-04        0.00122        0.13722        2.5230D+04
       2          4.4287D-05      0.00650        4.4523D-05        0.00200        0.17752        9.3759D+03
       3          9.1627D-06      0.02447        8.9761D-06        0.00343        0.22880        3.1964D+03
       4          2.2404D-06      0.01259        2.2398D-06        0.00637        0.33420        9.2849D+02
       5          6.2488D-07      0.01638        6.2468D 07        0.00572        0.09747        1.1512D+03
       6          1.8444D-07      0.02188        1.8613D 07        0.00907        0.21428        4.5682D+02
       7          5.7758D-08      0.03139        5.8928D-08        0.01125        0.05431        2.9727D+02
       8          1.8387D-08      0.06864        1.9024D 08        0.01860        0.13948        1.0877D+02
       9          5.8383D-09      0.06809        6.0442D 09        0.02137        0.03183        8.2390D+01
      10          2.1751D-09      0.10446        2.0664D 09        0.03236        0.04605        3.5921D+01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1      662277         0          0          0     105337     108947      37615      52237      31913       1673
       2      565774         0     187608      73382      62348      90329          0          0      15975       4583
       3      538490    167125     180134          0      35685      20868      25121      14123       9865        589
       4      651487    175594      70015      48809      30237          0      11539       6571       4183       1565
       5      797553    117404      46254      12758      10858      10711          0       3035          0       1427
       6      838013    109550      35143       4666       4526       4028       3504          0          0        570
       7      925919     52009      12262       3931       2774       1203        950        857          0         95
       8      963540     26067       5450       2954        657        412        461        221        218         20
       9      982734     13539       1486       1154        547        255        126         97        120         42
      10      993666      4160       1130        489        333          0        103         54         41         24

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)    FISHN   GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE    E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0        0        0   188934         0     98992        53    99947        0         0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 39 SECONDS.
```

A.1.2 Point Detector Estimator

```
***START BATCH    1                              RANDOM=13579BDFDB97

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF SOURCE WHICH SETS WATE TO DDF AND PROVIDES AN ENERGY IG.
YOU ARE USING THE DEFAULT VERSION OF GTMED WHICH ASSUMES GEOMETRY AND XSECT MEDIA ARE IDENTICAL.
  WTAVE        IAVE     UAVE    VAVE     WAVE       XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05     1.00    0.0022 -0.0005  0.0018    0.0          0.0          0.0           0.0

 PARTICLES PER CM**2 PER SECOND

                              RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR          BATCH           FSD          ACCUMULATED          FSD            CV(SD)            FOM
                     RESPONSE                        RESPONSE
       1            3.8195D-04      0.02420        3.8195D-04         0.02420         0.18380         9.6140D+01
       2            4.0657D-05      0.03220        4.0657D-05         0.03220         0.17042         5.4278D+01
       3            1.0318D-05      0.18977        1.0318D-05         0.18977         0.45750         1.5631D+00
       4            2.2010D-06      0.10484        2.2010D-06         0.10484         0.24872         5.1215D+00
       5            5.8018D-07      0.16399        5.8018D-07         0.16399         0.24418         2.0932D+00
       6            1.3670D-07      0.13992        1.3670D-07         0.13992         0.20058         2.8754D+00
       7            1.0456D-07      0.65115        1.0456D-07         0.65115         0.49116         1.3276D-01
       8            3.6075D-08      0.39992        3.6075D-08         0.39992         0.31527         3.5196D-01
       9            4.4715D-09      0.28794        4.4715D-09         0.28794         0.26414         6.7894D-01
      10            8.1909D-10      0.22073        8.1909D-10         0.22073         0.24052         1.1553D+00

    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1          82140         0          0          0       6767       7358        717        735        251         49
       2          69700         0      14634       7215       3000       1316        971         68        103          5
       3          78956     13447       1962       1753        289        144         23         13          1          0
       4          81208     12320       2213        405        198         43         21          6          5          0
       5          90177      4818       1100        165         76         18          7          3          1          1
       6          88529      6765        885        246         37         34          2          4          2          0
       7          96335       288         15          3          1          0          0          0          0          0
       8          96700        30          0          1          1          1          0          0          0          0
       9          96124       619         63         10          0          1          2          0          1          0
      10          96140       703         86         21          0          3          2          0          1          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT  TIMEKILL R R KILL R R SURV   GAMLOST
  100000         0          0         0   188986         0     98169        48     99952          0          0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 45 SECONDS.
```

```
***START BATCH    2                          RANDOM=E66B32896347

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE        XAVS        YAVE       ZAVE        AGEAVE
 1.000E+05     1.00    0.0012   0.0013  -0.0041      0.0         0.0        0.0         0.0

 PARTICLES PER CM**2 PER SECOND

                             RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR         BATCH           FSD          ACCUMULATED          FSD           CV(SD)           FOM
                   RESPONSE                        RESPONSE
       1          3.8376D 04      0.02215        3.8286D-04        0.01640        0.11907        1.0453D+02
       2          3.9041D-05      0.02885        3.9849D-05        0.02167        0.11323        5.9872D+01
       3          1.0062D-05      0.16319        1.0190D-05        0.12539        0.30288        1.7883D+00
       4          1.9870D-06      0.08756        2.0940D-06        0.06900        0.18091        5.9046D+00
       5          6.1231D 07      0.26090        5.9624D-07        0.15592        0.35056        1.1564D+00
       6          1.4004D-07      0.20385        1.3837D-07        0.12417        0.28457        1.8236D+00
       7          3.9161D-08      0.29061        7.1859D-08        0.48030        0.47789        1.2188D-01
       8          1.6220D-08      0.28986        2.6147D-08        0.29016        0.28698        3.3394D-01
       9          6.8172D-09      0.54999        5.6443D-09        0.35118        0.43483        2.2798D-01
      10          2.5654D-09      0.60292        1.6923D-09        0.46012        0.45654        1.3280D-01

   DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0        10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1        164540'         0          0          0      13448      14540       1483       1387        617         71
       2        139619          0      29245      12846       7456       2592       1562        563        156         40
       3        158315      26555       3992       3536        554        274         50         19          2          0
       4        156810      29433       4799       1350        415        145         64          7         12          0
       5        180518      10233       1634        356        126         37         11          4          2          0
       6        178712      12468       1426        384         90         51          4          6          2          1
       7        191744       1422        153         21          4          2          1          0          0          0
       8        193308        223         18          0          1          2          1          0          0          0
       9        193073        643         21         10          5          2          1          1          0          0
      10        193791        173          5          5          1          1          0          0          0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN   GAMGEN REALCOLL     ALBEDO      BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000         0        0         0   189185          0      98732        45     99955         0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 48 SECONDS.
```

```
***START BATCH    3                           RANDOM=6A5E9CFC0C63

SOURCE DATA
  WTAVE        IAVE     UAVE      VAVE      WAVE        XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05      1.00   -0.0026 -0.0011  -0.0049     0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                 RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR           BATCH             FSD            ACCUMULATED          FSD           CV(SD)             FOM
                      RESPONSE                            RESPONSE
        1           3.6996D-04        0.01881            3.7856D-04        0.01264        0.10199         1.1718D+02
        2           5.7066D 05        0.12868            4.5588D-05        0.05516        0.23556         6.1556D+00
        3           8.3233D-06        0.06395            9.5679D-06        0.09094        0.29039         2.2647D+00
        4           2.7789D-06        0.29318            2.3223D-06        0.12408        0.37712         1.2165D+00
        5           6.1005D-07        0.18897            6.0085D-07        0.12137        0.27644         1.2714D+00
        6           1.2668D-07        0.13773            1.3447D-07        0.09553        0.23163         2.0523D+00
        7           3.4557D-08        0.19707            5.9425D-08        0.38908        0.47330         1.2372D-01
        8           9.9329D-09        0.24349            2.0743D-08        0.24692        0.28005         3.0717D-01
        9           6.2381D-09        0.68128            5.8422D-09        0.33160        0.33411         1.7033D-01
       10           1.0023D-09        0.49815            1.4623D-09        0.37279        0.41650         1.3476D 01

    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

        1       247165         0          0          0      19974      21772       2183       2061        890        158
        2       209621         0      43915      30314       2570       3398       1046        275         19          4
        3       237775     28522      16993       5227        441        801         42         63          8          0
        4       244352     36525       6569       1379        459        129         49         23          1          1
        5       273349     13344       1903        593        149         37         13          7          2          0
        6       267957     18414       2318        721        125         84          6          8          4          1
        7       286616      2900        448         27         17          5          3          0          0          0
        8       289571       691         53         13          0          3          1          1          0          0
        9       289756       841         32         13          4          2          0          1          0          0
       10       290482       465         38         10          3          0          1          0          0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      PISHN     GANGEN REALCOLL     ALBEDO      BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
  100000         0          0         0   189879          0      99126        55     99945        0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 49 SECONDS.
```

```
***START BATCH    4                              RANDOM-C19F64C8B3EF

SOURCE DATA
  WTAVE        IAVE    UAVE    VAVE     WAVE      XAVE        YAVE        ZAVE        AGEAVE
 1.000E+05     1.00  -0.0006 -0.0027  0.0006    0.0         0.0         0.0         0.0

 PARTICLES PER CM**2 PER SECOND

                                     RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR            BATCH                FSD           ACCUMULATED           FSD            CV(SD)            FOM
                      RESPONSE                              RESPONSE
       1            3.9975D-04           0.02955           3.8386D-04          0.01211         0.08832         9.5714D+01
       2            4.3695D-05           0.03807           4.5115D 05          0.04281         0.22480         7.6597D+00
       3            7.9846D-06           0.06284           9.1721D-06          0.07245         0.28023         2.6741D+00
       4            2.5077D-06           0.20961           2.3686D-06          0.10678         0.29830         1.2310D+00
       5            9.0974D-07           0.42952           6.7807D-07          0.16511         0.37187         5.1487D-01
       6            1.4804D-07           0.13330           1.3786D-07          0.07851         0.18837         2.2771D+00
       7            4.7006D-08           0.17820           5.6320D-08          0.31013         0.46651         1.4594D-01
       8            1.0194D-08           0.15543           1.8105D-08          0.21329         0.27711         3.0853D-01
       9            2.5654D-09           0.13361           5.0230D-09          0.28976         0.33295         1.6718D-01
      10            7.1483D-10           0.15417           1.2754D-09          0.32129         0.41462         1.3598D-01

   DETECTOR               PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                 0.0         10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

       1         329383          0           0           0       26758       29111        2845        2969        1045         139
       2         279176          0       58615       40614        3386        4215        1424         580         112          10
       3         316751      31761       26433        7785        2501         789         274         116          21           2
       4         329392      45013        8839        1833         610         184          66          19           2           1
       5         366931      15938        2285         605          62          44           6           6           1           0
       6         359588      22629        2929         751         174         110          16           9           7           1
       7         382177       3856         522         131          45           4           5           3           0           0
       8         385734       1235         174          21           8           2           2           1           0           0
       9         385546       1862         158          35          11           4           2           1           0           0
      10         387066        878          94           9           8           2           1           0           0           0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL     ALBEDO      BDRYX     ESCAPE      E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0          0         0   189773          0      99096         59      99941        0          0          0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 50 SECONDS.
```

```
***START BATCH    5                              RANDOM-DCB41BUCA4AF

SOURCE DATA
  WTAVE        IAVE    UAVE     VAVE     WAVE       XAVE        YAVE        ZAVE        AGEAVR
 1.000E+05     1.00  -0.0009  0.0035   0.0044    0.0         0.0         0.0         0.0

 PARTICLES PER CM**2 PER SECOND

                               RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR        BATCH            FSD          ACCUMULATED        FSD           CY(SD)           FOM
                  RESPONSE                         RESPONSE
       1         3.8594D-04       0.02529        3.8427D-04       0.01093       0.07949        9.3948D+01
       2         4.7310D-05       0.09346        4.5554D-05       0.03908       0.19146        7.3487D+00
       3         8.3560D-06       0.06706        9.0089D-06       0.06031       0.26855        3.0857D+00
       4         2.6277D-06       0.23210        2.4205D-06       0.09761       0.25087        1.1779D+00
       5         5.4350D-07       0.11686        6.5116D 07       0.13893       0.36455        5.8148D-01
       6         1.2134D-07       0.08476        1.3456D-07       0.06614       0.17845        2.5652D+00
       7         3.3158D 08       0.22320        5.1600D-08       0.27186       0.46134        1.5185D-01
       8         1.0061D-08       0.43582        1.6496D-08       0.19460       0.25836        2.9613D-01
       9         3.1820D-09       0.38605        4.6548D-09       0.25565       0.31928        1.7171D-01
      10         7.7008D-10       0.29357        1.1743D-09       0.28179       0.40696        1.4133D-01

   DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0      100.0

       1      411146         0         0         0     33573     36822      3569      3713      1290       171
       2      348547         0     73254     51120      4290      5595      1424       763       105         9
       3      395456     39924     33235      9785      3145       893       421       140        35         4
       4      415652     54026      9262      2890       201       254        77        23         2         2
       5      455252     22537      3636       613       208        65        20        12         2         1
       6      446245     30955      3986      1200       193       129        44        26        13         1
       7      477683      4892       575       213        74        19         7         3         0         0
       8      482198      1554       184        45        18         1         3         1         1         0
       9      481923      2282       268        42        12         6         2         0         1         0
      10      483851      1036       138        34        14         5         0         1         0         0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL     ALBEDO     BDRYX    ESCAPE     E CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0         0         0   189793          0     98928        51     99949        0         0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 51 SECONDS.
```

```
***START BATCH    6                                  RANDOM-3C2EC98ECCAB

SOURCE DATA
  WTAVE         IAVE     UAVE    VAVE    WAVE       XAVE         YAVE         ZAVE         ACEAVE
 1.000E+05      1.00   0.0020  0.0012  0.0014    0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                  RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR          BATCH              FSD           ACCUMULATED           FSD            CV(SD)            FOM
                    RESPONSE                            RESPONSE
       1          4.0937D-04         0.03452          3.8846D 04         0.01086          0.06968          7.9325D+01
       2          4.1039D-05         0.03509          4.4801D-05         0.03354          0.18672          8.3148D+00
       3          8.6497D-06         0.05580          8.9490D-06         0.05138          0.26036          3.5432D+00
       4          1.9491D-06         0.11055          2.3419D-06         0.08546          0.24309          1.2810D+00
       5          1.2141D-06         0.36437          7.4498D-07         0.14155          0.25251          4.6696D-01
       6          5.4674D-07         0.51612          2.0326D-07         0.23425          0.45784          1.7050D-01
       7          9.3560D-08         0.28522          5.8667D-08         0.21351          0.40497          2.0523D-01
       8          1.4453D-08         0.16652          1.6156D-08         0.16750          0.25272          3.3345D-01
       9          3.6859D-09         0.18078          4.4933D-09         0.22208          0.31534          1.8969D-01
      10          9.1095D-10         0.19769          1.1304D-09         0.24539          0.40221          1.5537D-01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1      493545          0          0          0      40204      44033       4260       4706       1309        207
       2      418609          0      87715      61190       5141       6336       2076        865        192         15
       3      474818      47695      39771      11181       4351       1078        500        166         44          6
       4      493592      67864      13228       2644       1046        269        115         50          1          3
       5      555655      19530       2746        670        112         52         15          4          3          0
       6      562898      13980       1978        343        112         17         11          3          1          0
       7      573180       5878        940         76         42         13          5          3          1          0
       8      577035       3424        222         71         18         10          3          2          1          0
       9      578294       2760        304         78         19         10          4          1          1          0
      10      580614       1257        161         46         22          7          3          1          0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)    FISHN    GAMGEN REALCOLL    ALBEDO    BDRYX    ESCAPE    E CUT TIMEKILL R R KILL R R SURV  GAMLOST
 100000        0         0        0   188636         0     98973        47    99953        0          0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 47 SECONDS.
```

```
***START BATCH    7                              RANDOM-C9F1F9A47EAB

SOURCE DATA
  WTAVE          JAVE     UAVE     VAVE     WAVE        XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05       1.00    0.0016   0.0003   0.0004     0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                               RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR          BATCH            FSD          ACCUMULATED          FSD            CV(SD)            FOM
                     RESPONSE                         RESPONSE
        1           3.9492D-04       0.03325         3.8938D-04       0.01046         0.07045         7.3247D+01
        2           4.3649D-05       0.07982         4.4637D-05       0.03094         0.17086         8.3766D+00
        3           8.6719D 06       0.04958         8.9094D-06       0.04477         0.25421         3.9992D+00
        4           1.7272D-06       0.05357         2.2527D-06       0.07638         0.24167         1.3744D+00
        5           6.6371D-07       0.28177         7.3337D-07       0.12852         0.23500         4.8540D-01
        6           1.0434D-07       0.42497         1.8912D-07       0.21837         0.44716         1.6812D-01
        7           4.0265D 08       0.15088         5.6338D-08       0.19222         0.40234         2.1698D-01
        8           1.1255D-08       0.23298         1.5456D-08       0.15202         0.24644         3.4691D-01
        9           2.6400D-09       0.14657         4.2286D-09       0.20270         0.31403         1.9513D-01
       10           5.4344D-10       0.23364         1.0466D-09       0.22784         0.39989         1.5443D-01


    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS .

                  0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

        1        575773          0           0           0       46873       51419        4982        5490        1526         245
        2        488250          0      102341       71537        6007        7400        2424        1016         227          19
        3        553868      49189       52998       13059        4007        2319         599         186          64          10
        4        560939      90717       16617        5130        1155         545         181          40          13           3
        5        648353      22695        3237         776         123          68          14           7           3           0
        6        649229      23861        2316         366         124          50          14           3           1           0
        7        668738       6858        1079          75          70           8          11           5           1           0
        8        672824       4369         244          98          36          14           1           4           1           0
        9        673264       4502         496          83          34           7           7           2           1           0
       10        675952       2906         158          91          20          14           3           1           0           0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL     ALBEDO      BDRYX     ESCAPE      E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0          0         0   189063          0      98413         48      99952        0          0          0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 50 SECONDS.
```

```
***START BATCH    0                              RANDOM=B064BD2C4417

SOURCE DATA
  WTAVE        JAVE    UAVE     VAVE      WAVE        XAVE         YAVE         ZAVE         AGEAVE
 1.000E+05     1.00  -0.0021  -0.0007   0.0024     0.0          0.0          0.0          0.0

 PARTICLES PER CM**2 PER SECOND

                                RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR          BATCH              FSD           ACCUMULATED           FSD             CV(SD)             FOM
                    RESPONSE                           RESPONSE
      1           3.9315D-04          0.03379          3.8985D-04          0.01009          0.06970          6.8916D+01
      2           4.9785D-05          0.10155          4.5280D-05          0.03011          0.15079          7.7312D+00
      3           7.9145D-06          0.05531          8.7851D-06          0.04022          0.24815          4.3349D+00
      4           1.8957D-06          0.09951          2.2080D-06          0.06901          0.23591          1.4721D+00
      5           5.4595D-07          0.09778          7.0994D-07          0.11654          0.23347          5.1622D-01
      6           2.2764D-07          0.33281          1.9394D-07          0.19262          0.41951          1.8896D-01
      7           4.3906D-08          0.23969          5.4521D-08          0.17455          0.39471          2.3013D-01
      8           1.3283D-08          0.25663          1.5184D-08          0.13827          0.23668          3.6670D-01
      9           3.8514D-09          0.35037          4.1814D-09          0.18384          0.29964          2.0745D-01
     10           8.2293D-10          0.27035          1.0186D-09          0.20665          0.39295          1.6419D-01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                 0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

      1        658062         0          0          0      53486      58798       5697       6304       1747        274
      2        557855         0     117020      81862       6880       8414       2793       1207        226         20
      3        632771     56397      60570      13962       5503       2483        850        187         93         11
      4        640882    103804      18966       5551       1628        606        215         36         24          4
      5        737559     28696       3853       1259        126        109          9         13          3          0
      6        741932     27199       2654        433        143         52         22          3          2          0
      7        764194      7823       1085        240        107          8         11          7          1          0
      8        768331      5580        255        147         33         21          2          3          2          0
      9        769421      5169        574         88         44         10          8          1          1          0
     10        772134      3464        395        126         21         16          1          3          0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISSN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000        0          0         0   189675         0     99466        54     99946        0          0          0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 53 SECONDS.
```

```
***START BATCH    9                              RANDOM-F82DB70E05FF

SOURCE DATA
  WTAVE          IAVE     UAVE     VAVE     WAVE       XAVE       YAVE       ZAVE        AGEAVE
 1.000E+05       1.00  -0.0014  -0.0059   0.0008    0.0        0.0        0.0         0.0

 PARTICLES PER CM**2 PER SECOND

                                 RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR          BATCH              FSD           ACCUMULATED          FSD          CV(SD)           FOM
                    RESPONSE                            RESPONSE
       1          4.0154D-04         0.03399          3.9115D-04         0.00974        0.06437       6.5685D+01
       2          5.4395D 05         0.23517          4.6293D-05         0.03158        0.14705       6.2514D+00
       3          8.4099D-06         0.09029          8.7434D-06         0.03719        0.23244       4.5057D+00
       4          1.7494D-06         0.07066          2.1570D 06         0.06312        0.23352       1.5646D+00
       5          7.2271D 07         0.20546          7.1136D-07         0.10596        0.22280       5.5515D-01
       6          1.4888D-07         0.13552          1.8893D-07         0.17616        0.41761       2.0084D 01
       7          3.8305D-08         0.12517          5.2720D 08         0.16077        0.39315       2.4113D-01
       8          4.0050D-08         0.83420          1.7947D-08         0.23151        0.40162       1.1628D-01
       9          1.8052D-09         0.85224          5.7226D-09         0.32169        0.43233       6.0227D-02
      10          1.9662D-09         0.53664          1.1239D-09         0.19646        0.31187       1.6148D-01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1      740468         0          0          0      60023      66136      .6371      7417       1748        227
       2      627553         0     149852      73934       7733      10578       2670       788        193         10
       3      711802     63459      60160      15730       6169       2788        955       209        104         13
       4      704967    132823      21391       6219       1831        606        321        39         25          4
       5      829883     32297       4735       1007        147        125         15        10          1          2
       6      834804     30599       2972        463        194         53         28         2          3          0
       7      859764      8801       1182        303         95         31         17        10          1          0
       8      867854      2824        393         85         32          2          4         1          1          0
       9      870112      1914        118         33         12          1          2         0          0          0
      10      870879      1882        271         59         32          0          5         0          0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN  REALCOLL    ALBEDO      BDRYX    ESCAPE     E-CUT  TIMEKILL  R R KILL  R R SURV  GAMLOST
  100000        0         0         0    189195         0      99061        51     99949         0         0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 49 SECONDS.
```

```
***START BATCH  10                          RANDOM=ACF312D08EDB

SOURCE DATA
  WTAVE        IAVE   UAVE     VAVE    WAVR       XAVE       YAVE        ZAVE       AGEAVE
 1.0000E+05     1.00  -0.0019  0.0022  0.0032    0.0        0.0         0.0        0.0

 PARTICLES PER CM**2 PER SECOND

                                 RESPONSES(DETECTOR)    NEUTRON  FLUX
    DETECTOR          BATCH             FSD           ACCUMULATED          FSD            CV(SD)           FOM
                    RESPONSE                            RESPONSE
        1           3.8024D-04        0.02493          3.9006D-04        0.00912         0.06178        6.7442D+01
        2           4.2527D-05        0.08825          4.5916D-05        0.02979         0.14028        6.3208D+00
        3           1.2067D-05        0.30180          9.0757D-06        0.05148         0.30909        2.1171D+00
        4           1.7908D-06        0.08807          2.1204D-06        0.05826         0.22978        1.6529D+00
        5           6.4603D-07        0.16061          7.0483D-07        0.09736         0.21780        5.9187D-01
        6           2.3432D-07        0.11396          1.8347D-07        0.16347         0.41652        2.0995D-01
        7           3.3903D-08        0.24804          5.0838D-08        0.15096         0.38844        2.4621D-01
        8           1.2554D-08        0.29538          1.7408D-08        0.21587         0.39772        1.2040D-01
        9           4.3514D-09        0.40431          5.5855D-09        0.29830         0.42753        6.3056D-02
       10           1.1817D-10        0.17964          1.0233D-09        0.21594         0.26580        1.2032D-01

    DETECTOR            PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                 0.0       10.0      20.0      30.0       40.0      50.0      60.0       70.0      80.0       90.0     100.0

        1         822591         0         0          0     66835     73435       7097      8228      1869        333
        2         696863         0    146584     102522      8582     11123       3570       870       201         18
        3         790633     80272     66264      19624      6252      1790        840       281        73          9
        4         782995    147859     23780       6419      2527       667        362        30        38          4
        5         922034     35882      4794       1585       161       140         10        19         1          2
        6         927492     33989      3311        456       270        43         48         1         4          0
        7         955274      9730      1164        469       101        39         19         8         3          1
        8         964274      3122       424         95        34         4          4         1         1          0
        9         966562      2324       125         40        15         1          1         1         0          0
       10         967635      2078       295         65        27         8          4         2         0          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN   GAMGEN REALCOLL    ALBEDO    BDRYX    ESCAPE    E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
  100000         0        0        0   189336         0    98637       56    99944        0        0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 17 MINUTES, 46 SECONDS.
```

A.2 Sample Problem 2

```
***START BATCH    1                              RANDOM=000035FA731A

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF SOURCE WHICH SETS WATE TO DDF AND PROVIDES AN ENERGY IG.
  WTAVE        IAVE     UAVE     VAVE     WAVE        XAVE        YAVE        ZAVE        AGEAVE
 9.687E-04     10.23   0.0007 -0.0001 -0.0027    0.0         0.0         0.0          0.0

4 PI R**2 FLUENCE

                                      RESPONSES(DETECTOR)  NEUTRON FLUX
  DETECTOR          BATCH                 FSD            ACCUMULATED            FSD           CV(SD)            FOM
                   RESPONSE                               RESPONSE
     1            1.9356D+00           0.01002          1.9356D+00           0.01002        0.06790         1.5650D+03
     2            2.1357D+00           0.01358          2.1357D+00           0.01358        0.09257         8.5118D+02
     3            1.8077D+00           0.01382          1.8077D+00           0.01382        0.06938         8.2183D+02
     4            6.6802D-01           0.01746          6.6802D-01           0.01746        0.07040         5.1510D+02
     5            4.3622D-01           0.01763          4.3622D-01           0.01763        0.04574         5.0528D+02
     6            1.7872D-01           0.02763          1.7872D-01           0.02763        0.08732         2.0563D+02
     7            4.2268D-02           0.04378          4.2268D-02           0.04378        0.12017         8.1923D+01

  DETECTOR            PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0      10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0     100.0

     1          54507     23578         0      6594      5412      3180      2963         0       676       298
     2          53277     17817      6512      5591      3485      3430         0       599       399       334
     3          52490     13827      5429      4839      2998      2862         0       525       351       283
     4          38218     11400         0      2219      1946      1115       948         0       212       128
     5          33332      5916      3919      1514       819       991       785         0       146        74
     6          23365      3780      3779         0       638       304       434       189        31        35
     7          12721       898      1170       966         0       118        85        71        26         9

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
  100000      1960        0         0  1917308         0   1084495         0     90000        0      11960       832         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 6 MINUTES, 22 SECONDS.
```

```
***START BATCH    1                          RANDOM-195E7A42575A

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE       XAVE         YAVE         ZAVE          AGEAVE
 9.687E+04    10.23   -0.0007   0.0045   0.0007    0.0          0.0          0.0           0.0

4 PI R**2 FLUENCE

                                    RESPONSES(DETECTOR)  NEUTRON FLUX
  DETECTOR            BATCH                FSD           ACCUMULATED           FSD            CV(SD)             FOM
                     RESPONSE                              RESPONSE
     1            1.9364D+00            0.00965          1.9323D+00         0.00509          0.03785          1.5200D+03
     2            2.0632D+00            0.01234          2.0821D+00         0.00669          0.07246          8.8054D+02
     3            1.7538D+00            0.01129          1.7815D+00         0.00678          0.05849          8.5936D+02
     4            6.5784D-01            0.01681          6.5927D-01         0.00886          0.04068          5.0220D+02
     5            4.3723D-01            0.01896          4.3334D-01         0.00960          0.03323          4.2843D+02
     6            1.7664D-01            0.02762          1.7415D-01         0.01414          0.08946          1.9727D+02
     7            4.1097D-02            0.03931          4.0930D-02         0.02236          0.09756          7.8876D+01

  DETECTOR              PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

     1          218391      93707           0       26647       21541       12808       11736           0        2675        1185
     2          212557      71346       26538       14053       19413       16649           0        2393        1502        1192
     3          209494      55161       22094       19438       11965       11508           0        2140        1409        1044
     4          152410      45703           0        8688        7537        4354        3793           0         904         479
     5          123567      32732       15823        6192        3144        3996        2974           0         542         271
     6           92951      15200       15051           0        2432        1170        1453         967         133         128
     7           50402       3695        4640        2052        1627         503         352         240          87          46

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN     GAMGEN REALCOLL    ALBEDO     BDRYX     ESCAPE      E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000      1764         0          0  1910548         0  1082420          0      89976        0     11788       877        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 6 MINUTES, 20 SECONDS.
```

```
***START BATCH    8                           RANDOM-E18DB73979A2

SOURCE DATA
  WTAVE         IAVE     UAVE     VAVE     WAVE        XAVE         YAVE         ZAVE          AGEAVE
 9.687E+04     10.24   -0.0006   0.0023  -0.0008    0.0          0.0          0.0           0.0

4 PI R**2 FLUENCE

                                        RESPONSES(DETECTOR)  NEUTRON FLUX
    DETECTOR            BATCH                FSD           ACCUMULATED          FSD             CV(SD)             FOM
                       RESPONSE                              RESPONSE
        1            1.9682D+00            0.01336        1.9363D+00          0.00395          0.05527          1.2603D+03
        2            2.0506D+00            0.01050        2.0677D+00          0.00474          0.06635          8.7580D+02
        3            1.7800D+00            0.01507        1.7806D+00          0.00481          0.04436          8.5148D+02
        4            6.5762D-01            0.01868        6.5702D-01          0.00636          0.03780          4.8643D+02
        5            4.3619D-01            0.02124        4.3198D-01          0.00685          0.02481          4.1995D+02
        6            1.6936D-01            0.02359        1.7512D-01          0.00991          0.05817          2.0069D+02
        7            3.9773D-02            0.04340        4.0704D-02          0.02215          0.25553          4.0139D+01

    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

        1        437702     186731           0       53153       43326       25502       23435           0        5417        2349
        2        425354     142808       52742       27985       39041       33274           0        4796        3099        2332
        3        420041     110019       43804       39019       18223       28678           0        4138        2754        2138
        4        305183      91018           0       17699       15086        8620        7538           0        1801         933
        5        247532      65714       31791       12246        6246        7958        5898           0        1078         526
        6        186051      30806       29902           0        4812        2417        3282        1554         281         264
        7        100953       7476        9302        4107        3358        1038         438         629         240          82

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000     1791         0         0  1901602         0  1082298         0     89847         0    11944       849        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 6 MINUTES, 20 SECONDS.
```

```
***START BATCH    9                               RANDOM=363F30DB5BBA

SOURCE DATA
  WTAVE       IAVE    UAVE     VAVE     WAVE       XAVE        YAVE        ZAVE        AGEAVE
 9.687E-04    10.23   0.0044   0.0034   0.0008    0.0         0.0         0.0         0.0

4 PI R**2 FLUENCE

                                     RESPONSES(DETECTOR)  NEUTRON FLUX
   DETECTOR          BATCH                FSD           ACCUMULATED           FSD           CV(SD)            FOM
                    RESPONSE                              RESPONSE
      1           1.9292D+00           0.01005         1.9355D+00          0.00369         0.05057         1.2876D+03
      2           2.1031D+00           0.01235         2.0716D+00          0.00443         0.06004         8.9132D+02
      3           1.7628D+00           0.01144         1.7786D+00          0.00446         0.04096         8.7957D+02
      4           6.6494D-01           0.02153         6.5790D-01          0.00614         0.04018         4.6374D+02
      5           4.3820D 01           0.02111         4.3267D-01          0.00653         0.02528         4.1115D+02
      6           1.7186D 01           0.02725         1.7476D-01          0.00931         0.05359         2.0186D+02
      7           3.9749D-02           0.05783         4.0598D-02          0.02072         0.23381         4.0779D+01

   DETECTOR            PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0      10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0     100.0

      1          492473    210126         0     59699     48725     28672     26386         0      6104      2635
      2          478452    160782     59350     31469     43961     37351         0      5374      3505      2649
      3          472611    123689     49149     43934     26885     25905         0      4673      3124      2403
      4          343184    102559         0     19949     16905      9682      8490         0      2044      1038
      5          278473     73966     35804     13712      7017      8913      6642         0      1227       595
      6          209353     34641     33618         0      5427      2715      3676      1746       318       295
      7          113578      8439     10468      4628      3778      1172       485       690       270        91

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000      1699        0         0  1907716         0  1081844         0     89972        0     11727       811         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 6 MINUTES, 20 SECONDS.
```

```
**START BATCH  10                             RANDOM=6ARAA5FC1202

SOURCE DATA
  WTAVE         IAVE     CAVE     VAVE      WAVE       XAVE         YAVE         ZAVE         AGEAVE
 9.687E+04      10.23  -0.0008 -0.0001   0.0029     0.0          0.0          0.0          0.0

4 PI R**2 FLUENCE

                                      RESPONSES(DETECTOR)  NEUTRON FLUX
    DETECTOR          BATCH                FSD          ACCUMULATED          FSD          CV(SD)          FOM
                     RESPONSE                             RESPONSE
        1           1.9403D+00          0.01053          1.9360D+00          0.00348      0.04639        1.2998D+03
        2           2.1167D+00          0.01896          2.0762D+00          0.00442      0.07557        8.0479D+02
        3           1.7697D+00          0.01374          1.7777D+00          0.00424      0.03745        8.7482D+02
        4           6.6807D-01          0.01736          6.5892D-01          0.00580      0.03690        4.6913D+02
        5           4.3582D-01          0.02043          4.3298D 01          0.00622      0.02446        4.0742D+02
        6           1.7285D-01          0.04121          1.7457D-01          0.00933      0.08119        1.8099D+02
        7           3.7749D 02          0.03733          4.0313D-02          0.01910      0.22601        4.3177D+01

    DETECTOR             PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                    0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

        1        547397     233200          0      66455      54167      31751      29358          0       6761       2952
        2        531797     178649      65945      34897      48774      41508          0       5968       3934       2967
        3        525358     137393      54507      48902      22747      35862          0       5154       3475       2652
        4        381496     113951          0      22222      18850      10745       9453          0       2284       1154
        5        309433      82274      39828      15236       7791       9984       7363          0       1363        652
        6        232767      38494      37346          0       6042       2991       4070       1987        331        318
        7        126159       9453       9125       7702       4160       1308        535        753        298        100

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
  100000      1765         0         0  1914825         0  1084431         0     89874          0     11891       811         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 6 MINUTES, 21 SECONDS.
```

A.3 Sample Problem 3

A.3.1 Without Source Step Biasing

```
***START BATCH    1                              RANDOM-13579BDFDB97

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF GTMED WHICH ASSUMES GROMETRY AND XSECT MEDIA ARE IDENTICAL.
  WTAVE       JAVE     UAVE     VAVE     WAVE       XAVE        YAVE        ZAVE       AGEAVE
 1.000E+04    1.00   0.0050 -0.0036   0.4944   2.392E 01   1.607E-01   9.996E-05    0.0

 PARTICLES PER CM**2 PER SECOND

                              RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR         BATCH             FSD          ACCUMULATED          FSD          CV(SD)          FOM
                   RESPONSE                          RESPONSE
       1          3.3598D-09        0.16022         3.3598D 09        0.16022        0.09256        2.2377D+01


    DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1             9281       23          0          0          0          5          0          5          0          0


NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE    E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000         0        0         0   112485         0     12841      6122     3878        0        0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 1 MINUTE, 44 SECONDS.
```

```
***START BATCH    4                                 RANDOM-E56A1E7C7747

SOURCE DATA
  WTAVE          IAVE     UAVE     VAVE     WAVE        XAVE         YAVE         ZAVE        AGEAVE
 1.000E+04       1.00    0.0005   0.0072   0.5050   -5.096E-01   5.352E-01   9.976E-05    0.0

 PARTICLES PER CM**2 PER SECOND

                              RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR          BATCH           FSD          ACCUMULATED          FSD           CV(SD)           FOM
                     RESPONSE                        RESPONSE
        1            2.9534D-09     0.16513         3.5668D-09        0.09836        0.12074        1.4634D+01


    DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

        1          37191        103           0           0           0           0          21           0           2           3


NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL      ALBEDO      BDRYX     ESCAPE      E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
   10000          0         0         0   114911          0      12750       6109       3891          0          0          0         0
TIME REQUIRED FOR THE PRECEDING BATCH WAS 1 MINUTE, 46 SECONDS.
```

```
***START BATCH    8                             RANDOM=8E6E7EC9CACF

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE       XAVE        YAVE         ZAVE        AGEAVE
 1.000E+04     1.00    0.0012   0.0040   0.4985   2.509E-01   1.348E+00   9.996E-05    0.0

 PARTICLES PER CM**2 PER SECOND

                                     RESPONSES(DETECTOR)    NEUTRON FLUX
    DETECTOR           BATCH                FSD            ACCUMULATED             FSD              CV(SD)              FOM
                      RESPONSE                               RESPONSE
        1            3.0808D-09           0.16312            3.3484D-09           0.06688           0.07893            1.5792D+01

    DETECTOR              PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                    0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0       100.0

        1             74328        211           0           0           0           0          15          24           5           5

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000        0          0         0   113448         0     12917      6103      3897        0         0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 1 MINUTE, 45 SECONDS.
```

```
***START BATCH    9                          RANDOM-B2C9EE7DEBBF

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE       XAVE        YAVE        ZAVE       AGEAVE
 1.000E+04     1.00   -0.0063 -0.0054   0.4925   4.844E-01   6.450E-01   9.996E-05    0.0

 PARTICLES PER CM**2 PER SECOND

                             RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR          BATCH             FSD          ACCUMULATED          FSD          CV(SD)          FOM
                    RESPONSE                          RESPONSE
        1          3.3047D-09        0.23897         3.3436D-09        0.06507        0.08243        1.4827D+01

    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0       10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0     100.0

        1         83616      226         0         0         0         0        48         0         6         6

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000        0         0         0   114571         0     12820     6084      3916        0         0         0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 1 MINUTE, 46 SECONDS.
```

```
***START BATCH  10                           RANDOM-3A171C3BB4F7

SOURCE DATA
  WTAVE       TAVE    UAVE     VAVE     WAVE       XAVE        YAVE        ZAVE       AGEAVE
 1.000E+04    1.00  -0.0021  0.0031   0.4982  -5.126E-01  1.146E+00  9.996E-05   0.0

 PARTICLES PER CM**2 PER SECOND

                            RESPONSES(DETECTOR)    NEUTRON FLUX
   DETECTOR          BATCH          FSD           ACCUMULATED         FSD           CV(SD)          FOM
                    RESPONSE                       RESPONSE
       1          2.9991D-09      0.17266         3.3091D 09        0.06120        0.07737       1.5070D+01

   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                 0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1        92914       246          0          0          0          0         56          0          6          6

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL     ALBEDO     BDRYX     ESCAPE     E-CUT TIMEKILL R R KILL R R SURV   CAMLOST
   10000        0         0         0   115350          0     13177       6062      3938        0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 1 MINUTE, 47 SECONDS.
```

A.3.2 With Source Step Biasing

```
***START BATCH    1                         RANDOM=13579BDFDB97

SOURCE DATA
YOU ARE USING THE DEFAULT VERSION OF GTMED WHICH ASSUMES GEOMETRY AND XSECT MEDIA ARE IDENTICAL.
  WTAVE        IAVE    UAVE     VAVE     WAVE        XAVE        YAVE         ZAVE       AGEAVE
 1.051E+04     1.00   0.0049  0.0008   0.4940   -1.962E+00 -1.798E+00  1.001E-04    0.0

 PARTICLES PER CM**2 PER SECOND

                                RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR         BATCH             FSD           ACCUMULATED          FSD           CV(SD)          FOM
                   RESPONSE                            RESPONSE
        1         3.6444D-09        0.14482          3.6444D-09         0.14482        0.42367       2.2735D+01


    DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0       10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

        1          8896          0          0          0          0        636        212         32          1          0

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL     ALBEDO      BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV   GAMLOST
   10000        0          0         0   139497          0      25875      4895      5105        0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 5 SECONDS.
```

```
***START BATCH    4                    RANDOM-C8E89B10B42B

SOURCE DATA
  WTAVE       IAVE    UAVE    VAVE     WAVE       XAVE        YAVE        ZAVE       AGEAVE
 9.821E+03    1.00   0.0087 -0.0100   0.5000   1.636E+00   1.488E+00   1.001E-04   0.0

 PARTICLES PER CM**2 PER SECOND

                            RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR        BATCH           FSD         ACCUMULATED         FSD          CV(SD)          FOM
                  RESPONSE                       RESPONSE
       1         2.9534D-09       0.02186       3.1261D 09        0.04334        0.40199       6.2840D+01


    DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                  0.0      10.0     20.0     30.0     40.0     50.0     60.0     70.0     80.0     90.0    100.0

       1         35475        0        0        0        0     1167     1278      890      255       30


NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN   GAMGEN REALCOLL   ALBEDO    BDRYX    ESCAPE    E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000        0        0        0   141886        0    26247     4888     5112        0        0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 7 SECONDS.
```

```
***START BATCH    6                              RANDOM-/A4EEDEADA3B

SOURCE DATA
  WTAVE         IAVE     UAVE     VAVE     WAVE        XAVE        YAVE        ZAVE        AGEAVE
 9.718E+03      1.00    0.0196   0.0123   0.4962   -1.058E+00   5.241E+00   1.001E 04     0.0

 PARTICLES PER CM**2 PER SECOND

                                         RESPONSES(DETECTOR)   NEUTRON FLUX
   DETECTOR           BATCH                    FSD           ACCUMULATED          FSD            CV(SD)            FOM
                     RESPONSE                                  RESPONSE
       1            3.4362D-09               0.06156           3.1387D-09         0.02471         0.30883         9.5769D+01


   DETECTOR          PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0        10.0       20.0       30.0       40.0       50.0       60.0       70.0       80.0       90.0      100.0

       1          70879          0          0          0          0       2374       2538       1836        580         40


NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)       FISHN    GAMGEN REALCOLL     ALBEDO      BDRYX    ESCAPE     E CUT TIMEKILL R R KILL R R SURV   GAMLOST
   10000          0          0          0   143372          0      26648      4737      5263          0          0          0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 9 SECONDS.
```

```
***START BATCH    9                         RANDOM-7AA14D712B1F

SOURCE DATA
 WTAVE        IAVE    UAVE    VAVE    WAVE       XAVE        YAVE        ZAVE       AGEAVE
1.064E+04     1.00   0.0058  0.0070  0.5098   9.696E-01   1.867E+00   1.001E-04    0.0

 PARTICLES PER CM**2 PER SECOND

                                   RESPONSES(DETECTOR)   NEUTRON FLUX
    DETECTOR          BATCH              FSD           ACCUMULATED          FSD           CV(SD)          FOM
                     RESPONSE                            RESPONSE
       1            2.9282D-09         0.02840          3.1153D-09        0.02233        0.30345       1.0446D+02

    DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0      10.0      20.0      30.0      40.0      50.0      60.0      70.0      80.0      90.0      100.0

       1          79739       0         0         0         0       2650      2863      2080       612        72

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)     FISHN    GAMGEN REALCOLL    ALBEDO    BDRYX    ESCAPE     R-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000        0         0         0   140324         0    26042     4881      5119        0         0        0        0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 6 SECONDS.
```

```
***START BATCH  10                          RANDOM-2B3CE58136FF

SOURCE DATA
  WTAVE        IAVE     UAVE     VAVE     WAVE        XAVE        YAVE        ZAVE        ACEAVE
 9.892E+03     1.00   0.0109  -0.0012   0.4969   -2.111E+00   3.172E+00   1.001E-04    0.0

 PARTICLES PER CM**2 PER SECOND

                                   RESPONSES(DETECTOR)    NEUTRON FLUX
    DETECTOR            BATCH            FSD           ACCUMULATED          FSD           CV(SD)            FOM
                       RESPONSE                          RESPONSE
        1             4.7195D-09       0.24259          3.2757D-09        0.03984        0.24139         2.9548D+01

    DETECTOR           PARTICLE DISTRIBUTION IN PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF CONTRIBUTIONS

                   0.0        10.0        20.0        30.0        40.0        50.0        60.0        70.0        80.0        90.0        100.0

        1         88620          0           0           0           0        2940        5474         457         287           4

NUMBER OF COLLISIONS OF TYPE NCOLL
  SOURCE SPLIT(D)      FISHN    GAMGEN REALCOLL    ALBEDO     BDRYX    ESCAPE     E-CUT TIMEKILL R R KILL R R SURV  GAMLOST
   10000        0          0         0   141903         0     26173      4843      5157         0         0         0         0

TIME REQUIRED FOR THE PRECEDING BATCH WAS 2 MINUTES, 7 SECONDS.
```

Appendix B

Another Approach to the Monte Carlo Point-Estimator Concept

The usual formulation of the point-detector estimator is given by:

$$CON = W \times P_s \times P(\Omega \rightarrow \Omega') \frac{e^{-\Sigma_t r}}{r^2}, \tag{B.1}$$

where W is the particle current weight, P_s the non-absorption probability, $P(\Omega \rightarrow \Omega')$ the probability per steradian of scattering into the detector direction, $e^{-\Sigma_t r}$ the probability of hitting the detector, and r the distance between the collision point and the detector position.

The approach proposed in this work is based on the fact that instead of using $P(\Omega \rightarrow \Omega')$ on a steradian basis, $P(\Omega \rightarrow \Omega')$ is interpreted in terms of a probability that the particle hits a sphere of unit cross section area placed in the detector position as shown in Figure B.1-a. This eliminates the need for the $1/r^2$ factor and the contribution becomes

$$CON = W \times P_s \times P'(\Omega \rightarrow \Omega') e^{-\Sigma_t r}. \tag{B.2}$$

This is accomplished by calculating the solid angle formed by the sphere of unit cross section area at the detector position and the collision point, i.e

$$SA = \frac{1}{4\pi} \int_0^{2\pi} d\varphi \int_0^{\theta} \sin\theta d\theta = \frac{1}{2}(1 - \cos\theta) \tag{B.3}$$

Table B.1: Comparison of the Calculated Results.

Detector	SXE	SPD	FSD	NPD	FSD
1	3.8269E-04	3.6421E-04	0.0071	3.9522E-04	0.0164
2	4.4637E-05	3.9779E-05	0.0228	4.3999E-05	0.0512
3	8.9468E-06	8.1489E-06	0.0363	8.7902E-06	0.0754
4	2.2423E-06	2.1442E-06	0.0822	2.1719E-06	0.0865
5	6.2343E-07	7.0446E-07	0.1363	7.1119E-07	0.1392
6	1.8914E-07	1.9343E-07	0.2276	1.9432E-07	0.2301
7	5.9068E-08	5.5311E-08	0.2102	5.5492E-08	0.2118
8	1.9177E-08	1.4919E-08	0.1699	1.4935E-08	0.1699
9	6.2713E-09	4.2579E-09	0.2194	4.2640E-09	0.2195
10	2.0408E-09	1.0450E-09	0.2492	1.0464E-09	0.2490

and the new probability $P'(\Omega \to \Omega')$ is given by

$$P'(\Omega \to \Omega') = 4\pi P(\Omega \to \Omega')\frac{1}{2}(1 - \cos\theta). \tag{B.4}$$

The factor 4π eliminates the per steradian basis of $P(\Omega \to \Omega')$ and the angle θ is shown in Figure B.1-b.

It can be seen from Figure B.1-b that whenever r is less than r', θ assumes the value $\pi/2$ so that the emergent particle will be normally incident to a unit area in the plane that contains the detector and the collision point and the contribution becomes simply $W/2$.

Table B.1 shows the results of the calculations for Sample Problem 1 but with the first three groups being analyzed. The results for the standard point detector estimator (SPD) were obtained using Equation B.1 with a unboundness correction scheme which uses an average contribution when particles collide within a distance of 0.5642 cm off the detector position. The calculations for Detectors 1 and 2 show a very good agreement for results with acceptable standard deviations. However, the accuracy of the new formalism (NPD) is better when compared with the results obtained with a surface crossing estimator (SXE). The results for the outer detectors, although with unacceptable standard deviations ($FSD > 0.5$), are presented to illustrate the excellent agreement between the two point detector estimators.

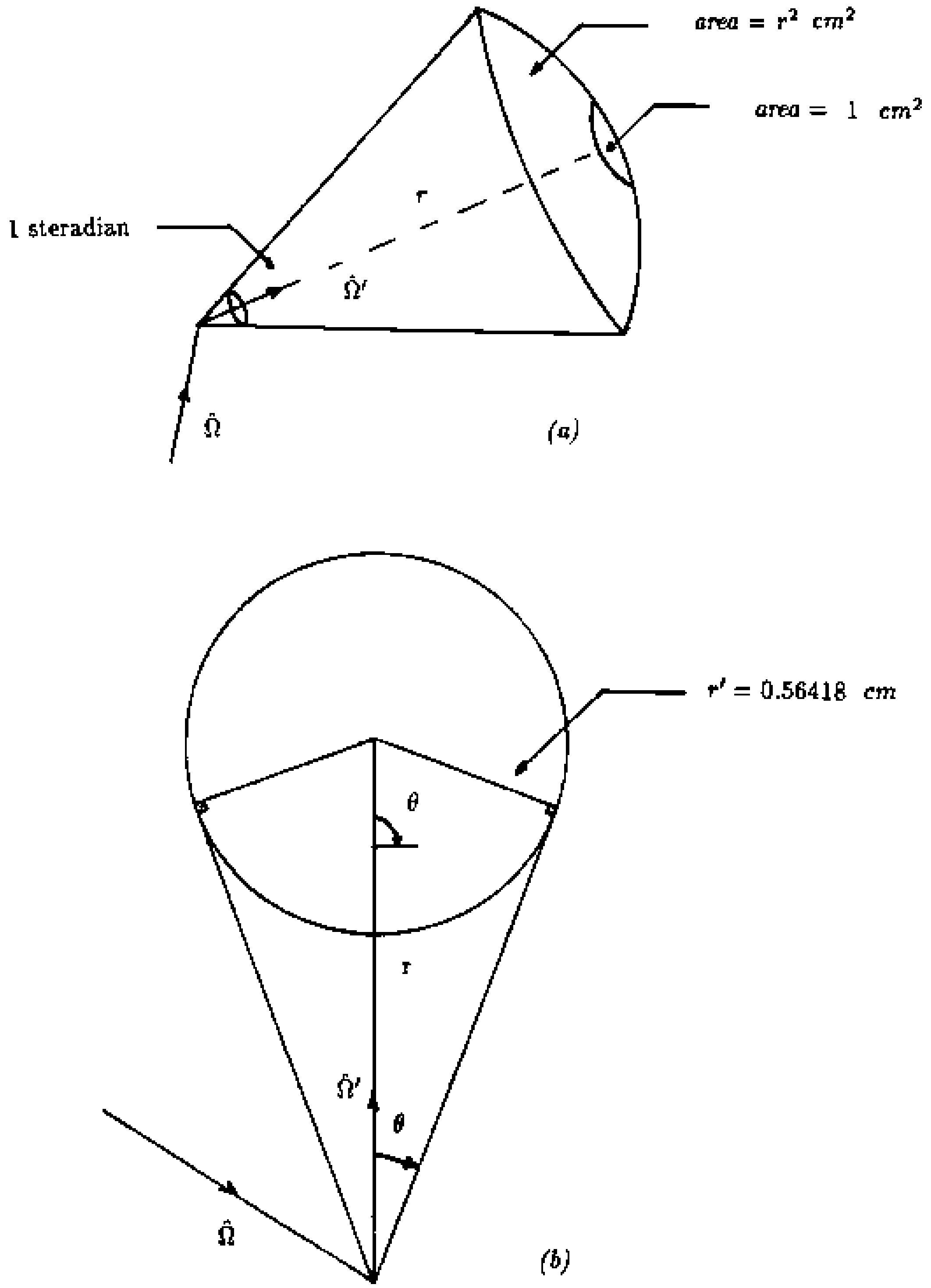

Figure B.1: Schemes of the New Point Detector Estimator.

Appendix C

Modified and Added Subroutines

The subroutines listed in this appendix are meant to be used as a guideline for the reader if he wants to implement the same modifications in his or her version of the MORSE code.

```
C**** WILSON   MAY  11, 1989 ****                                       MOR 1150
C * * THIS IS THE MAIN ROUTINE * * * * * * * * * * * * * * * * * * * * * * * * *
C * *
C * * THE FOLLOWING CARD DETERMINES THE SIZE ALLOWED FOR BLANK COMMON  * * * * *
C * *  (REGION SIZE NEEDED IS ABOUT 150K + 4*(SIZE OF BLANK COMMON IN WORDS) ) *
C * * NOTE - THE ORDER OF COMMONS IN THIS ROUTINE IS IMPORTANT AND MUST CORRES-
C * *   POND TO THE ORDER USED IN DUMP ROUTINES SUCH AS HELP, XSCHLP, AND USRHLP
C * *
C * * LABELLED COMMONS FOR WALK ROUTINES    * * * * * * * * * * * * * * * * * *
      REAL * 8 SS
      COMMON NC( 50000)
      COMMON /APOLLO/ AGSTRT,DDF,DEADWT(26),ITOUT,ITIN
      COMMON /FISBNK/ MFISTP
      COMMON /NUTRON/ NAME
C * *
C * * LABELLED COMMONS FOR CROSS-SECTION ROUTINES  * * * * * * * * * * * * * * *
     . COMMON /LOCSIG/ ISCCOG
      COMMON /MEANS/ NM
      COMMON /MOMENT/ NMOM
      COMMON /QAL/ Q
      COMMON /RESULT/ POINT
C * *
C * * LABELLED COMMONS FOR GEOMETRY INTERFACE ROUTINES  * * * * * * * * * * * *
      COMMON /NORMAL/ UNORM
C * *
C * * LABELLED COMMONS FOR USER ROUTINES  * * * * * * * * * * * * * * * * * * *
      COMMON /PDET/ ND
      COMMON /USER/ AGST
C * *
C * * COMMON /DUMMY/ WILL NOT BE FOUND ELSEWHERE IN THE PROGRAM * * *
      COMMON /DUMMY/ DUM
      COMMON /SSS/ SS(1000)
C * *
      DATA JUNK/Z40404040/
```

```
      NLFT = (LOC(DUM) - LOC(NC(1)))/4
      DO 10 I=1,NLFT
   10 NC(I) = JUNK
      ITOUT = 6
      ITIN = 5
      NLFT = (LOC(AGSTRT) - LOC(NC(1)))/4
      CALL MORSE(NLFT)
      STOP
      END
      SUBROUTINE RELCOL                                                 RELCO 10
C * * THIS VERSION IS FOR USE WITH MORSE-CGA * * * * * * *
C                                                                       RELCO 20
C   THIS VERSION IS FOR POINT DETECTORS LOCATED AT (XD,YD,ZD)           RELCO 30
C                                                                       RELCO 40
      COMMON /USER/ AGSTRT,WTSTRT,XSTRT,YSTRT,ZSTRT,DFF,EBOTN,EBOTG,    RELCO 50
     1 TCUT,IO,I1,IADJM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,NLAST,  RELCO 51
     2 NLEFT,NMGP,NMTG,NSTRT                                            RELCO 52
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,RELCO 60
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   RELCO 61
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                RELCO 62
      COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDRELCO 70
     1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGERELCO 71
      COMMON BL(1)                                                      RELCO 80
      DIMENSION NL(1)                                                   RELCO 90
      EQUIVALENCE (BL(1),NL(1))                                         RELC 100
      DATA NEST /1/, FNEST /1./                                         RELC 110

C   NEST + FNEST ARE THE NO. OF ESTIMATES TO BE MADE TO EACH DETECTOR   RELC 130
C * * * ISTAT MUST BE EQUAL TO 1.                                       * * * *
C * * * NEX MUST BE AT LEAST 1                                          * * * *
C * * * NEXND MUST BE AT LEAST 1                                        * * * *
      DO 30 I=1,ND                                                      RELC 160
      IA=LOCXD+I                                                        RELC 170
      XE = BL(IA)                                                       RELC 180
      YE = BL(IA+ND)                                                    RELC 190
      ZE = BL(IA+2*ND)                                                  RELC 200
C * * PTPT CALCULATES DISTANCE TO DETECTOR USING GLOBAL COORD. * *
      CALL PTPT(XE,YE,ZE,A,B,C,THETA,BL,BL)
C     A = XE - X    - CALCULATED IN PTPT * *                            RELC 210
C     B = YE - Y    - CALCULATED IN PTPT * *                            RELC 220
C     C = ZE - Z    - CALCULATED IN PTPT * *                            RELC 230
      SD2=A*A+B*B+C*C                                                   RELC 240
      DS=SQRT (SD2)                                                     RELC 250
C * * * COS DEPENDS ON THE ANGLE OF INTEREST                            * * * *
      COS=C/DS                                                          RELC 270
C     THETA = (A*UOLD + B*VOLD + C*WOLD)/DS                             RELC 280
      IGOLD = IGO                                                       RELC 290
      IGQ = NGPQT3                                                      RELC 300
      IF (IGO.LE.NGPQT1) IGQ=NGPQT1                                     RELC 310
      IA = LOCRSP + NRESP*NMTG + 1                                      RELC 320
      CALL PTHETA(NMED,IGOLD,IGQ,THETA,BL(IA),NMTG)                     RELC 330
      NES = 0                                                           RELC 340
```

```
      PSUM = 0.                                                         RELC 350
      IA = IA - 1                                                       RELC 360
      DO 5 IL=IGOLD,IGQ                                                 RELC 370
    5 PSUM = PSUM + ABS (BL(IA+IL))                                     RELC 380
   10 R = FLTRNF(0) * PSUM                                              RELC 390
      DO 15 IL=IGOLD,IGQ                                                RELC 400
      IF (R-ABS (BL(IA+IL))) 20,20,15                                   RELC 410
   15 R = R - ABS (BL(IA+IL))                                           RELC 420
      IL = IGQ                                                          RELC 430
   20 MARK=1                                                            RELC 440
      AGED = AGE + DS/BL(NMTG+IL)                                       RELC 450
      MEDIUM=NMED                                                       RELC 470
      CALL EUCLID(MARK,X,Y,Z,XE,YE,ZE,DS,IL,ARG,0,MEDIUM,BLZNT,NREG)    RELC 480
      CON = 0.0
      IF (ARG.LT.-1.E+64) GO TO 24                                      RELC 490
C********BEWARE THIS VERSION WILL NOT WORK IF ENERGY BIASING IS USED     * * * *
C************WILSON 11/14/88 *****************
C************* CORRECTED FOR THE UNBOUNDED CONTRIBUTION
C      CON = WATE*EXP (ARG)*SIGN (PSUM,BL(IA+IL))/FNEST                 RELC 510
      R1=0.564189584
      ST= R1/DS
      IF(ST.GT.1.0) GO TO 7
C      CALL NSIGTA(IG,NMED,TSIG,P)
C      DS1 = ARG-R1*TSIG
      CON = WATE*EXP (ARG)*SIGN (PSUM,BL(IA+IL))/FNEST                  RELC 510
      CT=SQRT(SD2-R1*R1)/DS
      CON=CON*4.*3.1415962*(1-CT)/2.
      GO TO 24
    7 CON = WATE*0.5
C************WILSON 11/14/88 *****************
   24 CALL FLUXST (I,IL,CON,AGED,COS,0)                                 RELC 530
   25 NES = NES + 1                                                     RELC 540
      INN=LOCXD+6*ND+I                                                  RELC 550
      NL(INN)=NL(INN)+1                                                 RELC 560
      IF (NES-NEST) 10,30,30                                            RELC 570

   30 CONTINUE                                                          RELC 580
      RETURN                                                            RELC 590
      END                                                               RELC 600
C*****************************************************************      BDRY 380
C                                                               *       BDRY 380
C                                                               *       BDRY 380
C                                                               *       BDRY 380
C       SUBROUTINES OF INTEREST                                 *       BDRY 380
C                                                               *       BDRY 380
C       FBANK           NOT MODIFIED
C       FLUXST          MODIFIED
C       GETNT           NOT MODIFIED
C       GSTORE          MODIFIED
C       INPUT1              "
C       MORSE               "
C       MSOUR               "
```

```
C        NBATCH            "
C        NPART         ADDED
C        NRNPRI        ADDED
C      - NRUN          MODIFIED
C        OUTPT         MODIFIED
C        SCORIN            "
C        STBTCH            "
C        STPART        ADDED
C        VAR2          MODIFIED
C        VAR3          MODIFIED
C        VAR4          ADDED
C                                                                  *      BDRY 380
C                                                                  *      BDRY 380
C                                                                  *      BDRY 380
C******************************************************************      BDRY 380
      SUBROUTINE FLUXST(I,IGE,FLUX,AGE,COS,SWITCH)                        FLUXS 10
      REAL * 8 SS, SCORE,FLUX
      INTEGER*4 SWITCH                                                    FLUXS 20
C  *  *  SWITCH =  1  -- STORE IN ALL RELEVANT ARRAYS                     FLUXS 30
C  *  *  SWITCH =  0  -- STORE IN ALL RELEVANT ARRAYS EXCEPT UD           FLUXS 40
C  *  *  SWITCH = -1  -- STORE IN ARRAY UD ONLY                           FLUXS 50
      COMMON /USER/ AGSTRT,WTSTRT,XSTRT,YSTRT,ZSTRT,DFF,EBOTN,EBOTG,      FLUXS 60
     1 TCUT,IO,I1,IADJM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,NLAST,    FLUXS 61
     2 NLEFT,NMGP,NMTG,NSTRT                                              FLUXS 62
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,FLUXS 70
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,     FLUXS 71
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                  FLUXS 72
      COMMON BC(1)                                                        FLUXS 80
      COMMON /SSS/ SS(1)
      DIMENSION NC(1)                                                     FLUXS 90
      EQUIVALENCE (BC(1),NC(1))                                           FLUX 100
      DATA K/1/                                                           FLUX 110
      IF(IGE.GT.NMTG) GO TO 40                                            FLUX 115
      IF (SWITCH) 170,10,10                                               FLUX 120
 10   IF (NE) 60,60,20                                                    FLUX 130
   20 J=NC(LOCIB+3*NE+IGE)                                                6-15-71
      IF(J.LE.0) RETURN                                                   FLUX 160
 50   IB = LOCQE + (I-1)*NE + J                                           FLUX 200
      SS(IB) = SS(IB) + FLUX                                              FLUX 210
 60   IF (NT) 90,90,70                                                    FLUX 220
 70   IA = LOCT + (I-1)*NT + 1                                            FLUX 230
      DO 80 K=1,NT                                                        FLUX 240
      IF (AGE-BC(IA)) 90,90,80                                            FLUX 250
 80   IA = IA + 1                                                         FLUX 260

      K = NT                                                              FLUX 270
      IA=IA-1
      BC(IA)=AGE
      IDT = LOCT + (ND+I)*NT                                              FLUX 290
      BC(IDT) = BC(IA) - BC(IA-1)                                         FLUX 300
 90   IF (NA) 120,120,100                                                 FLUX 310
 100  IA = LOCCO + 1                                                      FLUX 320
```

```
      DO 110 L=1,NA                                                     FLUX 330
      IF (COS-BC(IA)) 120,120,110                                       FLUX 340
  110 IA = IA + 1                                                       FLUX 350
   40 CALL HELP(4HFXST,1,1,-1,1)                                        FLUX 180
      CALL ERROR                                                        FLUX 190
  120 IF (NE) 170,170,130                                               FLUX 370
  130 IF (NT) 150,150,140                                               FLUX 380
  140 ID = LOCQTE + (I-1)*NTNE  + (J-1)*NT + K                          FLUX 390
      SS(ID) = SS(ID) + FLUX                                            FLUX 400
  150 IF (NA) 170,170,160                                               FLUX 410
  160 IE = LOCQAE + (I-1)*NANE  + (J-1)*NA + L                          FLUX 420
      SS(IE) = SS(IE) + FLUX                                            FLUX 430
  170 IS = (I-1)*NRESP + 1                                              FLUX 440
      IU = LOCUD + IS                                                   FLUX 450
      IS = LOCSD + IS                                                   FLUX 460
C**** WILSON   MAY  11, 1989 ****                                       MOR 1150
      IW = LOCSD + (I-1)*NRESP + 3*NDNR + 1                             FLUX 460
      IC = LOCQT + (I-1)*NTNR + (K-1)*NRESP + 1                         FLUX 470
      IB = LOCRSP + IGE - NMTG                                          FLUX 480
      DO 220 IR = 1,NRESP                                               FLUX 490
      IB = IB + NMTG                                                    FLUX 500
      SCORE = FLUX*BC(IB)                                               FLUX 510
      IF (SWITCH) 180,200,180                                           FLUX 520
  180 SS(IU) = SS(IU) + SCORE                                           FLUX 530
      IU = IU + 1                                                       FLUX 540
  190 IF (SWITCH) 220,200,200                                           FLUX 550
  200 SS(IS) = SS(IS) + SCORE                                           FLUX 560
      SS(IW) = SS(IW) + SCORE                                           FLUX 560
      IS = IS + 1                                                       FLUX 570
      IW = IW + 1                                                       FLUX 570
C**** WILSON   MAY  11, 1989 ****                                       MOR 1150
      IF (NT) 220,220,210                                               FLUX 580
  210 SS(IC) = SS(IC) + SCORE                                           FLUX 590
      IC = IC + 1                                                       FLUX 600
  220 CONTINUE                                                          FLUX 610
      RETURN                                                            FLUX 620
      END                                                               FLUX 630
      SUBROUTINE GSTORE(WGG,IGG)                                        GSTOR 10
C * * THIS VERSION OF GSTORE IS FOR MORSE-CGA * * * *
C     THIS ROUTINE CHECKS TO SEE IF THERE IS ROOM IN THE BANK AND IF SO GSTOR  *
C        STORES DATA FOR THE GENERATED GAMMA                            GSTOR  *
C     IT ASSUMES THAT THE GAMMA IS EMITTED UNIFORMLY IN DIRECTION       GSTOR  *
      COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDGSTOR 20
     1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGEGSTOR 21
      COMMON /APOLLO/ AGSTRT,DDF,DEADWT(5),ETA,ETATH,ETAUSD,UINP,VINP,  GSTOR 30
     1 WINP,WTSTRT,XSTRT,YSTRT,ZSTRT,TCUT,XTRA(10),                     GSTOR 31
     2   IO,I1,MEDIA,IADJM,ISBIAS,ISOUR,ITERS,ITIME,ITSTR,LOCWTS,LOCFWL,GSTOR 32
     3 LOCEPR,LOCNSC,LOCFSN,MAXGP,MAXTIM,MEDALB,NGPREG,MXREG,NALB,      GSTOR 33
     4 NDEAD(5),NEWNM,NGEOM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,    GSTOR 34
     5 NKCALC,NKILL,NLAST,NMEM,NMGP,NMOST,NMTG,NOLEAK,NORMF,NPAST,      GSTOR 35
     6 NPSCL(13),NQUIT,NSIGL,NSOUR,NSPLT,NSTRT,NXTRA(10)                GSTOR 36
```

```
C**** WILSON   MAY  11, 1989 ****                                     MOR 1150

      COMMON /NS/ NS                                                  GSTOR 40
C**** WILSON   MAY  11, 1989 ****                                     MOR 1150
      COMMON BC(1)                                                    GSTOR 40
      DIMENSION NC(1)                                                 GSTOR 50
      EQUIVALENCE (BC(1),NC(1))                                       GSTOR 60
      IF (NMOST-NS) 10,10,15                                          GSTOR 70
 10   IF (NPSCL(13) .EQ. 0) WRITE (IO,1010)                           GSTOR 80
 1010 FORMAT ('0WARNING * * * NO ROOM IN BANK FOR SECONDARIES * * * *'/)GSTOR 90
      NPSCL(13) = NPSCL(13) + 1                                       GSTO 100
      RETURN                                                          GSTO 110
 15   NS = NS + 1                                                     GSTO 120
      NEWNM = NEWNM + 1                                               GSTO 130
      NAS = NAME                                                      GSTO 140
      WTS = WATE                                                      GSTO 150
      IGS = IG                                                        GSTO 160
      NAME = NEWNM                                                    GSTO 170
      WATE = W8G                                                      GSTO 180
      IG = IGG                                                        GSTO 190
      US=U                                                            GSTO 200
      VS=V                                                            GSTO 210
      WS=W                                                            GSTO 220
      CALL GTISO(U,V,W)                                               GSTO 230
      CALL STORNT(NS,0)                                               GSTO 240
C     CALL BANKR(4)  FOR GAMMA GENERATION ANALYSIS                    * * * *
      CALL BANKR(4)                                                   GSTO 245
      NAME = NAS                                                      GSTO 250
      WATE = WTS                                                      GSTO 260
      IG = IGS                                                        GSTO 270
      U=US                                                            GSTO 280
      V=VS                                                            GSTO 290
      W=WS                                                            GSTO 300
      ISCT = LOCNSC + 5*NMTG*MXREG + (NREG-1)*NMTG + IG               GSTO 310
      NC(ISCT) = NC(ISCT) + 1                                         GSTO 320
      ISCT = ISCT + IGG - IG                                          GSTO 330
      NC(ISCT) = NC(ISCT) + 1                                         GSTO 340
      ISCT = ISCT + NMTG*MXREG                                        GSTO 350
      BC(ISCT) = BC(ISCT) + W8G                                       GSTO 360
      ISCT = ISCT + IG - IGG                                          GSTO 370
      BC(ISCT) = BC(ISCT) + WATE                                      GSTO 380
      NPSCL(4) = NPSCL(4) + 1                                         GSTO 390
      RETURN                                                          GSTO 410
      END                                                             GSTO 420
      SUBROUTINE INPUT1
C * * THIS VERSION OF INPUT1 IS FOR MORSE-CGA * * *
C     THIS ROUTINE READS THE RANDOM WALK DATA AND CALLS ROUTINES TO READ * * * *
C      SOURCE DATA AND GEOMETRY DATA.       INITIALIZES SOME VARIABLES   * * * *


C**** WILSON   MAY  11, 1989 ****                                     MOR 1150
 1041 FORMAT (2X,I8,8I5,F5.0,2I5)
```

```
 1040 FORMAT (10I5,F5.0,2I5)
C**** WILSON   MAY  11, 1989 ****                                      MOR 1150
      WRITE(IO,1050)NSTRT,NMOST,NITS,NQUIT,NGPQTN,NGPQTG,NMGP,NMTG,NCOLTINPU 960
     1P,IADJM, AXTIM,MEDIA,MEDALB                                      INPU
 1050 FORMAT('0',3X,'NSTRT',3X,'NMOST',4X,'NITS',3X,'NQUIT',2X,        INPU
     1'NGPQTN',2X,'NGPQTG',4X,'NMGP'.4X,'NMTG',2X,'NCOLTP',3X,'IADJM'   INPU
     2,2X,'MAXTIM',3X,'MEDIA',2X,'MEDALB',/1X,10I8,F8.2,2I8)           INPU

      SUBROUTINE MORSE(NLFT)                                           MORSE 10
C     THIS IS THE EXECUTIVE ROUTINE FOR THE RANDOM WALK PROCESS * * *  MORSE  *

C         IT CONTROLS THE SUCCESSION OF EVENTS WHICH COMPRISE THE      MORSE  *
C     *        *    MONTE CARLO PROCESS  *    *    *      *     *      MORSE  *
C **** THIS VERSION OF SUBROUTINE MORSE KILLS JOBS ON IO REQUESTS*******
C *** THIS VERSION OF MORSE IS FOR MORSE-CGA * * * *
      REAL*8 RANDOM, SS                                                MORSE 15
      COMMON /APOLLO/ AGSTRT,DDF,DEADWT(5),ETA,ETATH,ETAUSD,UINP,VINP,  MORSE 20
     1 WINP,WTSTRT,XSTRT,YSTRT,ZSTRT,TCUT,XTRA(10),                    MORSE 21
     2   IO,I1,MEDIA,IADJM,ISBIAS,ISOUR,ITERS,ITIME,ITSTR,LOCWTS,LOCFWL,MORSE 22
     3 LOCEPR,LOCNSC,LOCFSN,MAXGP,MAXTIM,MEDALB,NGPREG,MXREG,NALB,     MORSE 23
     4 NDEAD(5),NEWNM,NGEOM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,    MORSE 24
     5 NKCALC,NKILL,NLAST,NMEM,NMGP,NMOST,NMTG,NOLEAK,NORMF,NPAST,      MORSE 25
     6 NPSCL(13),NQUIT,NSIGL,NSOUR,NSPLT,NSTRT,NXTRA(10)                MORSE 26
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP
      COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDMORSE 30
     1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGEMORSE 31
      COMMON /FISBNK/ MFISTP,NFISBN,NFISH,FTOTL,FWATE,WATEF            MORSE 40
C**** WILSON   MAY  11, 1989 ****                                      MOR 1150
      COMMON /NS/ NS                                                   GSTOR 40
      COMMON /SSS/ SS(1)                                               GSTOR 40
      COMMON /OUTB/ SWATE,XAVE,YAVE,ZAVE,EAVE,UAVE,VAVE,WAVE,AGEAVE     GSTOR 40
C**** WILSON   MAY  11, 1989 ****                                      MOR 1150
      DIMENSION NTS(1)                                                 MORSE 50
      COMMON WTS(1)                                                    MORSE 60
      EQUIVALENCE(WTS(1),NTS(1))                                       MORSE 70
      CALL MSGR(IO)                                                    MORS
C  BEGIN NEW PROBLEM                                                   MORSE 80
 10   NLAST=NLFT                                                       MORSE 90
      CALL TIMER(-2,ITRA)                                              MORS 100
      NXT = ICLOCK(0)                                                  MORS 110
      CALL INPUT                                                       MORS 120
C       READS CARDS A THRU D - CALLS SORIN FOR CARDS E IF ISOUR .LE. ZERO   140
C       READS CARDS F THRU Q - CALLS JOMIN, XSEC AND SCORIN            MORS 150
      CALL IOLEFT(NIO)
C ** A DUMMY IOLEFT IS PROVIDED FOR SITES NOT HAVING THIS CAPABILITY*
C   NIO IS TOTAL NO. IO'S LEFT AFTER INPUT ROUTINES FINISH
      WRITE(IO,1001) NIO
 1001 FORMAT(1X,'NUMBER OF IO REQUESTS LEFT AFTER INPUT IS ',I10)
      NIOB=NIO
      NTIO=0
```

```
      ISIG=1
      NTSTW=NKILL+NSPLT                                                 MORS 150
      NGPREG = NMTG*MIREG                                               MORS 160
      NXTRA(6) = NGPREG
      NCOMB = NGPQTN*NGPQTG                                             MORS 170
      IA = LOCFWL + 1                                                   MORS 180
      IU = LOCFWL + MIREG                                               MORS 190
      DO 15 I=IA,IU                                                     MORS 200
      IB = I + MIREG                                                    MORS 210
   15 WTS(IB) = WTS(I)                                                  MORS 220
      IAW = LOCWTS + 1                                                  MORS 230
      IUW = LOCWTS + 3*MGPREG                                           MORS 240
      DO 20 I=IAW,IUW                                                   MORS 250
      IB = I + 12*MGPREG                                                MORS 260
   20 WTS(IB) = WTS(I)                                                  MORS 270
      INITS = NITS                                                      MORS 280
      IRUNS = NQUIT
      INDX=0
      CALL TIMER(INDX,XTRA)                                             MORS 310
      WRITE (IO,1010) (XTRA(I),I=1,INDX)                                MORS 320

 1010 FORMAT (29H0TIME REQUIRED FOR INPUT WAS ,10A4)                    MORS 330
C    BEGIN NEW RUN                                                      MORS 320
   25 NITS = INITS                                                      MORS 350
      CALL BANKR(-1)                                                    MORS 360
      DO 30 I=IA,IU                                                     MORS 370
      IB = I + MXREG                                                    MORS 380
   30 WTS(I) = WTS(IB)                                                  MORS 390
      DO 35 I=IAW,IUW                                                   MORS 400
      IB = I + 12*MGPREG                                                MORS 410
   35 WTS(I) = WTS(IB)                                                  MORS 420
      ITERS=NITS                                                        MORS 430
      ITSTR=0                                                           MORS 440
C**** WILSON   APRIL 19,1989 ****                                       MOR 1150
      ITIMEI=ICLOCK(0)                                                  MORS 470
C  BEGIN NEW BATCH                                                      MORS 420
   40 NMEM=NSTRT                                                        MORS 460
      SWATE=0.                                                          OUTP 220
      XAVE=0.                                                           OUTP 230
      YAVE=0.                                                           OUTP 240
      ZAVE=0.                                                           OUTP 250
      UAVE=0.                                                           OUTP 260
      VAVE=0.                                                           OUTP 270
      WAVE=0.                                                           OUTP 280
      AGEAVE=0.                                                         OUTP 290
      EAVE=0.                                                           OUTP 300
C**** WILSON   APRIL 19,1989 ****                                       MOR 1150
      IF (ITSTR) 45,50,45                                               MORS 470
   45 NMEM = NFISH                                                      MORS 480
   50 CALL BANKR(-2)                                                    MORS 490
C       CALLS STBTCH                                                    * * * *
      NBATCH=NITS-ITERS+1                                               MORS 500
```

```
      CALL RNDOUT(RANDOM)                                               MORS 601
      WRITE (IO,1015) NBATCH,RANDOM                                     MORS 602
 1015 FORMAT (15H1***START BATCH ,I4,25X,7HRANDOM=,Z12/12H0SOURCE DATA) MORS 603
 60   CALL MSOUR                                                        MORS 610
C  BEGIN NEW HISTORY                                                    MORS 610
      NS = 1
      NMEM = NMEM - 1                                                   MORS 660
 61   CALL GETWT(NS,1)                                                  MORS 550
      NS = NS - 1                                                       MORS 560
      NALB = 0                                                          MORS 570
      NGPQT = NGPQT1                                                    MORS 580
 65   IF (WATE) 70,165,70                                               MORS 590
 70   IF(IG-NGPQT)90,90,75                                              MORS 600
 75   IF(IG-NGPQT2)160,160,80                                           MORS 610
 80   IF(IG-NGPQT3)85,85,160                                            MORS 620
 85   NGPQT=NGPQT3                                                      MORS 630
 90   IGO=IG                                                            MORS 640
      UOLD=U                                                            MORS 650
      VOLD=V                                                            MORS 660
      WOLD=W                                                            MORS 670
      OLDWT=WATE                                                        MORS 680
      XOLD=X                                                            MORS 690
      YOLD=Y                                                            MORS 700
      ZOLD=Z                                                            MORS 710
      BLZON=BLZNT                                                       MORS 720
      MEDOLD=NMED                                                       MORS 730
      OLDAGE=AGE                                                        MORS 740
      IF(NTSTV.GT.0) CALL TESTW                                         MORS 750
      IF(WATE)100,95,100                                                MORS 760

 95   NDEAD(1)=NDEAD(1)+1                                               MORS 770
      DEADWT(1)=DEADWT(1)+OLDWT                                         MORS 780
C  R R KILL                                                             MORS 790
      GO TO 165                                                         MORS 800
 100  CALL NXTCOL                                                       MORS 810
      IF (NREG-MXREG) 102,102,101                                       MORS
 101  WRITE (IO,1016) NREG,MXREG                                        MORS
 1016 FORMAT  (6H0NREG=,I5,8H, MXREG=,I5,61H, MXREG ON CARD I MUST BE GEMORS
     1 TO THE NUMBER OF REGIONS DESCRIBED IN GEOMETRY INPUT)            MORS
      CALL EXIT                                                         MORS
 102  IF (TCUT.LE.0..OR.AGE.LT.TCUT) GO TO 110                          MORS
      NDEAD(4)=NDEAD(4) +1                                              MORS 840
      DEADWT(4) =DEADWT(4)+OLDWT                                        MORS 850
C  AGE KILL                                                             MORS 820
      NPSCL(10) = NPSCL(10) + 1                                         MORS 870
C     CALL BANKR(10)  FOR TIME-KILL ANALYSIS                            MORS 840
      GO TO 165                                                         MORS 890
 110  IF(WATE)120,115,120                                               MORS 900
 115  NDEAD(2)=NDEAD(2)+1                                               MORS 910
      DEADWT(2)=DEADWT(2)+WTBC                                          MORS 920
C  ESCAPE                                                               MORS 890
      GO TO 165                                                         MORS 940
```

```
 120  IF(NALB)130,130,125                                                   MORS 950
 125  ISCT = LOCNSC + 2*NGPREG + (NREG-1)*NMTG + IG                          MORS 960
      NTS(ISCT) = NTS(ISCT) + 1                                             MORS 970
      ISCT = ISCT + NGPREG                                                  MORS 980
      WTS(ISCT) = WTS(ISCT) + WATE                                          MORS 990
      CALL ALBDO(IG,U,V,W,WATE,NMED,NREG)                                   MOR 1000
      NPSCL(6) = NPSCL(6) + 1                                               MOR 1010
      CALL BANKR(6)                                                         MOR 1020
      GO TO 65                                                              MOR 1030
 130  CALL GTMED(NMED,IMED)                                                 MOR 1040
      IF (MFISTP) 140,140,135                                               MOR 1050
 135  IF (WTS(LOCFSN+(IMED-1)*NMTG+IG)) 140,140,136                         MORS
 136  CALL FPROB                                                            MORS
 140  IF (NCOMB) 155,155,145                                                MOR 1070
 145  IF(WTS(LOCFSN+(2*MEDIA+IMED-1)*NMTG+IG))155,155,150                   MOR 1080
 150  CALL GPROB                                                            MOR 1090
C                                                                           MOR 1020
 155  ISCT = LOCNSC + (NREG-1)*NMTG + IG                                    MOR 1110
      NTS(ISCT) = NTS(ISCT) + 1                                             MOR 1120
      ISCT = ISCT + NGPREG                                                  MOR 1130
      WTS(ISCT) = WTS(ISCT) + WATE                                          MOR 1140
      ISCT = LOCNSC + 8*NGPREG + IMED                                       MOR 1150
      NTS(ISCT) = NTS(ISCT) + 1                                             MOR 1160
      CALL COLISN( IG,U,V,W,WATE,NMED,NREG)                                 MOR 1170
C                                                                           MOR 1060
      NPSCL(5) = NPSCL(5) + 1                                               MOR 1190
      CALL BANKR(5)                                                         MOR 1200
C       CALLS RELCOL                                                        MOR 1090
      GO TO 65                                                              MOR 1220
 160  NDEAD(3)=NDEAD(3)+1                                                   MOR 1230
      DEADWT(3)=DEADWT(3)+WATE                                              MOR 1240
      NPSCL(9) = NPSCL(9) + 1                                               MOR 1250
C     CALL BANKR(9)  FOR E-CUT ANALYSIS                                     MOR 1140
C  ENERGY CUTOFF                                                            MOR 1150
C**** WILSON   APRIL 18,1989 ****                                           MOR 1150
 165  IF (NS.GT.0) GO TO 61                                                 MOR 1280
      CALL NPART                                                            MOR 1280
      CALL STPART                                                           MOR 1280

      IF (NMEM) 170,170,60                                                  MOR 1280
C**** WILSON   APRIL 18,1989 ****                                           MOR 1150
C  END OF HISTORY                                                           MOR 1170
 170  CALL BANKR(-3)                                                        MOR 1300
C       CALLS NBATCH                                                        MOR 1190
C**** WILSON   APRIL 18,1989 ****                                           MOR 1150
      IF (ITERS) 195,195,55                                                 MORS 520
 55   CALL OUTPT(1)                                                         MORS 530
      CALL NRNPRI(ITIMEI,NBATCH)                                            MOR 1410
C**** WILSON   APRIL 18,1989 ****                                           MOR 1150
      CALL OUTPT(2)                                                         MOR 1320
      IF(ICLOCK(0)-MXT-MAXTIM) 181,181,175                                  MOR 1330
 181  CALL IOLEFT(NIO)                                                      MOR
```

```
C    NIO IS NO. IO'S LEFT AFTER CURRENT BATCH COMPLETED                MOR
      NIOB=NIOB-NIO                                                    MOR
C   NIOB IS  NO. IO'S  IN BATCH JUST COMPLETED                         MOR
      IF(NIOB.GT.NTIO) NTIO=NIOB                                       MOR
      IF(NIO.GT.NTIO) GO TO 180                                        MOR
      ISIG=2                                                           MOR
  175 NITS = NITS - ITERS + 1                                          MOR 1340
      ITERS = 0                                                        MOR 1350
      NQUIT = NQUIT - IRUNS                                            MOR 1360
      IRUNS = -NQUIT                                                   MOR 1370
      IF(ISIG.EQ.1) WRITE(IO,1020) IRUNS,INITS,NITS
 1020 FORMAT(1H0/39H0RUN TERMINATED BY EXECUTION TIME LIMIT            MOR 1390
     1                     /I8,8H RUNS OF,I3,8H BATCHES,16H   AND 1  RUN MOR 1391
     2OF,I3,19H BATCHES COMPLETED./)                                   MOR 1392
      IF(ISIG.EQ.2) WRITE(IO,1050) IRUNS,INITS,NITS                    MOR
 1050 FORMAT(1H0/39H0RUN TERMINATED BY LIMIT ON IO                     MOR
     1                     /I8,8H RUNS OF,I3,8H BATCHES,16H   AND 1  RUN MOR
     2OF,I3,19H BATCHES COMPLETED./)                                   MOR
  180 ITERS = ITERS - 1                                                MOR 1400
      IF(ITERS)195,195,185                                             MOR 1410
  185 IF(MSOUR)40,40,190                                               MOR 1420
  190 ITSTR=1                                                          MOR 1430
C  END OF BATCH                                                        MOR 1320
      GO TO 40                                                         MOR 1450
  195 CALL BANKR(-4)                                                   MOR 1460
C       CALLS NRUN                                                     MOR 1350
      NQUIT=NQUIT-1                                                    MOR 1480
      INDX = -1                                                        MOR 1490
      CALL TIMER(INDX,XTRA)                                            MOR 1500
      WRITE (IO,1030) NITS,(XTRA(I),I=1,INDX)                          MOR 1510
 1030 FORMAT (32H0TIME REQUIRED FOR THE PRECEDING,I4,13H BATCHES WAS ,10MOR 1520
     1A4)                                                              MOR 1521
      CALL TIMER(-2,XTRA)                                              MOR 1530
C  END OF NITS BATCHES                                                 MOR 1370
      IF (NQUIT) 200,200,25                                            MOR 1550
  200 CALL OUTPT(3)                                                    MOR 1560
C  END OF RUN                                                          MOR 1400
      FTIME = ICLOCK(0) - MXT                                          MOR 1580
      FTIME=FTIME/6000.                                                MOR 1590
      WRITE(IO,1040) FTIME                                             MOR 1600
 1040 FORMAT (37H0TOTAL CPU TIME FOR THIS PROBLEM WAS ,F6.2,9H MINUTES.)MOR 1610
      GO TO 10                                                         MOR 1620
      END                                                              MOR 1630

      SUBROUTINE MSOUR                                                 MS000100
C                                                                      * * * *
C     THIS IS THE EXECUTIVE ROUTINE FOR THE GENERATION AND STORAGE OF  MSOUR  *
C      SOURCE PARAMETERS AT THE START OF EACH BATCH      *             MSOUR  *

C *** THIS VERSION OF MSOUR IS DESIGNED FOR THE COMBGEOM PKG WITH MARS MSOUR  *
      REAL*8 XDUM,DIST,UDUM
      COMMON /FISBNK/ MFISTP,NFISBN,NFISH,FTOTL,FWATE,WATEF            MS000200
```

```
       COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDMS000300
      1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGEMS000400
       COMMON /APOLLO/ AGSTRT,DDF,DEADWT(5),ETA,ETATH,ETAUSD,UINP,VINP,  MS000500
      1 WINP,WTSTRT,XSTRT,YSTRT,ZSTRT,TCUT,XTRA(10),                     MS000600
      2   IO,I1,MEDIA,IADJM,ISBIAS,ISOUR,ITERS,ITIME,ITSTR,LOCWTS,LOCFWL,MS000700
      3 LOCEPR,LOCESC,LOCFSN,MAXGP,MAXTIM,MEDALB,NGPREG,MXREG,NALB,      MS000800
      4 NDEAD(5),NEWNM,NGEOM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,    MS000900
      5 NKCALC,NKILL,NLAST,NMEM,NMGP,NMOST,NNTG,NOLEAK,NORMF,NPAST,      MS001000
      6 NPSCL(13),NQUIT,NSIGL,NSOUR,NSPLT,NSTRT,NXTRA(10)                MS001100
       COMMON/GOMLOC/ KMA,KFPD,KLCR,KNSD,KIOR,KRIZ,KRCZ,KMIZ,KMCZ,       MS001300
      1  KKR1,KKR2,KNSR,KVOL,NADD,LDATA,LTMA,LFPD,NUMR,IRTRU,NUMB,NIR    MS001400
      2  ,KBIZ,KBCZ                                                      MS00
       COMMON /ORGI/ DIST ,MARK,NMEDG,NBLZ                               MS00
       COMMON/ARAB/NBY,NLEV,NAR,NQ,IAW,IAY,NF,NX1(3)                     MS00
       COMMON/REPEAT/ JP(20)                                             MS00
       COMMON /MGDNV/  MUS,MUZ,LL,IPRET,IFLOW,IECT,NLO,IGX               MS00
       COMMON /OUTB/ SWATE,XAVE,YAVE,ZAVE,EAVE,UAVE,VAVE,WAVE,AGEAVE     GSTOR 40
       COMMON NSTOR(1)                                                   MS001500
       DIMENSION XDUM(3),UDUM(3)                                         MS00
       INTEGER BLZNT                                                     MS001600
       UOLD=0.                                                           MS001700
       VOLD=0.                                                           MS001800
       WOLD=0.                                                           MS001900
       ETATH=0.                                                          MS002000
       XOLD=0.                                                           MS002100
       YOLD=0.                                                           MS002200
       ZOLD=0.                                                           MS002300
       BLZON=0.                                                          MS002400
       OLDWT=WTSTRT                                                      MS002500
       ETA=0.                                                            MS002600
       IGO=0                                                             MS002700
       MEDOLD=0.                                                         MS002800
       OLDAGE=0.                                                         MS002900
       WATE=WTSTRT                                                       MS003700
       X=XSTRT                                                           MS003800
       Y=YSTRT                                                           MS003900
       Z=ZSTRT                                                           MS004000
       AGE=AGSTRT                                                        MS004100
       NAMEX=1                                                           MS004200
       CALL SOURCE(IG,UINP,VINP,WINP,X,Y,Z,WATE,NMED,AGE,ISOUR,ITSTR,NGP MS004300
      1QT3,DDF,ISBIAS,NNTG)                                              MS004400
C *** DEFINE SOURCE ANGLE BEFORE CALLING CALI - LOOKZ     **             MSOUR **
       IF(ABS(UINP)+ABS(VINP)+ABS(WINP)) 35,35,30                        MS005400
 30    U1=UINP                                                           MS005500
       V1=VINP                                                           MS005600
       W1=WINP                                                           MS005700
       GO TO 40                                                          MS005800
 35    CALL GTISO(U1,V1,W1)                                              MS005900
C       SELECT ISOTROPIC DIRECTION COSINES                *     *        MSOUR  *
 40    U=U1                                                              MS006000
       V=V1                                                              MS006100
```

```
      W=W1                                                              MS006200
      UDUM(1)=U                                                         MS004500
      UDUM(2)=V                                                         MS004600

      UDUM(3)=W                                                         MS004700
      XDUM(1)=X                                                         MS004500
      XDUM(2)=Y                                                         MS004600
      XDUM(3)=Z                                                         MS004700
      NIX = 6*NLEV+4                                                    MS00
      IF(NLEV.LE.0) GO TO 110                                           MS00
      DO 100 I=1,NIX                                                    MS00
      IX1 = JP(12)+I-1                                                  MS00
  100 NSTOR(IX1) = 0                                                    MS00
  110 CONTINUE                                                          MS00
      JP4 = JP(4)                                                       MS00
      JP5 = JP(5)                                                       MS00
      JP7 = JP(7)                                                       MS00
      JP8 = JP(8)                                                       MS00
      JP0 = JP(10)                                                      MS00
      LL = 0                                                            MS00
      JL =0                                                             MS00
      JLU = 0                                                           MS00
C    SUBROUTINE CALI WILL CALL LOOKZ                                    MS00
C     THIS CALL TO LOOKZ CORRESPONDS TO THE COMBGEOM VERSION OF LOOKZ   MSOUR  *
      CALL CALI(JL,JLU,XDUM,UDUM,JP,NSTOR(JP4),NSTOR(JP5),NSTOR(JP7),   MS004800
     1 NSTOR(JP8),NSTOR(JP0),NSTOR(KFPD),NSTOR(KNA ),NSTOR(KLCR),       MS004800
     . NSTOR,NSTOR)                                                     MS004900
      BLZNT=NBLZ                                                        MS00
      NREG=NSTOR(KRIZ+NMEDG-1)                                          MS005100
      NMED=NSTOR(KMIZ+NMEDG-1)                                          MS005200
      NAME=1                                                            MS005300
      IF(ISOUR)50,50,45                                                 MS006300
   45 IG=ISOUR                                                          MS006400
   50 CALL STORNT(1,0)                                                  MS006500
C     PLACE SOURCE PARAMETERS IN PARTICLE BANK    *            *        MSOUR  *
      NPSCL(1) = NPSCL(1) + 1                                           MS006600
C ** PARTICLE POSITION MUST BE IN LOCAL COORDINATES WHEN SDATA CALLED * MS00
C **  GETNT SELECTS X,Y,Z FROM BANK AND CONVERTS TO LOCAL COORD         MS00
      CALL GETNT(1,1)                                                   MS00
      SWATE=SWATE+WATE                                                  OUTP 330
      XAVE=XAVE+WATE*X                                                  OUTP 340
      YAVE=YAVE+WATE*Y                                                  OUTP 350
      ZAVE=ZAVE+WATE*Z                                                  OUTP 360
      EAVE=EAVE+WATE*IG                                                 OUTP 370
      UAVE=UAVE+WATE*U                                                  OUTP 380
      VAVE=VAVE+WATE*V                                                  OUTP 390
      WAVE=WAVE+WATE*W                                                  OUTP 400
      AGEAVE=AGEAVE+WATE*AGE                                            OUTP 410
      CALL BANKR(1)                                                     MS006700
      NEWNM=1                                                           MS006900
      RETURN                                                            MS007300
      END                                                               MS007400
```

```
      SUBROUTINE NPART                                                  NBATC 10
      REAL * 8 F, BC
C     THIS ROUTINE IS CALLED AT THE END OF EACH HISTORY TO DO THE SUMS  NBATC  *
C         NEEDED FOR ESTIMATED MEANS AND FOR CALC. OF BATCH STATISTICS  NBATC  *
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,NBATC 20
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   NBATC 21
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                NBATC 22
C     COMMON BC(1)                                                      NBATC 30
      COMMON /FRE/ W(189),NFREQ(10,188)
      COMMON /SSS/ BC(1)                                                NBATC 30
      DIMENSION IW(9),JW(21)
      DATA IW/1,2,3,4,5,6,7,8,9/,JW/-19,-18,-17,-16,-15,-14,-13,
     *-12,-11,-10,-9,-8,-7,-6,-5,-4,-3,-2,-1,0,1/
      DATA IFLA/0/

      IF(IFLA.GT.0) GO TO 301
      IFLA=1
      K=0
      DO 303 J=1,21
      DO 302 I=1,9
      K=K+1
      W(K)=IW(I)*10.**JW(J)
 302  CONTINUE
 303  CONTINUE
 301  DO 70 I=1,ND                                                      NBATC 40
      IU = LOCUD+(I-1)*NRESP                                            NBATC 50
      IS = LOCSD + (I-1)*NRESP                                          NBATC 60
      IF = LOCQE + (I-1)*NE                                             NBATC 70
      IT = LOCQT + (I-1)*NTNR                                           NBATC 80
      IE = LOCQTE + (I-1)*NTNE                                          NBATC 90
      IA = LOCQAE + (I-1)*NANE                                          NBAT 100
      IF (NE) 15,15,5                                                   NBAT 110
 5    DO 10 J=1,NE                                                      NBAT 120
      ISUB = IF + J                                                     NBAT 130
      F = BC(ISUB)                                                      NBAT 140
      ISUB = ISUB + NEND                                                NBAT 150
      BC(ISUB) = BC(ISUB) + F                                           NBAT 160
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 170
      ISUB = ISUB + NEND                                                NBAT 180
 10   BC(ISUB) = BC(ISUB) + F*F                                         NBAT 190
 15   DO 30 M=1,NRESP                                                   NBAT 200
C ***********THIS PART FOR THE UNCOLL*************                      NBAT 300
      ISUB = IU + M                                                     NBAT 280
      F = BC(ISUB)                                                      NBAT 290
      ISUB = ISUB + NDNR                                                NBAT 300
      BC(ISUB) = BC(ISUB) + F                                           NBAT 310
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 320
      ISUB = ISUB + NDNR                                                NBAT 330
      BC(ISUB) = BC(ISUB) + F*F                                         NBAT 340
C **********THIS PART FOR THE BATCH *************                       NBAT 300
      ISUB = IS + M                                                     NBAT 260
      ISUB = ISUB + 3*NDNR                                              NBAT 300
```

```
      F = BC(ISUB)                                                      NBAT 290
      ISUB = ISUB + NDNR                                                NBAT 300
      BC(ISUB) = BC(ISUB) + F                                           NBAT 310
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 320
      ISUB = ISUB + NDNR                                                NBAT 330
      BC(ISUB) = BC(ISUB) + F*F                                         NBAT 340
C **********THIS PART FOR THE ACCUMULATIVE *****                        NBAT 300
      ISUB = ISUB + NDNR                                                NBAT 300
      BC(ISUB) = BC(ISUB) + F                                           NBAT 310
C     WRITE(50,650) I,F
C650  FORMAT(1H ,I2,3X,1PE11.2)
C****************FOR FREQUENCY CALCULATIONS*********

      IF(F.GT.0.) GO TO 300
      NFREQ(I,1)=NFREQ(I,1)+1
      GO TO 310
 300  IF(F.LT.90.) GO TO 322
      NFREQ(I,188)=NFREQ(I,188)+1
      GO TO 310
 322  DO 323 L=2,187
      TEST=W(L)-F
      IF(TEST.GT.0.) GO TO 333
 323  CONTINUE
 333  NFREQ(I,L)=NFREQ(I,L)+1
C****************FOR FREQUENCY CALCULATIONS*********
 310  IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 320
      ISUB = ISUB + NDNR                                                NBAT 330
      BC(ISUB) = BC(ISUB) + F*F                                         NBAT 340
      ISUB = IS + M                                                     NBAT 280
      ISUB = ISUB + 10*NDNR                                             NBAT 330
      BC(ISUB) = BC(ISUB) + F*F*F                                       NBAT 340
      ISUB = ISUB + NDNR                                                NBAT 330
      BC(ISUB) = BC(ISUB) + F*F*F*F                                     NBAT 340
      IF (NT) 30,30,20                                                  NBAT 350
 20   DO 25 K=1,NT                                                      NBAT 360
      ISUB = IT + (K-1)*NRESP + M                                       NBAT 370
      F = BC(ISUB)                                                      NBAT 380
      ISUB = ISUB + NTNDNR                                              NBAT 390
      BC(ISUB) = BC(ISUB) + F                                           NBAT 400
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 410
      ISUB = ISUB + NTNDNR                                              NBAT 420
 25   BC(ISUB) = BC(ISUB) + F*F                                         NBAT 430
 30   CONTINUE                                                          NBAT 440
      IF (NE) 70,70,35                                                  NBAT 450
 35   DO 65 J=1,NE                                                      NBAT 460
      ITE = IE + (J-1)*NT                                               NBAT 470
      IAE = IA + (J-1)*NA                                               NBAT 480
      IF (NT) 50,50,40                                                  NBAT 490
 40   DO 45 K=1,NT                                                      NBAT 500
      ISUB = ITE + K                                                    NBAT 510
      F = BC(ISUB)                                                      NBAT 520
      ISUB = ISUB + NTNEND                                              NBAT 530
```

```
      BC(ISUB) = BC(ISUB) + F                                           NBAT 540
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 550
      ISUB = ISUB + NTHEND                                              NBAT 560
   45 BC(ISUB) = BC(ISUB) + F*F                                         NBAT 570
   50 IF (NA) 65,65,55                                                  NBAT 580
   55 DO 60 L=1,NA                                                      NBAT 590
      ISUB = IAE + L                                                    NBAT 600
      F = BC(ISUB)                                                      NBAT 610
      ISUB = ISUB + NANEND                                              NBAT 620
      BC(ISUB) = BC(ISUB) + F                                           NBAT 630
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 640
      ISUB = ISUB + NANEND                                              NBAT 650
   60 BC(ISUB) = BC(ISUB) + F*F                                         NBAT 660
   65 CONTINUE                                                          NBAT 670
   70 CONTINUE                                                          NBAT 680
      RETURN                                                            NBAT 690
      END                                                               NBAT 700

      SUBROUTINE NBATCH(NSORC)                                          NBATC 10
C     THIS ROUTINE IS CALLED AT THE END OF EACH BATCH TO DO THE SUMS    NBATC  *
C         NEEDED FOR ESTIMATED MEANS AND FOR CALC. OF BATCH STATISTICS  NBATC  *
      REAL * 8 F ,BC
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,NBATC 20
     1 NANE,NTHDNR,NTHEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   NBATC 21
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                NBATC 22
      COMMON /SSS/ BC(1)                                                NBATC 30
      DO 70 I=1,ND                                                      NBATC 40
      IS = LOCSD + (I-1)*NRESP                                          NBATC 60
      DO 30 K=1,NRESP                                                   NBAT 200
      ISUB = IS + K                                                     NBAT 280
      F = BC(ISUB)                                                      NBAT 290
      ISUB = ISUB + NDNR                                                NBAT 300
      BC(ISUB) = BC(ISUB) + F                                           NBAT 310
      IF (DABS(F) .LT. 1.0E-35) F = 0.0                                 NBAT 320

      ISUB = ISUB + NDNR                                                NBAT 330
      BC(ISUB) = BC(ISUB) + F*F/NSORC                                   NBAT 340
   30 CONTINUE                                                          NBAT 440
   70 CONTINUE                                                          NBAT 680
      RETURN                                                            NBAT 690
      END                                                               NBAT 700

      SUBROUTINE NRNPRI(ITIMEI,NBATCH)                                  NRUN  10
C     THIS ROUTINE IS CALLED AT THE END OF EACH BATCH                   NRUN * *
C      IT NORMALIZES AND OUTPUTS CALCULATED QUANTITIES ALONG WITH       NRUN * *
C             FRACTIONAL STANDARD DEVIATIONS                            NRUN * *
      REAL * 8 FN1,FN2,FSAVE1,FSAVE2,E,ANUMB,FOM,FTIMEO
      COMMON /USER/ AGSTRT,WTSTRT,XSTRT,YSTRT,ZSTRT,DFF,EBOTN,EBOTG,    NRUN  20
     1 TCUT,I0,I1,IADJN,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,NLAST,  NRUN  21
     2 NLEFT,NMGP,NMTG,NSTRT                                            NRUN  22
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,NRUN  30
     1 NANE,NTHDNR,NTHEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   NRUN  31
```

```
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                NRUN  32
      COMMON /SSS/ E(1)
      COMMON /FRE/ W(189),NFREQ(10,188)
      COMMON A(1)                                                       NRUN  40
      DIMENSION NUMB(1),IHOLL(10),ANUMB(1),P(10),PP(10),NSUM(20,10)     NRUN  50
      EQUIVALENCE (A(1),NUMB(1)), (IHOLL(1),STA)                        NRUN  60
      EQUIVALENCE (E(1),ANUMB(1))                                       NRUN  60
      DATA IH1/1H1/, IH2/1H2/, IH0/1H0/                                 NRUN  70
      DATA P/0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.999/                 NRUN  70
      DATA PP/10.,20.,30.,40.,50.,60.,70.,80.,90.,100./                 NRUN  70
      NPART1= NSTRT                                                     NRUN  80
      IFL  = 1
      IF(NITS.EQ.NBATCH) IFL  = -1
      NPART2= NBATCH*NSTRT                                              NRUN  80
C     NPART = NBATS*NSTRT                                               NRUN  80
      DO 350 K=1,188
      WRITE(60,670)W(K),(NFREQ(I,K),I=1,ND)
 670  FORMAT(1H ,1PE9.2,10(1X,I6))
 350  CONTINUE
      FN1= 1.0/NPART1                                                   NRUN  90
      FN2= 1.0/NPART2                                                   NRUN  90
      IF (IADJM) 10,10,5                                                NRUN 100
 5    FN1= FN1*DFF                                                      NRUN 110
      FN2= FN2*DFF                                                      NRUN 110
 10   IX = LOCSD + 4*NDNR + 1                                           NRUN 170
      IX2 = IX + NDNR                                                   NRUN 180
      CALL VAR2(E(IX),E(IX2),NRESP,ND,1,NSTRT)                          NRUN 190
      IX1 = LOCSD + 6*NDNR                                              NRUN 190
      IX3 = IX1 + NDNR                                                  NRUN 190
      IX = IX3 + NDNR                                                   NRUN 170
      IX2 = IX + NDNR                                                   NRUN 180
      DO 300 I=1,NDNR
      E(IX+I)  = E(IX1+I)
      E(IX2+I) = E(IX3+I)
 300  CONTINUE
      IX  = LOCSD + 8*NDNR + 1                                          NRUN 190
      IX2 = IX  + NDNR                                                  NRUN 190
      IX3 = IX2 + NDNR                                                  NRUN 190
      IX4 = IX3 + NDNR                                                  NRUN 190
      IX5 = IX4 + NDNR                                                  NRUN 190
      IX6 = IX5 + NDNR                                                  NRUN 190
      CALL VAR4(E(IX),E(IX2),E(IX3),E(IX4),E(IX5),E(IX6),               NRUN 190
     *NRESP,ND,1,NPART2,IFL)
C  *  *  THE STATEMENTS IN THE FOLLOWING LOOP NORMALIZE ALL ARRAYS TO BE OUTPUT

      FSAVE1= FN1                                                       NRUN 380
      FSAVE2= FN2                                                       NRUN 380
      DO 120 I=1,ND                                                     NRUN 390
      FN1= FSAVE1*A(LOCXD+5*ND+I)                                       NRUN 400
      FN2= FSAVE2*A(LOCXD+5*ND+I)                                       NRUN 400
      IA = LOCSD + (I-1)*NRESP + 4*NDNR                                 NRUN 420
      IB = LOCSD + (I-1)*NRESP + 8*NDNR                                 NRUN 420
```

```
      DO 76 M=1,NRESP                                                   NRUN 470
      E(IA+M) = E(IA+M)*FN1                                             NRUN 490
      E(IB+M) = E(IB+M)*FN2                                             NRUN 490
   76 CONTINUE                                                          NRUN 490
  120 CONTINUE                                                          NRUN 490
      IHP = IB0                                                         NRUN 890
C  *  *  OUTPUT SBD, SBD2, SAD AND SAD2 ARRAYS IN THE FOLLOWING LOOP    NRUN 870
      DO 125 M=1,NRESP                                                  NRUN 910
      IHOL = NLAST + (M-1)*20 + 1                                       NRUN 920
      IHOL2 = IHOL + 19                                                 NRUN 930
      WRITE (IO,1020) IHP,(A(I),I=IHOL,IHOL2)                           NRUN 940
 1020 FORMAT (A1,20A4)                                                  NRUN 950
      IHP = IB2                                                         NRUN 960
      IST = LOCQE -  9                                                  NRUN 970
      WRITE (IO,1030) (ANUMB(I),I=IST,LOCQE)                            NRUN 980
 1030 FORMAT (1H0,32X,20HRESPONSES(DETECTOR) ,10A8,/,5X,8HDETECTOR,9X,  NRUN 990
     1 6HBATCH ,12X,3HFSD,10X,11HACCUMULATED,10X,3HFSD,11X,6HCV(SD),    NRUN 991
     2 11X,3HFOM,/,
     3 21X,8HRESPONSE,26X,8HRESPONSE)                                   NRUN 992
C1030 FORMAT (1H0,32X,20HRESPONSES(DETECTOR) ,10A8,/.5X,8HDETECTOR,7X,  NRUN 990
C    1 6HBATCH ,10X,3HFSD, 8X,11HACCUMULATED, 6X,3HFSD. 9X,7HCV(SDI),   NRUN 991
C    2  8X,6HCV(SD),9X,3HFOM,/,
C    3 19X,8HRESPONSE,24X,8HRESPONSE)                                   NRUN 992
      WRITE(40,222) NPART2
  222 FORMAT (1H ,/,14X,' NUMBER OF PARTICLES ',I8,/)
      WRITE(30,232)        NPART2
  232 FORMAT (1H ,/,20X,I8,/)
      DO 125 I=1,ND                                                     NRU 1000
      IA = LOCSD + 4*NDNR + (I-1)*NRESP + M                             NRU 1030
      IB = IA + NDNR                                                    NRU 1040
      IC = LOCSD + 8*NDNR + (I-1)*NRESP + M                             NRU 1030
      ID = IC + NDNR                                                    NRU 1040
      IL = LOCSD + 12*NDNR + (I-1)*NRESP + M                            NRU 1030
      IF = IL + NDNR                                                    NRU 1040
      ITIMEQ = ICLOCK(0)-ITIMEI                                         NRU 1040
      FTIMEO = ITIMEO/6000.0                                            NRU 1040
      FOM =1./(E(ID)**2*FTIMEO)                                         NRU 1040
C125  WRITE (IO,1040) I,E(IA),E(IB),E(IC),E(ID),E(IL),FOM               NRU 1050
      WRITE (30,1051) I,E(IA),E(IB),E(IC),E(ID),E(IF),FOM               NRU 1050
      WRITE (40,1041) I,E(IA),E(IB),E(IC),E(ID),E(IF),FOM               NRU 1050
 1051 FORMAT (2X,I3,'&',1PE11.4,'&',0PF7.4,'&',                         NRU 1060
     *1PE11.4,'&',0PF7.4,'&',0PF7.4,'&',                                NRU 1060
     *1PE11.4)
 1041 FORMAT (I3,1PE13.4,0PF10.4,1PE13.4,0PF10.4,0PF10.4,               NRU 1060
     *1PE13.4)
 125  WRITE (IO,1040) I,E(IA),E(IB),E(IC),E(ID),E(IF),FOM               NRU 1050
 1040 FORMAT (I10,1PE20.4,0PF15.5,1PE19.4,0PF15.5,0PF15.5,1PE18.4)      NRU 1060
C1040 FORMAT (I10,1PE18.4,0PF13.5,1PE17.4,0PF13.5,0PF13.5,              NRU 1060
C    *0PF13.5,1PE16.4)                                                  NRU 1060
      DO 414 I=1,ND
      NPA=0
```

```
      SUM=0.0
      TSUM=0.0

      DO 408 K=2,188
      DEL=(W(K)+W(K+1))/2.
      TSUM=TSUM+NFREQ(I,K)*DEL
  408 CONTINUE
      J=1
      DO 412 K=2,188
      NPA=NPA+NFREQ(I,K)
      DEL=(W(K)+W(K+1))/2.
      SUM=SUM+DEL*NFREQ(I,K)
      TEST=SUM/TSUN
      IF(TEST.LT.P(J)) GO TO 412
  406 NSUM(I,J)=NPA
      J=J+1
      NPA=0
      IF(J.GT.10) GO TO 414
      IF(TEST.GT.P(J)) GO TO 406
  412 CONTINUE
  414 CONTINUE
      WRITE(IO,634)
  634 FORMAT(1H ,//,5X,'DETECTOR',10X,' PARTICLE DISTRIBUTION IN',
     *' PERCENTAGE BINS RELATIVE TO INCREASING VALUE OF',
     *' CONTRIBUTIONS',/)
      PR=0.0
      WRITE(IO,636)PR,(PP(J),J=1,10)
  636 FORMAT(1H ,10X,11(5X,F5.1)/)
      DO 422 I=1,ND
      WRITE(IO,644) I,(NSUM(I,J),J=1,10)
  644 FORMAT(1H ,I9,4X,10(3X,I7))
  422 CONTINUE
      RETURN                                                            NRU 3170
      END                                                               NRU 3180

      SUBROUTINE NRUN(NBATS,NRUNS)                                      NRUN  10
C     THIS ROUTINE IS CALLED AT THE END OF EACH RUN                     NRUN * *
C      IT NORMALIZES AND OUTPUTS CALCULATED QUANTITIES ALONG WITH       NRUN * *
C            FRACTIONAL STANDARD DEVIATIONS                             NRUN * *
      REAL * 8 FM, FSAVE,E,ANUMB
      COMMON /USER/ AGSTRT,WTSTRT,XSTRT,YSTRT,ZSTRT,DFF,EBOTN,EBOTG,    NRUN  20
     1 TCUT,IO,I1,IADJM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,NLAST,  NRUN  21
     2 NLEFT,NNGP,NMTG,NSTRT                                            NRUN  22
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,NRUN  30
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   NRUN  31
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                NRUN  32
      COMMON /SSS/ E(1)
      COMMON A(1)                                                       NRUN  40
      DIMENSION NUMB(1),ANUMB(1),IHOLL(10)                              NRUN  50
      EQUIVALENCE (A(1),NUMB(1)), (IHOLL(1),STR)                        NRUN  60
      EQUIVALENCE (E(1),ANUMB(1))                                       NRUN  60
      DATA IH1/1H1/, IH2/1H2/, IHO/1HO/                                 NRUN  70
```

```
      NPART = NBATS*NSTRT                                               NRUN  80
      FN = 1.0/NPART                                                    NRUN  90
      IF (IADJM) 10,10,5                                                NRUN 100
    5 FN = FN*DFF                                                       NRUN 110
      WRITE (IO,1000)                                                   NRUN 120
 1000 FORMAT (82H2THIS IS AN ADJOINT PROBLEM - ADJOINT ENERGY DEPENDENT NRUN 130
     1FLUENCE IS NOT DIFFERENTIAL)                                      NRUN 131
   10 IX = LOCUD + NDNR + 1                                             NRUN 140
      IX2 = IX + NDNR                                                   NRUN 150
      CALL VAR2(E(IX),E(IX2),NRESP,ND,1,NPART)                          NRUN 160
C*********WILSON SEPT/26/89*******

      IX = LOCSD + 2*NDNR + 1
      IZ = LOCSD + 6*NDNR + 1
      IY  = IZ + NDNR                                                   NRUN 150
      CALL FTEST(E(IX),E(IY),E(IZ),NRESP,ND,NBATS,NPART)                NRUN 190
C*********WILSON SEPT/26/89*******
      IX = LOCSD + NDNR + 1                                             NRUN 170
      IX2 = IX + NDNR                                                   NRUN 180
      CALL VAR2(E(IX),E(IX2),NRESP,ND,NBATS,NPART)                      NRUN 190
      IF (NE) 20,20,15                                                  NRUN 200
   15 IX = LOCQE + NEND + 1                                             NRUN 210
      IX2 = IX + NEND                                                   NRUN 220
      CALL VAR2(E(IX),E(IX2),NE,ND,1,NPART)                             NRUN 230
   20 IF (NT) 35,35,25                                                  NRUN 240
   25 IX = LOCQT + NTNDNR + 1                                           NRUN 250
      IX2 = IX + NTNDNR                                                 NRUN 260
      CALL VAR3(E(IX),E(IX2),NRESP,NT,ND,1,NPART)                       NRUN 270
      IF (NE) 45,45,30                                                  NRUN 280
   30 IX = LOCQTE + NTNEND + 1                                          NRUN 290
      IX2 = IX + NTNEND                                                 NRUN 300
      CALL VAR3(E(IX),E(IX2),NT,NE,ND,1,NPART)                          NRUN 310
   35 IF (NA) 45,45,40                                                  NRUN 320
   40 IX = LOCQAE + NANEND + 1                                          NRUN 330
      IX2 = IX + NANEND                                                 NRUN 340
      CALL VAR3(E(IX),E(IX2),NA,NE,ND,1,NPART)                          NRUN 350
   45 IF = LOCIB + 2*NE                                                 NRUN 360
C  *  * THE STATEMENTS IN THE FOLLOWING LOOP NORMALIZE ALL ARRAYS TO BE OUTPUT
      FSAVE = FN                                                        NRUN 380
      DO 120 I=1,ND                                                     NRUN 390
      FN = FSAVE*A(LOCXD+5*ND+I)                                        NRUN 400
      IA = LOCUD + (I-1)*NRESP+NDNR                                     NRUN 410
      IB = LOCSD + (I-1)*NRESP + 6*NDNR                                 NRUN 420
      IC = LOCQE + (I-1)*NE + NEND                                      NRUN 430
      IF (NE) 60,60,50                                                  NRUN 440
   50 DO 55 J=1,NE                                                      NRUN 450
   55 E(IC+J) = E(IC+J)/A(IF+J)*FN                                      NRUN 460
   60 DO 75 M=1,NRESP                                                   NRUN 470
      E(IA+M) = E(IA+M)*FN                                              NRUN 480
      E(IB+M) = E(IB+M)*FN                                              NRUN 490
      IF (NT) 75,75,65                                                  NRUN 500
   65 IE = LOCQT + (I-1)*NTNR + NTNDNR                                  NRUN 510
```

```
      IDT = LOCT + ND*NT + (I-1)*NT
      DO 70 K=1,NT
      IQE = IE + (K-1)*NRESP + M
      IDT = IDT + 1
   70 E(IQE) = E(IQE)*FN/A(IDT)
   75 CONTINUE
      IF (NE) 120,120,80
   80 IQTE = LOCQTE + (I-1)*NTNE + NTNEND
      IQAE = LOCQAE + (I-1)*NANE + NANEND
      DO 115 J=1,NE
      IF (NT) 95,95,85
   85 IQT = IQTE + (J-1)*NT
      IDT = LOCT + ND*NT + (I-1)*NT
      DO 90 K=1,NT
      IQ = IQT + K
      IDT = IDT + 1
   90 E(IQ) = E(IQ)/A(IDT)/A(IF+J)*FN
   95 IF (NA) 115,115,100
  100 IQA = IQAE + (J-1)*NA
C    THE DA ARRAY IS BEING STORED IN THE FIRST NA CELLS IN QAE
      IDA = LOCQAE + 1

      IC = LOCCO + 1
      E(IDA) = (A(IC) + 1.0)*6.2832
      DO 105 L=2,NA
      IDA = IDA + 1
      IC = IC + 1
  105 E(IDA) = (A(IC) - A(IC-1))*6.2832
      IDA = LOCQAE
      DO 110 L=1,NA
      IQ = IQA + L
      IDA = IDA + 1
  110 E(IQ) = E(IQ)/A(IDA)/A(IF+J)*FN
  115 CONTINUE
  120 CONTINUE
      CALL DATE (IHOLL,I)
      WRITE (IO,1010) (IHOLL(J),J=1,I)
 1010 FORMAT (22H1THIS CASE WAS RUN ON ,10A4)
      IHP = IHO
C  *  *  OUTPUT SUD, SUD2, SSD AND SSD2 ARRAYS IN THE FOLLOWING LOOP
      DO 125 M=1,NRESP
      IHOL = NLAST + (M-1)*20 + 1
      IHOL2 = IHOL + 19
      WRITE (IO,1020) IHP,(A(I),I=IHOL,IHOL2)
 1020 FORMAT (A1,20A4)
      IHP = IH2
      IST = LOCQE -  9
      WRITE (IO,1030) (ANUMB(I),I=IST,LOCQE)
 1030 FORMAT (1H0,32X,20HRESPONSES(DETECTOR) ,10A6,/,5X,8HDETECTOR,9X,
     1 6HUNCOLL,12X,3HFSD,13X,5HTOTAL,13X,3HFSD,12X,3HFSD,10X,
     3 7HF VALUE,/,21X,8HRESPONSE,26X,8HRESPONSE,10X,5HBATCH)
      DO 125 I=1,ND
```

```
      IA = LOCUD + NDNR + (I-1)*NRESP + N                               NRU 1010
      IB = IA + NDNR                                                    NRU 1020
      IC = LOCSD + NDNR + (I-1)*NRESP + N                               NRU 1030
      ID = IC + NDNR                                                    NRU 1040
      IC = LOCSD + 6*NDNR + (I-1)*NRESP + N                             NRU 1030
      IF = IC + NDNR
      IE = LOCSD + 9*NDNR + (I-1)*NRESP + N                             NRU 1030
  125 WRITE (IO,1040) I,E(IA),E(IB),E(IC),E(ID),E(IE),E(IF)             NRU 1050
 1040 FORMAT (I10,1PE20.4,0PF15.5,1PE19.4,0PF15.5,0PF15.5,              NRU 1060
     *0PF15.5)
      IF (NE) 210,210,130                                               NRU 1070
  130 NDM = (ND-1)/10 + 1                                               NRU 1080
      NEM = (NE-1)/17 + 1                                               NRU 1090
      IHP = IE1                                                         NRU 1100
C  *  *  OUTPUT SQE AND SQE2 IN THE FOLLOWING LOOP                      NRU 1180
      DO 205 IND = 1,NDM                                                NRU 1120
      ID1 = (IND-1)*10 + 1                                              NRU 1130
      IF (IND-NDM) 135,140,140                                          NRU 1140
  135 ID2 = ID1 + 9                                                     NRU 1150
      GO TO 145                                                         NRU 1160
  140 ID2 = ND                                                          NRU 1170
  145 DO 205 INE=1,NEM                                                  NRU 1180
      IE1 = (INE-1)*17 + 1                                              NRU 1190
      IF (INE-NEM) 150,155,155                                          NRU 1200
  150 IE2 = IE1 + 16                                                    NRU 1210
      GO TO 160                                                         NRU 1220
  155 IE2 = NE                                                          NRU 1230
  160 IF (INE-1) 165,165,170                                            NRU 1240
  165 ETOP = EFIRST                                                     NRU 1250
      GO TO 175                                                         NRU 1260

  170 ETOP = A(LOCIB+NE+IE1-1)                                          NRU 1270
  175 IST = LOCQT -  9                                                  NRU 1280
      WRITE (IO,1050) IHP,(ANUMB(I),I=IST,LOCQT),(I,I=ID1,ID2)          NRU 1290
 1050 FORMAT (A1,27X,24HFLUENCE(ENERGY,DETECTOR),1X,10A6,/,16H0   DETECTNRU 1300
     1OR NO.,I8,9I10)                                                   NRU 1301
      IF (NE-7) 180,180,185                                             NRU 1310
  180 IHP = IE2                                                         NRU 1320
  185 WRITE (IO,1060) ETOP                                              NRU 1330
 1060 FORMAT (9H ENERGIES,/,1X,1PE11.3)                                 NRU 1340
      DO 205 IE = IE1,IE2                                               NRU 1350
      IF (NNE) 190,200,190                                              NRU 1360
  190 IF (IE-NNE-1) 200,195,200                                         NRU 1370
  195 WRITE (IO,1060) EGTOP                                             NRU 1380
  200 IEP = LOCIB + NE + IE                                             NRU 1390
      ID11 = LOCQE + (ID1-1)*NE + NEND + IE                             NRU 1400
      ID12 = ID11 + (ID2-ID1)*NE                                        NRU 1410
      WRITE (IO,1070)        (E(I),I=ID11,ID12,NE)                      NRU 1420
 1070 FORMAT (17X,1P10E10.3)                                            NRU 1430
      ID11 = ID11 + NEND                                                NRU 1440
      ID12 = ID12 + NEND                                                NRU 1450
      WRITE (IO,1080) (E(I),I=ID11,ID12,NE)                             NRU 1460
```

```
 1080 FORMAT (17X,10(F9.3,1X))                                          NRU 1470
 205  WRITE (IO,1090) A(INP)                                            NRU 1480
 1090 FORMAT (1X,1PE11.3)                                               NRU 1490
 210  IF (NT) 375,375,215                                               NRU 1500
 215  NRM = (NRESP-1)/10 + 1                                            NRU 1510
      NTM = (NT-1)/17 + 1                                               NRU 1520
      INP = IN1                                                         NRU 1530
C  *  *  OUTPUT SQT AND SQT2 IN THE FOLLOWING LOOP                      NRU 1580
      DO 275 I=1,ND                                                     NRU 1550
      DO 275 INR=1,NRM                                                  NRU 1560
      IR1 = (INR-1)*10 + 1                                              NRU 1570
      IF (INR-NRM) 220,225,225                                          NRU 1580
 220  IR2 = IR1 + 9                                                     NRU 1590
      GO TO 230                                                         NRU 1600
 225  IR2 = NRESP                                                       NRU 1610
 230  DO 275 INT=1,NTM                                                  NRU 1620
      IST = LOCQTE -  9                                                 NRU 1630
      WRITE (IO,1100) INP,I,(ANUMB(IPR),IPR=IST,LOCQTE)                 NRU 1640
 1100 FORMAT (A1,11HDETECTOR NO,I3,6X,32HRESPONSE(RESPONSE,TIME,DETECTORNRU 1650
     1),1X,10A8)                                                        NRU 1651
      IF (NT-8) 235,235,240                                             NRU 1660
 235  INP = IN2                                                         NRU 1670
 240  IT1 = (INT-1)*17 + 1                                              NRU 1680
      IF (INT-NTM) 245,250,250                                          NRU 1690
 245  IT2 = IT1 + 16                                                    NRU 1700
      GO TO 255                                                         NRU 1710
 250  IT2 = NT                                                          NRU 1720
 255  IF (INT-1) 260,260,265                                            NRU 1730
 260  AGS = A(LOCXD + 4*ND + I)                                         NRU 1740
      GO TO 270                                                         NRU 1750
 265  ISUB = LOCT + (I-1)*NT + IT1 - 1                                  NRU 1760
      AGS = A(ISUB)                                                     NRU 1770
 270  WRITE (IO,1110) (IR,IR=IR1,IR2)                                   NRU 1780
 1110 FORMAT (4X,9H RESPONSE,10I10)                                     NRU 1790
      WRITE (IO,1120) AGS                                               NRU 1800
 1120 FORMAT (7H  TIMES /1PE12.3)                                       NRU 1810
      DO 275 IT=IT1,IT2                                                 NRU 1820
      INDT = (I-1)*NT + IT + LOCT                                       NRU 1830
      ITR1 = LOCQT + NTNDNR + (I-1)*NTNR + (IT-1)*NRESP + IR1           NRU 1840

      ITR2 = ITR1 + IR2 - IR1                                           NRU 1850
      WRITE (IO,1070)          (E(IP),IP=ITR1,ITR2)                     NRU 1860
      ITR1 = ITR1 + NTNDNR                                              NRU 1870
      ITR2 = ITR2 + NTNDNR                                              NRU 1880
      WRITE (IO,1080) (E(IP),IP=ITR1,ITR2)                              NRU 1890
 275  WRITE (IO,1090) A(INDT)                                           NRU 1900
      IF (NE) 475,475,280                                               NRU 1910
 280  NEM = (NE-1)/10 + 1                                               NRU 1920
      NTN = (NT-1)/17 + 1                                               NRU 1930
      INP = IN1                                                         NRU 1940
C  *  *  OUTPUT SQTE AND SQTE2 IN THE FOLLOWING LOOP                    NRU 1970
      DO 370 I=1,ND                                                     NRU 1960
```

```
      DO 370 INE=1,NEN                                                  NRU 1970
      IE1 = (INE-1)*10 + 1                                              NRU 1980
      IF (INE-NEN) 285,290,290                                          NRU 1990
  285 IE2 = IE1 + 9                                                     NRU 2000
      GO TO 295                                                         NRU 2010
  290 IE2 = NE                                                          NRU 2020
  295 IF (INE-1) 300,300,305                                            NRU 2030
  300 ETOP = EFIRST                                                     NRU 2040
      GO TO 310                                                         NRU 2050
  305 ETOP = A(LOCIB + NE + IE1 - 1)                                    NRU 2060
      IF (IE1.EQ.NNE+1) ETOP=EGTOP                                      NRU 2070
  310 I1E = LOCIB + NE + IE1                                            NRU 2080
      DO 370 INT=1,NTM                                                  NRU 2090
      I2 = LOCIB + NE + IE2 - 1                                         NRU 2100
      IST = LOCQAE -  9                                                 NRU 2110
      WRITE (IO,1130) INP,I,(ANUMB(IPR),IPR=IST,LOCQAE)                 NRU 2120
 1130 FORMAT (A1,11HDETECTOR NO,I3,8X,29HFLUENCE(TIME,ENERGY,DETECTOR), NRU 2130
     1 1X,10A8)                                                         NRU 2131
      IF (NT-8) 315,315,320                                             NRU 2140
  315 INP = IN2                                                         NRU 2150
  320 IT1 = (INT-1)*17 + 1                                              NRU 2160
      IF (INT-NTM) 325,330,330                                          NRU 2170
  325 IT2 = IT1 + 16                                                    NRU 2180
      GO TO 335                                                         NRU 2190
  330 IT2 = NT                                                          NRU 2200
  335 IX = LOCIB + NE + NNE                                             NRU 2210
      ESAV = A(IX)                                                      NRU 2220
      IF (NNE .LT. IE1 .OR. NNE .GE. IE2) GO TO 340                     NRU 2230
      A(IX) = EGTOP                                                     NRU 2240
  340 IF (I1E .LE. I2) GO TO 345                                        NRU 2250
      WRITE (IO,1140) ETOP                                              NRU 2260
      GO TO 350                                                         NRU 2270
  345 WRITE (IO,1140) ETOP,(A(IP),IP=I1E,I2)                            NRU 2280
 1140 FORMAT (1H0,3X,8HENERGIES,5X,1P10E10.3)                           NRU 2290
  350 A(IX) = ESAV                                                      NRU 2300
      I2 = I2 + 1                                                       NRU 2310
      IF (INT-1) 355,355,360                                            NRU 2320
  355 AGS = A(LOCXD + 4*ND + I)                                         NRU 2330
      GO TO 365                                                         NRU 2340
  360 ISUB = LOCT + (I-1)*NT + IT1 - 1                                  NRU 2350
      AGS = A(ISUB)                                                     NRU 2360
  365 WRITE (IO,1150) (A(IP),IP=I1E,I2)                                 NRU 2370
 1150 FORMAT (2X,5HTIMES,10X,1P10E10.3)                                 NRU 2380
      WRITE (IO,1160) AGS                                               NRU 2390
 1160 FORMAT (1PE12.3)                                                  NRU 2400
      DO 370 IT=IT1,IT2                                                 NRU 2410
      INDT = (I-1)*NT + IT + LOCT                                       NRU 2420
      ITE1 = (I-1)*NTNE + (IE1-1)*NT + IT + LOCQTE + NTNEND             NRU 2430

      ITE2 = ITE1 + (IE2-IE1)*NT                                        NRU 2440
      WRITE (IO,1070)         (E(IP),IP=ITE1,ITE2,NT)                   NRU 2450
      ITE1 = ITE1 + NTNEND                                              NRU 2460
```

```
      ITE2 = ITE2 + NTNEND                                              NRU 2470
      WRITE (IO,1080) (E(IP),IP=ITE1,ITE2,NT)                           NRU 2480
  370 WRITE (IO,1090) A(INDT)                                           NRU 2490
  375 IF (NA) 475,475,380                                               NRU 2500
  380 NEM = (NE-1)/10 + 1                                               NRU 2510
      NAM = (NA-1)/17 +1                                                NRU 2520
      IHP = IH1                                                         NRU 2530
C  *  *  OUTPUT SQAE AND SQAE2 IN THE FOLLOWING LOOP                    NRU 2540
      DO 470 I=1,ND                                                     NRU 2550
      DO 470 INE=1,NEM                                                  NRU 2560
      IE1 = (INE-1)*10 + 1                                              NRU 2570
      IF (INE-NEM) 385,390,390                                          NRU 2580
  385 IE2 = IE1 + 9                                                     NRU 2590
      GO TO 395                                                         NRU 2600
  390 IE2 = NE                                                          NRU 2610
  395 IF (INE-1) 400,400,405                                            NRU 2620
  400 ETOP = EFIRST                                                     NRU 2630
      GO TO 410                                                         NRU 2640
  405 ETOP = A(LOCIB + NE + IE1 - 1)                                    NRU 2650
  410 I1E = LOCIB + NE + IE1                                            NRU 2660
      DO 470 INA=1,NAM                                                  NRU 2670
      I2 = LOCIB + NE + IE2 - 1                                         NRU 2680
      IST = LMAX -  9                                                   NRU 2690
      WRITE (IO,1170) IHP,I,(ANUMB(IPR),IPR=IST,LMAX)                   NRU 2700
 1170 FORMAT (A1,11HDETECTOR NO.I3,6X,31HFLUENCE(COSINE,ENERGY,DETECTOR)NRU 2710
     1,1X,10A8)                                                         NRU 2711
      IF (NA-8) 415,415,420                                             NRU 2720
  415 IHP = IH2                                                         NRU 2730
  420 IX = LOCIB + NE + NNE                                             NRU 2740
      ESAV = A(IX)                                                      NRU 2750
      IF (NNE .LT. IE1 .OR. NNE .GE. IE2) GO TO 425                     NRU 2760
      A(IX) = EGTOP                                                     NRU 2770
  425 IF (I1E .LE. I2) GO TO 430                                        NRU 2780
      WRITE (IO,1140) ETOP                                              NRU 2790
      GO TO 435                                                         NRU 2800
  430 WRITE (IO,1140) ETOP,(A(IP),IP=I1E,I2)                            NRU 2810
  435 A(IX) = ESAV                                                      NRU 2820
      I2 = I2 + 1                                                       NRU 2830
      IA1 = (INA-1)*17 + 1                                              NRU 2840
      IF (INA-1) 440,440,445                                            NRU 2850
  440 FM1 = -1.0                                                        NRU 2860
      GO TO 450                                                         NRU 2870
  445 FM1 = A(LOCCO + IA1 - 1)                                          NRU 2880
  450 WRITE (IO,1160) (A(IP),IP=I1E,I2)                                 NRU 2890
 1160 FORMAT (5X,7HCOSINES,5X,1P10E10.3)                                NRU 2900
      WRITE (IO,1190) FM1                                               NRU 2910
 1190 FORMAT (2X,F12.6)
      IF (INA-NAM) 455,460,460                                          NRU 2930
  455 IA2 = IA1 + 16                                                    NRU 2940
      GO TO 465                                                         NRU 2950
  460 IA2 = NA                                                          NRU 2960
```

```
  465 DO 470 IA=IA1,IA2                                                 NRU 2970
      ICO = LOCCO + IA                                                  NRU 2980
      IAE1 = (I-1)*NANE + (IE1-1)*NA + IA + LOCQAE + NANEND             NRU 2990
      IAE2 = IAE1 + (IE2-IE1)*NA                                        NRU 3000
      WRITE (IO,1070)        (A(IP),IP=IAE1,IAE2,NA)
      IAE1 = IAE1 + NANEND                                              NRU 3020

      IAE2 = IAE2 + NANEND                                              NRU 3030
      WRITE (IO,1080) (A(IP),IP=IAE1,IAE2,NA)
  470 WRITE (IO,1190) A(ICO)
  475 IF (NEXND) 490,490,480                                            NRU 3050
C       OUTPUT EXTRA ARRAYS OF LENGTH ND  *    *    *    *   *          NRUN * *
  480 WRITE (IO,1200)                                                   NRU 3060
 1200 FORMAT (26H1EXTRA ARRAYS OF LENGTH ND/)                           NRU 3070
      DO 485 I=1,NEXND                                                  NRU 3080
      STR = 0.0                                                         NRU 3090
      CALL INSERT(STR,1,4,4HEXTR)                                       NRU 3100
      CALL INTBCD(I,DUM,L)                                              NRU 3110
      LP = 5 - L                                                        NRU 3120
      CALL INSERT(STR,LP,L,DUM)                                         NRU 3130
      IST = LOCXD + (5+I)*ND + 1                                        NRU 3140
  485 CALL HELPER(A(IST),1,ND,STR,IO)                                   NRU 3150
  490 CALL ENDRUN                                                       NRU 3160
      RETURN                                                            NRU 3170
      END                                                               NRU 3180

C$
      SUBROUTINE OUTPT(KEY)                                             OUTPT 10
C * * THIS VERSION OF OUTPT  IS FOR MORSE-CGA * * * *
C     THIS ROUTINE CONTROLS CALCULATION AND OUTPUT OF AVERAGE VALUES OF OUTPT  *
C       SOURCE PARAMETERS (KEY=1) ,THE COLLISION COUNTERS AT THE END OF OUTPT  *
C        EACH BATCH (KEY=2), AND OUTPUTS THE COUNTERS FOR NUMBER OF     OUTPT  *
C        SCATTERINGS,ETC., AT END OF RUN (KEY=3)                        OUTPT  *
      COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDOUTPT 30
     1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGEOUTPT 31
      COMMON /APOLLO/ AGSTRT,DDF,DEADWT(5),ETA,ETATH,ETAUSD,UINP,VINP,  OUTPT 40
     1 WINP,WTSTRT,XSTRT,YSTRT,ZSTRT,TCUT,XTRA(10),                     OUTPT 41
     2   IO,I1,MEDIA,IADJM,ISBIAS,ISOUR,ITERS,ITIME,ITSTR,LOCWTS,LOCFWL,OUTPT 42
     3 LOCEPR,LOCNSC,LOCFSN,MAXGP,MAXTIM,MEDALB,NGPREG,MXREG,NALB,      OUTPT 43
     4 NDEAD(5),NEWNM,NGEOM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,    OUTPT 44
     5 NKCALC,NKILL,NLAST,NMEM,NMGP,NMOST,NMTG,NOLEAK,NORMF,NPAST,      OUTPT 45
     6 NPSCL(13),NQUIT,NSIGL,NSOUR,NSPLT,NSTRT,NXTRA(10)                 OUTPT 46
      COMMON /FISBNK/ MFISTP,NFISBN,NFISH,FTOTL,FWATE,WATEF             OUTPT 50
      COMMON /OUTB/ SWATE,XAVE,YAVE,ZAVE,EAVE,UAVE,VAVE,WAVE,AGEAVE     GSTOR 40
      COMMON WTS(1)                                                     OUTPT 60
      DIMENSION NC(1)                                                   OUTPT 70
      EQUIVALENCE (WTS(1),NC(1))                                        OUTPT 80
      REAL*8 RANDOM                                                     OUTPT
      DATA  FNK/0.0/, FKSUM/0.0/, VARK/0.0/, FNKFW/0.0/, FKFW/0.0/      OUTPT 90
      DATA  NITSX/0/                                                    OUTP 100
      GO TO (1111,2222,3333), KEY                                       OUTP 110
 1111 NBATCH=NITS-ITERS+1                                               OUTP 120
```

```
      IF (NBATCH-1) 20,15,20                                            OUTP 130
15    ITCUT = 1                                                         OUTP 140
      FNKFW =0.                                                         OUTP 150
      FKFW =0.                                                          OUTP 160
20    FNKT = NMEM                                                       OUTP 170
      FNKFW = FNKFW + FNKT                                              OUTP 180
      IF(SWATE.EQ.0.) GO TO 30                                          OUTP 430
      XAVE=XAVE/SWATE                                                   OUTP 440
      YAVE=YAVE/SWATE                                                   OUTP 450
      ZAVE=ZAVE/SWATE                                                   OUTP 460
      EAVE=EAVE/SWATE                                                   OUTP 470
      UAVE=UAVE/SWATE                                                   OUTP 480
      VAVE=VAVE/SWATE                                                   OUTP 490
      WAVE=WAVE/SWATE                                                   OUTP 500
      AGEAVE=AGEAVE/SWATE                                               OUTP 510
30    WRITE (IO,1010)                                                   OUTP

1010 FORMAT ('   WTAVE        IAVE    UAVE     VAVE     WAVE       XAVE   OUTP
     1    YAVE         ZAVE        AGEAVE')                             OUTP
      WRITE (IO,1011)SWATE,EAVE,UAVE,VAVE,WAVE,XAVE,YAVE,ZAVE,AGEAVE    OUTP
1011 FORMAT(1H ,1PE10.3,2X,0PF7.2,2X,3(F7.4,1X),1X,3(1PE10.3,1X),1X,    OUTP
     1E10.3)                                                            OUTP
      RETURN                                                            OUTP 540
2222 WRITE(IO,1030)                                                     OUTP 550
1030 FORMAT(/35H0NUMBER OF COLLISIONS OF TYPE NCOLL/3X,6HSOURCE,1X,     OUTP 560
     1  8HSPLIT(D),4X,5HFISSN,3X,6HGAMGEN,1X,8HREALCOLL ,3X,6HALBEDO,4X,OUTP 561
     2 5HBDRYX,3X,6HESCAPE,4X,5HE-CUT,1X,8HTIMEKILL ,1X,8HR R KILL,1X,  OUTP 562
     3 8HR R SURV, 2X,7HGAMLOST)                                        OUTP 563
      WRITE (IO,1040) (NPSCL(I),I=1,  13)                               OUTP 570
1040 FORMAT(13I9)                                                       OUTP 580
      LENGTH = 0                                                        OUTP 590
      CALL TIMER(LENGTH,XTRA)                                           OUTP 600
      WRITE (IO,1050) (XTRA(I),I=1,LENGTH)                              OUTP 610
1050 FORMAT (43H0TIME REQUIRED FOR THE PRECEDING BATCH WAS ,10A4)       OUTP 620
      IF (ITCUT) 50,50,40                                               OUTP 630
40    IF (NPSCL(10)) 50,50,45                                           OUTP 640
45    ITCUT = 0                                                         OUTP 650
50    DO 55 I=1,13                                                      OUTP 660
55    NPSCL(I)=0                                                        OUTP 670
      IF (NORMF) 65,65,60                                               OUTP 680
60    FKFW = FKFW + FNKT*FTOTL/SWATE                                    OUTP 690
      IF (NFISH) 85,85,65                                               OUTP 700
65    ESTK = FKFW/FNKFW*NFISH/NSTRT                                     OUTP 710
      IF (ESTK) 85,85,70                                                OUTP 720
70    DO 75 I=1,NXREG                                                   OUTP 730
75    WTS(LOCFWL+I) = WTS(LOCFWL+I)*ESTK                                OUTP 740
      WTCHNG = FWATE/NFISH/SWATE*FNKT                                   OUTP 750
      NG1 = LOCWTS+1                                                    OUTP 760
      NG2 = LOCWTS+3*NGPREG                                             OUTP 770
      DO 80 I=NG1,NG2                                                   OUTP 780
80    WTS(I) = WTS(I)*WTCHNG                                            OUTP 790
85    IF (NKCALC) 105,105,90                                            OUTP 800
```

```
 90   FKT = FTOTL/SWATE                                                 OUTP 810
      IF (NBATCH .LT. NKCALC .OR. ITCUT .EQ. 0) GO TO 95                OUTP 820
      FKSUM = FKSUM + FNKT*FKT                                          OUTP 830
      VARK = VARK + FNKT*FKT*FKT                                        OUTP 840
      FNK = FNK + FNKT                                                  OUTP 850
      NITSK = NITSK + 1                                                 OUTP 860
      GO TO 100                                                         OUTP 870
 95   WRITE(IO,1060)                                                    OUTP 880
 1060 FORMAT(63H --- K FOR THIS BATCH WILL NOT BE USED IN AVERAGE K CALCOUTP 890
     1ULATION )                                                         OUTP 891
 100  WRITE (IO,1070) FKT      ,FTOTL,FWATE,NFISH                       OUTP 900
 1070 FORMAT(1H0/3H K=,F9.5,15X,6HFTOTL=,E13.5,5X,6HFWATE=,E13.5,       OUTP 910
     15X,6HNFISH=,I5)                                                   OUTP 911
 105  RETURN                                                            OUTP 920
 3333 WRITE (IO,1080) (NDEAD(I),DEADWT(I),I=1,4)                        OUTP 930
 1080 FORMAT(16H1NEUTRON DEATHS,20X,6HNUMBER,16X,6HWEIGHT/              OUTP 940
     127H0KILLED BY RUSSIAN ROULETTE,8X,I6, 9X,E13.5/                   OUTP 941
     28H ESCAPED,27X,I6, 9X,E13.5/                                      OUTP 942
     322H REACHED ENERGY CUTOFF,13X,I6, 9X,E13.5/20H REACHED TIME CUTOFFOUTP 943
     4 15X,I6, 9X,E13.5)                                                OUTP 944
      IF(MEDIA)125,125,117                                              OUTP 945
 117  WRITE(IO,1090)                                                    OUTP 950
 1090 FORMAT(1H0/22H0NUMBER OF SCATTERINGS)                             OUTP 960
C                                                                       OUTP 960
      WRITE(IO,1100)                                                    OUTP 980

 1100 FORMAT(7H0MEDIUM 13X,6HNUMBER)                                    OUTP 990
      N123=0                                                            OUTP 985
      DO 115 NMED=1,MEDIA                                               OUT 1000
      N1 = LOCNSC + 8*NMTG*MXREG + NMED                                 OUT 1010
      N123=N123+NC(N1)                                                  OUT
 115  WRITE(IO,1110)NMED,WTS(N1)                                        OUT 1020
 1110 FORMAT (I5,10X,I7)                                                OUT 1030
      WRITE (IO,1112) N123                                              OUT
 1112 FORMAT (' TOTAL', 9X,I7)                                          OUT
 120  WRITE (IO,1120)                                                   OUT 1050
 1120 FORMAT ('1REAL SCATTERING COUNTERS')                              OUT 1060
      N1 = LOCNSC + 1                                                   OUT 1070
      CALL OUTPT2(WTS(N1),NC(N1),NMTG,MXREG,IO)                         OUT 1080
 125  IF (MEDALB-7777) 130,135,130                                      OUT 1090
 130  WRITE (IO,1130)                                                   OUT 1100
 1130 FORMAT ('1ALBEDO SCATTERING COUNTERS')                            OUT 1110
      N1 = LOCNSC + 2*NMTG*MXREG + 1                                    OUT 1120
      CALL OUTPT2(WTS(N1),NC(N1),NMTG,MXREG,IO)                         OUT 1130
 135  IF (MFISTP) 145,145,140                                           OUT 1140
 140  WRITE (IO,1140)                                                   OUT 1150
 1140 FORMAT ('1FISSION PRODUCTION COUNTERS')                           OUT 1160
      N1 = LOCNSC + 4*NMTG*MXREG + 1                                    OUT 1170
      CALL OUTPT2(WTS(N1),NC(N1),NMTG,MXREG,IO)                         OUT 1180
 145  IF (NGPQTN*NGPQTG) 150,155,150                                    OUT 1190
 150  WRITE (IO,1150)                                                   OUT 1200
 1150 FORMAT ('1SECONDARY PRODUCTION COUNTERS (BOTH THE GROUPS CAUSING POUT 1210
```

```
     1RODUCTION AND RESULTING FROM PRODUCTION')                         OUT 1211
      N1 = LOCNSC + 6*NNTG*NXREG + 1                                    OUT 1220
      CALL OUTPT2(WTS(N1),NC(N1),NNTG,NXREG,IO)                         OUT 1230
155   IF (NSPLT) 165,165,160                                            OUT 1240
160   WRITE(IO,1160)                                                    OUT 1250
1160  FORMAT(21H1NUMBER OF SPLITTINGS)                                  OUT 1260
      N1 = LOCWTS+4*NGPREG+1                                            OUT 1270
      CALL OUTPT2(WTS(N1),NC(N1),MAXGP,NXREG,IO)                        OUT 1280
      WRITE (IO,1170)                                                   OUT 1290
1170  FORMAT(47H1NUMBER OF SPLITTINGS PREVENTED BY LACK OF ROOM)        OUT 1300
      N1 = N1 + 2*NGPREG                                                OUT 1310
      CALL OUTPT2(WTS(N1),NC(N1),MAXGP,NXREG,IO)                        OUT 1320
165   IF(NKILL)175,175,170                                              OUT 1330
170   WRITE(IO,1180)                                                    OUT 1340
1180  FORMAT(33H1NUMBER OF RUSSIAN ROULETTE KILLS)                      OUT 1350
      N1 = LOCWTS + 8*NGPREG + 1                                        OUT 1360
      CALL OUTPT2(WTS(N1),NC(N1),MAXGP,NXREG,IO)                        OUT 1370
      WRITE(IO,1190)                                                    OUT 1380
1190  FORMAT(37H1NUMBER OF RUSSIAN ROULETTE SURVIVALS)                  OUT 1390
      N1 = N1 + 2*NGPREG                                                OUT 1400
      CALL OUTPT2(WTS(N1),NC(N1),MAXGP,NXREG,IO)                        OUT 1410
175   IF(NKCALC)185,185,180                                             OUT 1420
180   VARK = VARK/FNK                                                   OUT 1430
      FKSUM = FKSUM/FNK                                                 OUT 1440
      VARK = SQRT ((VARK-FKSUM**2)/(NITSK-1))                           OUT 1450
      WRITE(IO,1200)  FKSUM,VARK,NITSK                                  OUT 1460
1200  FORMAT(11H1AVERAGE K=,E15.4,10X,14HSTANDARD DEV.=,E15.4,5X,3HFOR,IOUT 1470
     14,8H BATCHES)                                                     OUT 1471
185   FNK=  0.                                                          OUT 1480
      FKSUM=0.                                                          OUT 1490
      VARK=0.                                                           OUT 1500
      NITSK=0                                                           OUT 1510
      CALL RNDOUT(RANDOM)                                               OUT

      WRITE (IO,1210) RANDOM                                            OUT
1210  FORMAT('0    ** NEXT RANDOM NUMBER IS  ',Z12)                     OUT
      RETURN                                                            OUT 1520
      END                                                               OUT 1530

      SUBROUTINE  SCORIN                                                SCORI 10
C                                                                       * * * *
C     THIS ROUTINE READS INPUT DATA FOR THE ANALYSIS MODULE  * *  *  *  *  *  *
C                                                                       * * * *
      REAL * 8 SS, LNK1
      COMMON /USER/ AGSTRT,WTSTRT,XSTRT,YSTRT,ZSTRT,DFF,EBOTN,EBOTG,    SCORI 20
     1 TCUT,IO,I1,IADJM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,NLAST,  SCORI 21
     2 NLEFT,NMGP,NNTG,NSTRT                                            SCORI 22
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,SCORI 30
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCED,LOCIB,LOCCO,LOCT,LOCUD,   SCORI 31
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                SCORI 32
      COMMON BLNK(1)                                                    SCORI 40
      COMMON /SSS/ SS(1)
```

```
      DIMENSION LNK(1),LNK1(1),IHOL(20),IHF(6),IHA(6),IHP(6)            SCORI 50
      EQUIVALENCE(BLNK(1),LNK(1)),(SS(1),LNK1(1))                       SCORI 60
      DATA IHOLB,IHF,IHA/1H ,4HPRIM,4HARY ,4HENER,4HGY B,4HINS ,1H ,    SCORI 70
     1 4HSECO,4HNDAR,4HY EN,4HERGY,4H BIN,1HS/,JUP/3HUPP/,JLO/3HLOW/    SCORI 71

C**** WILSON   JUNE 2, 1989  ****                                       MOR 1150
C     LOCUD = LOCT + 2*ND*NT                                            SCOR 680
      LOCUD = 0                                                         SCOR 680
C**** WILSON   JUNE 2, 1989  ****                                       MOR 1150
      LOCSD = LOCUD + 3*NDNR                                            SCOR 690
C**** WILSON   APRIL 19,1989 ****                                       MOR 1150
C     LOCQE = LOCSD + 3*NDNR + 20                                       SCOR 700
      LOCQE = LOCSD + 14*NDNR + 20                                      SCOR 700
C**** WILSON   APRIL 19,1989 ****                                       MOR 1150


      RETURN                                                            SCO 3330
      END                                                               SCO 3340

      SUBROUTINE STBTCH (NBAT)                                          STBTC 10
C      THIS ROUTINE IS CALLED AT THE START OF EACH BATCH                STBTC  *
C     THIS ROUTINE ZEROES THE ARRAYS USED TO ACCUMULATE ESTIMATED       STBTC  *
C     QUANTITIES DURING A BATCH .                                       STBTC  *
C     AT THE START OF THE FIRST BATCH, THE ARRAYS WHICH ACCUMULATE      STBTC  *
C        ESTIMATES AND SQUARED ESTIMATES ARE ALSO ZEROED .   *    *     STBTC  *
      REAL *  8 BL
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,STBTC 30
     1 NANE,NTNDNR,NTNEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   STBTC 31
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                STBTC 32
      COMMON /SSS/ BL(1)                                                STBTC 40
      IF (NBAT) 10,20,150                                               STBTC 50
 10   CALL ERROR                                                        STBTC 60
 20   IA = LOCUD + 1                                                    STBTC 70
C**** WILSON   APRIL 18,1989 ****                                       MOR 1150
C     IB = IA + 6*NDNR - 1                                              STBTC 80
      IB = IA +17*NDNR - 1                                              STBTC 80
C**** WILSON   APRIL 18,1989 ****                                       MOR 1150
      DO 30 I=IA,IB                                                     STBTC 90
 30   BL(I) = 0.0                                                       STBT 100
      IF (NE) 60,60,40                                                  STBT 110
 40   IA = LOCQE + 1                                                    STBT 120
      IB = IA + 3*NEND - 1                                              STBT 130
      DO 50 I=IA,IB                                                     STBT 140
 50   BL(I) = 0.0                                                       STBT 150
 60   IF (NT) 110,110,70                                                STBT 160

 70   IA = LOCQT + 1                                                    STBT 170
      IB = IA + 3*NTNDNR - 1                                            STBT 180
      DO 80 I=IA,IB                                                     STBT 190
 80   BL(I) = 0.0                                                       STBT 200
      IF (NE) 140,140,90                                                STBT 210
 90   IA = LOCQTE + 1                                                   STBT 220
```

```
      IB = IA + 3*NTHEND - 1                                            STBT 230
      DO 100 I=IA,IB                                                    STBT 240
  100 BL(I) = 0.0                                                       STBT 250
  110 IF (NA*NE) 140,140,120                                            STBT 260
  120 IA = LOCQAE + 1                                                   STBT 270
      IB = IA + 3*NANEND - 1                                            STBT 280
      DO 130 I=IA,IB                                                    STBT 290
  130 BL(I) = 0.0                                                       STBT 300
  140 RETURN                                                            STBT 310
  150 IA = LOCSD                                                        STBT 330
      IJ = LOCSD + 4*NDNR                                               STBT 330
      IV = LOCSD + 5*NDNR                                               STBT 330
      DO 290 I=1,ND                                                     STBT 380
      DO 210 M=1,NRESP                                                  STBT 430
      IA = IA + 1                                                       STBT 450
      IJ = IJ + 1                                                       STBT 450
      IV = IV + 1                                                       STBT 450
      BL(IA) = 0.0                                                      STBT 470
      BL(IJ) = 0.0                                                      STBT 470
      BL(IV) = 0.0                                                      STBT 470
  210 CONTINUE                                                          STBT 650
  290 CONTINUE                                                          STBT 650
      RETURN                                                            STBT 650
      END                                                               STBT 660

      SUBROUTINE STPART                                                 STBTC 10
C      THIS ROUTINE IS CALLED AT THE START OF EACH SOURCE PARTICLE      STBTC  *
C     THIS ROUTINE ZEROES THE ARRAYS USED TO ACCUMULATE ESTIMATED       STBTC  *
C     QUANTITIES DURING A BATCH .                                       STBTC  *
      REAL * 8 BL
      COMMON /PDET/ ND,NNE,NE,NT,NA,NRESP,NEX,NEXND,NEND,NDNR,NTNR,NTNE,STBTC 30
     1 NANE,NTHDNR,NTHEND,NANEND,LOCRSP,LOCXD,LOCIB,LOCCO,LOCT,LOCUD,   STBTC 31
     2 LOCSD,LOCQE,LOCQT,LOCQTE,LOCQAE,LMAX,EFIRST,EGTOP                STBTC 32
      COMMON /SSS/ BL(1)                                                STBTC 40
      IA = LOCUD                                                        STBT 320
      IJ = LOCSD + 3*NDNR                                               STBT 330
      IC = LOCQE                                                        STBT 340
      ID = LOCQT                                                        STBT 350
      IE = LOCQTE                                                       STBT 360
      IF = LOCQAE                                                       STBT 370
      DO 290 I=1,ND                                                     STBT 380
      IF (NE) 180,180,160                                               STBT 390
  160 DO 170 J=1,NE                                                     STBT 400
      IC = IC + 1                                                       STBT 410
  170 BL(IC) = 0.0                                                      STBT 420
  180 DO 210 M=1,NRESP                                                  STBT 430
      IA = IA + 1                                                       STBT 440
      IJ = IJ + 1                                                       STBT 450
      BL(IA) = 0.0                                                      STBT 460
      BL(IJ) = 0.0                                                      STBT 470
      IF (NT) 210,210,190                                               STBT 480
  190 DO 200 K=1,NT                                                     STBT 490
```

```
      ID = ID + 1                                                       STBT 500
  200 BL(ID) = 0.0                                                      STBT 510
  210 CONTINUE                                                          STBT 520

      IF (NE) 290,290,220                                               STBT 530
  220 DO 280 J=1,NE                                                     STBT 540
      IF (NT) 250,250,230                                               STBT 550
  230 DO 240 K=1,NT                                                     STBT 560
      IE = IE + 1                                                       STBT 570
  240 BL(IE) = 0.0                                                      STBT 580
  250 IF (NA) 280,280,260                                               STBT 590
  260 DO 270 L=1,NA                                                     STBT 600
      IF = IF + 1                                                       STBT 610
  270 BL(IF) = 0.0                                                      STBT 620
  280 CONTINUE                                                          STBT 630
  290 CONTINUE                                                          STBT 640
      RETURN                                                            STBT 650
      END                                                               STBT 660

      SUBROUTINE TESTW                                                  TESTW 10
C * * THIS VERSION OF TESTW IS FOR MORSE-CGA * * *
C      THIS ROUTINE TESTS WHETHER RUSSIAN ROULETTE OR SPLITTING OPTIONS TESTW **
C         ARE IN EFFECT AND THEN PERFORMS THE SPLITTING OR ROULETTE     TESTW **
C    SPLITTING IS PERFORMED UNTIL THE WEIGHT FALLS BELOW WTHIR           TESTW **
      COMMON /NUTRON/ NAME,NAMEX,IG,IGO,NMED,MEDOLD,NREG,U,V,W,UOLD,VOLDTESTW 20
     1 ,WOLD,X,Y,Z,XOLD,YOLD,ZOLD,WATE,OLDWT,WTBC,BLZNT,BLZON,AGE,OLDAGETESTW 21
      COMMON /APOLLO/ AGSTRT,DDF,DEADWT(5),ETA,ETATH,ETAUSD,UINP,VINP,  TESTW 30
     1 WINP,WTSTRT,XSTRT,YSTRT,ZSTRT,TCUT,XTRA(10),                     TESTW 31
     2   IO,I1,MEDIA,IADJM,ISBIAS,ISOUR,ITERS,ITIME,ITSTR,LOCWTS,LOCFWL,TESTW 32
     3 LOCEPR,LOCNSC,LOCFSN,MAXGP,MAXTIM,MEDALB,MGPREG,MXREG,NALB,      TESTW 33
     4 NDEAD(5),NEWNM,NGEOM,NGPQT1,NGPQT2,NGPQT3,NGPQTG,NGPQTN,NITS,    TESTW 34
     5 NKCALC,NKILL,NLAST,NMEM,NMGP,NMOST,NMTG,NOLEAK,NORMF,NPAST,      TESTW 35
     6 NPSCL(13),NQUIT,NSIGL,NSOUR,NSPLT,NSTRT,NXTRA(10)                TESTW 36
      COMMON /NS/ NS                                                    GSTOR 40
      COMMON NWTS(1)                                                    TESTW 40
      DIMENSION WTS(1)                                                  TESTW 50
      EQUIVALENCE (NWTS(1),WTS(1))                                      TESTW 60
      IF (IG-MAXGP) 10,10,65                                            TESTW 70
   10 IF (NKILL+NSPLT) 65,65,15                                         TESTW 80
   15 NWT = (NREG-1)*MAXGP+IG+LOCWTS                                    TESTW 90
      WTHIR = WTS(NWT)                                                  TEST 100
      NWT = NWT+MGPREG                                                  TEST 110
      WTLOR = WTS(NWT)                                                  TEST 120
      NWT = NWT+MGPREG                                                  TEST 130
      WTAVR = WTS(NWT)                                                  TEST 140
      IF (NKILL) 45,45,20                                               TEST 150
   20 IF(WTLOR-ABS (WATE))40,40,25                                      TEST 160
   25 IF(FLTRNF(NARG)*WTAVR-ABS (WATE))35,35,30                         TEST 170
   30 WATE=0.                                                           TEST 180
      NWT = NWT + 6*MGPREG                                              TEST 190
      NWTS(NWT)=NWTS(NWT)+1                                             TEST 200
      NWT = NWT + MGPREG                                                TEST 210
```

```
      WTS(NWT) = WTS(NWT) + OLDWT                                       TEST 220
C  WEIGHT KILLED ENTERING COLLISION IS SCORED                           TEST  *
      NPSCL(11) = NPSCL(11) + 1                                         TEST 240
C     CALL BANKR(11)  FOR R R KILL ANALYSIS                             TEST 230
C  R R KILL                                                             TEST 240
      RETURN                                                            TEST 270
 35   W1=WATE                                                           TEST 280
      WATE=SIGN (WTAVR,W1)                                              TEST 290
      NWT = NWT + 8*NGPREG                                              TEST 300
      NWTS(NWT)=NWTS(NWT)+1                                             TEST 310
      NWT = NWT + NGPREG                                                TEST 320
      WTS(NWT) = WTS(NWT) + W1                                          TEST 330
C  WEIGHT ENTERING COLLISION (BUT BELOW WTLOR) AND SURVIVING IS SCORED  TEST  *

      NPSCL(12) = NPSCL(12) + 1                                         TEST 350
C     CALL BANKR(12)  FOR R R SURV ANALYSIS                             TEST 310
C  R R SURVIVAL                                                         TEST 320
      OLDWT=WATE                                                        TEST 380
      RETURN                                                            TEST 390
 40   IF (NSPLT) 65,65,45                                               TEST 400
 45   IF(WTHIR-ABS (WATE)) 50, 65, 65                                   TEST 410
 50   IF(NMOST-NS) 60,60,55                                             TEST 420
 55   NS=NS+1                                                           TEST 430
      NEWNM=NEWNM+1                                                     TEST 440
      WATE=WATE*.5                                                      TEST 450
      NAME1=NAME                                                        TEST 460
      NAME=NEWNM                                                        TEST 470
      CALL STORNT(NS,1)                                                 TEST 480
      NPSCL(2) = NPSCL(2) + 1                                           TEST 490
C     CALL BANKR(2)  FOR SPLIT DAUGHTER ANALYSIS                        TEST 450
C  SPLITD                                                               TEST 480
      NAME=NAME1                                                        TEST 520
      OLDWT=WATE                                                        TEST 530
      NWS = NWT+2*NGPREG                                                TEST 540
      NWTS(NWS) = NWTS(NWS)+1                                           TEST 550
      NWS = NWS + NGPREG                                                TEST 560
      WTS(NWS) = WTS(NWS) + WATE                                        TEST 570
C  WEIGHT AFTER SPLITTING IS SCORED                                     TEST  *
      GO TO 45                                                          TEST 590
 60   NWS = NWT + 4*NGPREG                                              TEST 600
      NWTS(NWS) = NWTS(NWS)+1                                           TEST 610
      NWS = NWS + NGPREG                                                TEST 620
      WTS(NWS) = WTS(NWS) + WATE                                        TEST 630
C  WEIGHT WHICH COULD HAVE SPLIT IS SCORED                              TEST  *
 65   RETURN                                                            TEST 650
      END                                                               TEST 660

      SUBROUTINE FTEST(SX, SY,SZ,N1,N2,NBAT,NPAR)                       VAR2  10
C    NBAT IS THE NO. OF INDEPENDENT BATCHES                             VAR2  20
C    NPAR IS THE TOTAL NUMBER OF PARTICLES PROCESSED                    VAR2  30
C    IT IS ASSUMED THAT THE SUMSQ ARRAY HAS ACCUMULATED THE NUMBER OF PARTICLES
C      TIMES THE SQUARE OF THE BATCH AVERAGE (THIS IS OBTAINED BY DIVIDING    50
```

```
C      THE SQUARED BATCH SUM BY THE NUMBER OF PARTICLES STARTING THE BATCH)   60
      REAL * 8 SX,  SY, SZ,AN, AI, AJ,SSB,SSW,SST
      DIMENSION SX(M1,M2), SY(M1,M2), SZ(M1,M2)                       VAR2  70
      AN = NPAR
      AI = NBAT
      AJ = AN/AI
      DO 29 I=1,M1
      DO 29 J=1,M2
       SSB = AJ*(   SX(I,J)/AJ-SZ(I,J)**2/AJ/AN)
       SST = SY(I,J) - SZ(I,J)**2/AN
       SSW = SST - SSB
      SY(I,J) = (SSB/(AI-1.))/(SSW/(AI*(AJ-1.)))
C     SY(I,J) = DSQRT(SSB/(AI-1.)/AN)/(SZ(I,J)/AN)
C     SY(I,J) = DSQRT(SSW/(AI*(AJ-1.))/AN)/(SZ(I,J)/AN)
C     SY(I,J) = DSQRT(SST/(AN-1.)/AN)/(SZ(I,J)/AN)
  29  CONTINUE
      RETURN                                                          VAR2 240
      END                                                             VAR2 250

      SUBROUTINE VAR2(SX,SX2,M1,M2,NBAT,NPAR)                         VAR2  10
C    NBAT IS THE NO. OF INDEPENDENT BATCHES                           VAR2  20
C    NPAR IS THE TOTAL NUMBER OF PARTICLES PROCESSED                  VAR2  30
C    IT IS ASSUMED THAT THE SUMSQ ARRAY HAS ACCUMULATED THE NUMBER OF PARTICLES
C      TIMES THE SQUARE OF THE BATCH AVERAGE (THIS IS OBTAINED BY DIVIDING    50
C      THE SQUARED BATCH SUM BY THE NUMBER OF PARTICLES STARTING THE BATCH)   60
      REAL * 8 SX, SX2, AN, DUM, BN, CN
      DIMENSION SX(M1,M2),SX2(M1,M2)                                  VAR2  70
      IF (NBAT-1) 5,5,15                                              VAR2  80
C**** WILSON   APRIL 20,1989 ****                                     MOR 1150
 5    AN = NPAR                                                       VAR2 130
      DO 29 I=1,M1                                                    VAR2 160
      DO 29 J=1,M2                                                    VAR2 170
      IF (SX(I,J)) 24,19,24                                           VAR2 180
 19   SX2(I,J) = 0.0                                                  VAR2 190
      GO TO 29                                                        VAR2 200
C24   WRITE(6,651)  SX(I,J),SX2(I,J),AN

C651  FORMAT(1H ,3(2X,1PE10.3))
 24   DUM       = SX2(I,J)/AN - (SX(I,J)/AN)**2                       VAR2 210
      SX2(I,J) = DSQRT(DABS(DUM)/AN)/(SX(I,J)/AN)                     VAR2 220
C     WRITE(6,651)  SX(I,J),SX2(I,J),DUM
 29   CONTINUE                                                        VAR2 230
C**** WILSON   APRIL 20,1989 ****                                     MOR 1150
      RETURN                                                          VAR2 120
 15   AN = NPAR                                                       VAR2 130
      BN = NBAT                                                       VAR2 140
      CN = DSQRT(BN - 1.)                                             VAR2 150
      DO 30 I=1,M1                                                    VAR2 160
      DO 30 J=1,M2                                                    VAR2 170
      IF (SX(I,J)) 25,20,25                                           VAR2 180
 20   SX2(I,J) = 0.0                                                  VAR2 190
      GO TO 30                                                        VAR2 200
```

```
 25   DUM       = SX2(I,J)*AN - SX(I,J)**2                             VAR2 210
      SX2 (I,J) = DSQRT(DABS(DUM      ))/(SX(I,J)*CN)                  VAR2 220
 30   CONTINUE                                                         VAR2 230
      RETURN                                                           VAR2 240
      END                                                              VAR2 250

      SUBROUTINE VAR3(SX,SX2,M1,M2,M3,NBAT,NPART)                      VAR3  10
C     NBAT IS THE NO. OF INDEPENDENT BATCHES                           VAR3  20
C     NPART IS THE TOTAL NUMBER OF PARTICLES PROCESSED                 VAR3  30
C     IT IS ASSUMED THAT THE SUMSQ ARRAY HAS ACCUMULATED THE NUMBER OF PARTICLES
C       TIMES THE SQUARE OF THE BATCH AVERAGE (THIS IS OBTAINED BY DIVIDING   50
C       THE SQUARED BATCH SUM BY THE NUMBER OF PARTICLES STARTING THE BATCH)  60
      REAL * 8 SX, SX2,  DUM
      DIMENSION SX(M1,M2,M3), SX2(M1,M2,M3)                            VAR3  70
      CALL ERRSET( 208,300,-1,1,1,208)
      IF (NBAT-1) 5,5,15                                               VAR3  80
  5   DO 10 I=1,M1                                                     VAR3  90
      DO 10 J=1,M2                                                     VAR3 100
      DO 10 K=1,M3                                                     VAR3 110
      IF (SX2(I,J,K)) 24,19,24                                         VAR3 110
 19   SX2(I,J,K) = 0.0                                                 VAR3 120
      GO TO 10
 24   DUM       = SX2(I,J,K)/AN - (SX(I,J,K)/AN)**2                    VAR2 210
      SX2(I,J,K) = DSQRT(DABS(DUM)/AN)/(SX(I,J,K)/AN)                  VAR2 220
 10   CONTINUE                                                         VAR3 130
      RETURN                                                           VAR3 130
 15   DO 30 I=1,M1                                                     VAR3 140
      DO 30 J=1,M2                                                     VAR3 150
      DO 30 K=1,M3                                                     VAR3 160
      IF (SX(I,J,K)) 25,20,25                                          VAR3 170
 20   SX2(I,J,K) = 0.0                                                 VAR3 180
      GO TO 30                                                         VAR3 190
 25   SX2(I,J,K) = (SX2(I,J,K)/NPART-(SX(I,J,K)/NPART)**2)/(NBAT-1.)   VAR3 200
      IF (SX2(I,J,K).LT.1.E-70) GO TO 30
      SX2(I,J,K) = DSQRT(DABS(SX2(I,J,K)))/SX(I,J,K)*NPART             VAR3 210
 30   CONTINUE                                                         VAR3 220
      CALL ERRSET( 208, 10,-1,1,1,208)
      RETURN                                                           VAR3 230
      END                                                              VAR3 240

      SUBROUTINE VAR4(SX,SX2,SX3,SX4,SX5,SX6,                          VAR2  10
     *M1,M2,NBAT,NPART,IFL)
C     NBAT IS THE NO. OF INDEPENDENT BATCHES                           VAR2  20
C     NPART IS THE TOTAL NUMBER OF PARTICLES PROCESSED                 VAR2  30
C     IT IS ASSUMED THAT THE SUMSQ ARRAY HAS ACCUMULATED THE NUMBER OF PARTICLES
C       TIMES THE SQUARE OF THE BATCH AVERAGE (THIS IS OBTAINED BY DIVIDING   50
C       THE SQUARED BATCH SUM BY THE NUMBER OF PARTICLES STARTING THE BATCH)  60
      REAL * 8 AN, X1, X2, X3, X4, DUM2, DUM4, DUM42
      REAL * 8 SX, SX2, SX3, SX4, SX5, SX6
      REAL * 8 BETA, BETAJ, S2, S4, AJ, V2S2, V2SL
      DIMENSION SX(M1,M2),SX2(M1,M2),SX3(M1,M2),SX4(M1,M2),SX5(M1,M2)  VAR2  70
     *,SX6(M1,M2)
```

```
C**** WILSON   APRIL 20,1989 ****                                    NOR 1150
      AN = NPART                                                     VAR2 130
      AJ = NPART/51.
      DO 29 I=1,N1                                                   VAR2 160
      DO 29 J=1,N2                                                   VAR2 170
      IF (SX(I,J)) 24,19,24                                          VAR2 180
 19   SX2(I,J) = 0.0                                                 VAR2 190
      SX5(I,J) = 0.0
      GO TO 29                                                       VAR2 200
 24   X1        = (1./AN)*SX(I,J)
      X2        = (1./AN)*SX2(I,J)
      X3        = (1./AN)*SX3(I,J)
      X4        = (1./AN)*SX4(I,J)
      S2        = X2 - X1**2                                         VAR2 210
C     S4        = X4-4.*X1*X3+8.*X1**2*X2-4.*X1**4-X2**2
      S4        = X4-4.*X1*X3+6.*X1**2*X2-3.*X1**4
      BETA      = S4/S2**2
      BETAJ     = BETA/AJ + 3.*(AJ-1.)/AJ
      V2S2      = 1./51.*(BETAJ -  48./50.)
C     IF(IFL .GT.0) GO TO 51
      V2SL      = (1./AN)*(BETA-(AN-3.)/(AN-1))
      SX5 (I,J) = DSQRT(DABS(V2S2/4.))
C51   SX2 (I,J) = DSQRT(DABS(  S2)/AN)/(SX(I,J)/AN)                  VAR2 220
      SX2 (I,J) = DSQRT(DABS(  S2)/AN)/(SX(I,J)/AN)                  VAR2 220
      SX6 (I,J) = DSQRT(DABS(V2SL/4.))
 29   CONTINUE                                                       VAR2 230
      RETURN                                                         VAR2 240
      END                                                            VAR2 250
```

Appendix D

Derivation of Some Equations of Chapter 2

The derivations in this appendix are found in Hansen, Hurwitz, and Madow [7].

D.1 Derivation of Equation 2.38

Equation 2.38 in Section 2.4 is given by

$$V_{S^2}^2 = \frac{1}{N}\left(\beta - \frac{N-3}{N-1}\right), \tag{D.1}$$

where

$$\beta = \frac{\mu_4}{(\sigma^2)^2}, \tag{D.2}$$

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2, \tag{D.3}$$

and

$$\mu_4 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^4. \tag{D.4}$$

Proof:

The relative variance of the variance is given by

$$V_{S^2}^2 = \frac{\sigma_{\sigma^2}^2}{(\sigma^2)^2}. \tag{D.5}$$

By definition

$$V_{S^2}^2 = \frac{E(S^2 - ES^2)^2}{(ES^2)^2} \doteq \frac{E(S^2 - \sigma^2)^2}{(\sigma^2)^2} = \frac{ES^4 - \sigma^4}{\sigma^4}. \tag{D.6}$$

To evaluate ES^4 the following transformation is used:

$$x_i - \mu = z_i \tag{D.7}$$

$$\bar{x} - \mu = \bar{z}, \tag{D.8}$$

which gives

$$ES^4 = E\left[\frac{\sum^N (z_i - \bar{z})^2}{N-1}\right]^2, \tag{D.9}$$

$$ES^4 = \frac{1}{(N-1)^2} E\left[(\sum^N (z_i^2))^2 - 2N\bar{z}^2 \sum^N z_i^2 + N^2 \bar{z}^4\right], \tag{D.10}$$

where

$$\left(\sum^N (z_i^2)\right)^2 = \sum^N z_i^4 + \sum_{i \neq j}^N z_i^2 z_j^2, \tag{D.11}$$

$$\bar{z}^2 \sum^N z_i^2 = \frac{1}{N^2}\left[\sum^N z_i^4 + 2\sum_{i \neq j}^N z_i^2 z_j + \sum_{i \neq j}^N z_i^2 z_j^2 + \sum_{i \neq j \neq k}^N z_i^2 z_j z_k\right], \tag{D.12}$$

and

$$\bar{z}^4 = \frac{1}{N^4}\left[\sum^N z_i^4 + 4\sum_{i \neq j}^N z_i^2 z_j + 3\sum_{i \neq j}^N z_i^2 z_j^2 + 6\sum_{i \neq j \neq k}^N z_i^2 z_j z_k + \sum_{i \neq j \neq k \neq m}^N z_i^2 z_j z_k z_m\right]. \tag{D.13}$$

For sampling with replacement, independent samples, and using the fact that $Ez_j = 0$, when $i \neq j \neq k \neq m$ it follows that

$$Ez_i^2 z_j^2 = Ez_i^2 Ez_j^2, \tag{D.14}$$

$$Ez_i^2 z_j = Ez_i^2 Ez_j = 0, \tag{D.15}$$

$$Ez_i^2 z_j z_k = Ez_i^2 Ez_j Ez_k = 0, \tag{D.16}$$

and

$$Ez_i^2 z_j z_k z_m = Ez_i^2 Ez_j Ez_k Ez_m = 0. \tag{D.17}$$

Therefore

$$E\left(\sum^N (z_i^2)\right)^2 = N\mu_4 + N(N-1)\sigma^4, \tag{D.18}$$

$$E\bar{z}^2\sum^{N} z_i^2 = \frac{1}{N^2}\left[N\mu_4 + N(N-1)\sigma^4\right], \tag{D.19}$$

and

$$E\bar{z}^4 = \frac{1}{N^4}\left[N\mu_4 + 3N(N-1)\sigma^4\right]. \tag{D.20}$$

Substituting equations D.18, D.19, and D.20 in Equation D.6, subtracting and dividing by σ^4

$$V^2_{S^2} = \frac{1}{N}\left(\beta - \frac{N-3}{N-1}\right). \tag{D.21}$$

D.2 Derivation of Equations 2.46 and 2.47

For I random groups of J elements

$$V^2_{S^2} = \frac{1}{I}\left(\beta_J - \frac{I-3}{I-1}\right), \tag{D.22}$$

where

$$\beta_J = \frac{\beta}{J} + 3\frac{J-1}{J}. \tag{D.23}$$

Proof:

In Chapter 2 it was shown that

$$S_b^2 = \frac{J}{I-1}\sum_{i=1}^{I}(\bar{x}_i - \bar{x})^2. \tag{D.24}$$

The relative variance of S_b^2 is also given by

$$V^2_{S_b^2} = \frac{\sigma^2_{\sigma^2}}{(\sigma^2)^2}. \tag{D.25}$$

Similar to Equation D.6

$$V^2_{S_b^2} = \frac{ES_b^4 - \sigma^4}{\sigma^4}. \tag{D.26}$$

Using the following transformation:

$$\bar{x}_i - \mu = \bar{z}_i \tag{D.27}$$

$$\bar{x} - \mu = \bar{z}, \tag{D.28}$$

Equation D.24 can be written as

$$S_b^2 = \frac{J}{I-1}\sum_{i=1}^{I}(\bar{z}_i - \bar{z})^2, \tag{D.29}$$

which gives

$$ES_b^4 = \frac{J^2}{(I-1)^2}E\left[\sum_{i}^{I}(\bar{z}_i - \bar{z})^2\right]^2, \tag{D.30}$$

or

$$ES_b^4 = \frac{J^2}{(I-1)^2}E\left[\left(\sum_{i}^{I}(\bar{z}_i^2)\right)^2 - 2I\bar{z}^2\sum_{i}^{I}\bar{z}_i^2 + I^2\bar{z}^4\right]. \tag{D.31}$$

The first term of Equation D.31 is given by

$$E\left(\sum_{i}^{I}\bar{z}_i^2\right)^2 = E\sum_{i}^{I}\bar{z}_i^4 + E\sum_{i\neq h}^{I}\bar{z}_i^2\bar{z}_h^2, \tag{D.32}$$

$$E\left(\sum_{i}^{I}\bar{z}_i^2\right)^2 = E\sum_{i}^{I}\left(\frac{\sum_j^J z_{ij}}{J}\right)^4 + E\sum_{i\neq h}^{I}\left(\frac{\sum_j^J z_{ij}}{J}\right)^2\left(\frac{\sum_j^J z_{ih}}{J}\right)^2, \tag{D.33}$$

which gives

$$E\left(\sum_{i}^{I}\bar{z}_i^2\right)^2 = I\left[\frac{\mu_4}{J^3} + \frac{3(J-1)\sigma^4}{J^3}\right] + I(I-1)\frac{\sigma^4}{J^2}. \tag{D.34}$$

The second term in Equation D.31 is given by

$$E\left(2I\bar{z}^2\sum_{i}^{I}\bar{z}_i^2\right) = 2IE\left[\left(\frac{\sum_i^I \bar{z}_i}{I}\right)^2\sum_{i}^{I}\bar{z}_i^2\right] = \frac{2}{I}E\left(\sum_{i}^{I}\bar{z}_i^2\right), \tag{D.35}$$

which becomes

$$E\left(2I\bar{z}^2\sum_{i}^{I}\bar{z}_i^2\right) = 2\left[\frac{\mu_4}{J^3} + \frac{3(J-1)\sigma^4}{J^3}\right] + 2(I-1)\frac{\sigma^4}{J^2}. \tag{D.36}$$

The third term in Equation D.31 becomes

$$EI^2\bar{z}^4 = \frac{1}{I^2}E\left(\sum_{i}^{I}\bar{z}_i\right)^4 = \frac{1}{I^2}E\left(\sum_{i}^{I}\bar{z}_i^4 + 3\sum_{i\neq h}^{I}\bar{z}_i^2\bar{z}_h^2\right), \tag{D.37}$$

which gives

$$EI^2\bar{z}^4 = \frac{1}{I^2}\left\{I\left[\frac{\mu_4}{J^3} + \frac{3(J-1)\sigma^4}{J^3}\right] + 3I(I-1)\frac{\sigma^4}{J^2}\right\}. \tag{D.38}$$

Substituting equations D.34, D.36, and D.38 in Equation D.31 Es_b^4 will be given by

$$Es_b^4 = \frac{\mu_4}{IJ} + \frac{3(J-1)}{IJ}\sigma^4 + \frac{I^2-2I+3}{(I-1)I}\sigma^4. \tag{D.39}$$

Substituting Equation D.39 in Equation D.26

$$V_{S_b^2}^2 = \frac{1}{I}\left[\frac{1}{J}\frac{\mu_4}{\sigma^4} + \frac{3(J-1)}{J} - \frac{I-3}{I-1}\right], \tag{D.40}$$

$$V_{S_b^2}^2 = \frac{1}{I}\left[\frac{\beta}{J} + \frac{3(J-1)}{J} - \frac{I-3}{I-1}\right], \tag{D.41}$$

or finally

$$V_{S_b^2}^2 = \frac{1}{I}\left(\beta_J - \frac{I-3}{I-1}\right). \tag{D.42}$$

D.3 Derivation of Equation 2.48

In terms of the relative variance Equation 2.48 in Section 2.4 can be written as

$$V_S^2 = \frac{V_{S^2}^2}{4}. \tag{D.43}$$

Proof:

Since

$$E(S-\sigma)^2 = ES^2 - 2\sigma ES + \sigma^2 = 2\sigma(\sigma - ES), \tag{D.44}$$

then

$$V_S^2 = 2\left(\frac{\sigma - ES}{\sigma}\right). \tag{D.45}$$

Using the expansion

$$\frac{S-\sigma}{\sigma} = \frac{(S^2-\sigma^2)}{2\sigma^2} - \frac{(S^2-\sigma^2)^2}{8\sigma^4} + \frac{(S^2-\sigma^2)^3}{16\sigma^6} - \dots \tag{D.46}$$

it follows that

$$\frac{\sigma - ES}{\sigma} = \frac{1}{8\sigma^4}E(S^2-\sigma^2)^2 - \frac{1}{16\sigma^6}E(S^2-\sigma^2)^3 + \dots \tag{D.47}$$

Since

$$\frac{V_S^2}{2} = \frac{\sigma - ES}{\sigma}, \tag{D.48}$$

Equation D.47 becomes

$$\frac{V_S^2}{2} = \frac{V_{S^2}^2}{8} - \frac{1}{16\sigma^6} E(S^2 - \sigma^2)^3 + \ldots \tag{D.49}$$

Finally for sufficiently large N

$$V_S^2 = \frac{V_{S^2}^2}{4}. \tag{D.50}$$

VITA

Wilson José Vieira was born in Anápolis, Goiás, Brazil on November 7, 1955. In 1979 he received his B.S. degree in Civil Engineering from the Universidade Federal de Uberlândia and in 1982 he received his M.S. degree in Nuclear Engineering from the Universidade de São Paulo. He started to work in 1982 for the Instituto de Pesquisas Energéticas e Nucleares in São Paulo in radiation shielding calculations. In 1986 he started his work toward the Ph.D. degree in Nuclear Engineering at the University of Tennessee, Knoxville.