HEDL-SA-1475 FP
CONF-780722--5

ANALYSIS AND EXTENSION OF THE SAND-II CODE IN DAMAGE FUNCTION UNFOLDING APPLICATIONS

G. L. Guthrie and R. L. Simons

Westinghouse Hanford Company

Paper to be presented at the ASTM 9th International Symposium on the Effects of Radiation on Structural Materials, Richland, Washington, July 10, 1978.

ANALYSIS AND EXTENSION OF THE SAND-II CODE IN DAMAGE FUNCTION UNFOLDING APPLICATIONS

G. L. Guthrie and R. L. Simons

Westinghouse Hanford Company
Richland, Washington 99352

Abstract

A way has been found to include a-priori information about the expected energy dependence of the damage function (together with uncertainties) as part of the data input of the SAND-II code in damage function unfolding procedures. Damage function analysis is useful when it is necessary to predict property change in a spectrum that is different from the spectra which were used to develop information about the degradation characteristics of the material. If the damage is a non-linear function of fluence, it is sometimes necessary to define an intermediate degradation parameter which is linear in the fluence. The integral equations can then be replaced by a set of linear equations with finite widths for the energy groups and a fixed number of groups. The inclusion of the a-priori information about the damage function causes the linear equations set to be overdetermined. If this overdetermined set is regarded as a least squares problem in log-normal statistics, then a solution can be obtained by a method of steepest descent. The steepest descent solution-algorithm is similar but not identical to the basic algorithm of the SAND-II code. In an overdetermined case, the basic SAND-II algorithm is shown to iterate towards a solution which is identical to the log-normal least squares solution.

ANALYSIS AND EXTENSION OF THE SAND-II CODE IN DAMAGE FUNCTION UNFOLDING APPLICATIONS

G. L. Guthrie and R. L. Simons

INTRODUCTION

The purpose of this work is to obtain an understanding of the theoretical mathematical basis of the SAND-II code and to extend the SAND -II code to the case where a-priori information on the damage function is available for use as input data. This will provide a well understood and easily usable method of analyzing data in an unfolding problem. This has application to the unfolding of damage functions for use in extrapolating Light Water Reactor Pressure Vessel Surveillance information to provide mechanical property degradation predictions at the 1/4 T position of the PV wall. The approach used in analyzing and extending the SAND-II algorithm is the following: First, the basic equations are derived for an algorithm which solves a log-normal least-squares problem of the type encountered in unfolding a damage function from property degradation observations obtained in a variety of spectra. Then minor differences are identified between such an algorithm and the one which is the basis of SAND-II. A method is given for extending or modifying SAND-II so as to treat a-priori information about the energy dependence of the damage function in a way that gives such information the same standing as weighted measurement results, rather than regarding the a-priori information as being simply a starting point in an iteration procedure. This extension of the method does not greatly increase the required number of computer operations in running the code.

THEORY AND DISCUSSION

In the usual unfolding problem, N samples have been damaged in N different neutron spectra. The spectra have each been divided into P separate energy groups. The symbol G_j is used for the coefficient of the i^{th} sample for damage by the neutron flux in the j^{th} energy group in the i^{th} spectrum. Then the calculated damage D_{ic} of the i^{th} sample is given by

$$D_{ic} = \sum_{j=1}^{P} G_j \phi_{ji} \qquad (1)$$

of SAND-II (Reference AFWL-TR-67-41, Volume 1, Page 14, Equation 31), it is found that in the basic SAND-II algorithm, the changes in $\log(G_j)$ are given by

$$\delta\left(\log(G_j)\right) = -F_j \Big/ \left(2 \sum_{i=1}^{N} \left(\frac{G_j \phi_{ji}}{D_{ic}}\right)\right) \tag{4}$$

if the SAND-II quantities are expressed in terms of the symbols used in the present report.

In attempting to derive the algorithm of SAND-II, then, we have calculated $\partial H/\partial (\log G_j)$ and found that we are missing a factor K_j^2 in the demoninator, where K_j^2 is given by

$$K_j^2 = 2 \sum_{i=1}^{N} \frac{G_j \phi_{ji}}{D_{ic}} \tag{5}$$

The reason for calling this factor K_j^2 instead of K_j will be apparent in the derivation that follows. The choice makes the subsequent discussion and manipulation easier.

Since the factor K_j is a function of j, the two vectors given by Equation (3) and Equation (4) are not parallel. The algorithm implied by Equation (3) and the basic algorithm of SAND-II both iterate towards lower values of H, but along slightly different paths.

The implied algorithm associated with Equation (3) constitutes a method of iterating along a path of steepest descent in $\log(G)$ space. The discrepancy between the algorithm of Equation (3) and the SAND-II algorithm of Equation (4) disappears if we replace $\partial H/\partial (\log G_j)$ by $\partial H/\partial (K_j \log G_j)$ in Equation (3), where K_j is to be considered a constant during the differentiation. Then the j^{th} component of the gradient is smaller by a factor of K_j and the actual step in $\log(G_j)$ space is smaller once again by a factor of K_j when we convert from changes in $K_j \log(G_j)$ to changes in $\log(G_j)$, thus giving the required factor of K_j^2 [from Equation (5)] in the denominator of the expression of Equation (4).

Thus the SAND-II algorithm is equivalent to a log-normal least-squares method where, at each step of the iteration, K_j is regarded as a constant and a path of steepest descent is used to proceed towards a minimum value of H in a space where the coordinate axes are proportional to $K_j \log(G_j)$.

The basic SAND-II algorithm does not reach a stage in the iteration giving zero length correction step in the G_j until all the components in Equation (4) are zero, and that can happen only when all the F_j are zero, as given by Equation (3). Thus the algorithm of Equation (3) and the basic SAND-II algorithm both iterate towards the same values of G_j which minimize the discrepancies between measured and calculated damage in a log-normal least-squares sense.

In the case of an overdetermined set of linear relations given by Equation (1), the algorithm of Equation (3) and the basic SAND-II algorithm give identical answers. The overdetermination can come from having more spectra (and associated damage experiments) than there are energy groups, or can be partly due to the restriction against use of negative values for damage coefficients and group flux levels. In either case, the solution of the log-normal least squares problem is unique and will be approached as a limiting value by the basic algorithm of SAND-II.

In the case of an underdetermined problem the answers arrived at by SAND-II and the algorithm of Equation (3) will differ slightly. In principle, the occurrence of underdetermined cases can be avoided in the use of SAND-II using the method described below.

It is possible to regard the a-priori information about the damage function as being equivalent to a set of P damage measurements conducted in P separate conceptual spectra. In this scheme, the r^{th} conceptual spectrum is considered to have unit flux in the r^{th} energy group and zero flux in all other energy groups. Thus the a-priori information of the damage function in the r^{th} group is treated as being a "measured" value with an appropriate value for the uncertainty or weight.

This concept provides P extra damage readings and guarantees that the solution of the problem of Equation (1) is not underdetermined.

To save computing time, the SAND-II procedures can be programmed to sum over the real spectra in steps of the type implied by the summations in Equation (4), and then jump to the single conceptual spectrum which contributes a nonzero addition to the sum, for each value of j in Equation (4).

Explicitly then, for the case with equally weighted input data, the form of the algorithm for the iterative adjustment in log G_j is as follows:

$$\delta (\log G_j) = \frac{-\log\left(\frac{G_j^k}{G_j^o}\right) + \sum_{i=1}^{n} \left[W_{ij} \log (R_i^k)\right]}{1 + \sum_{i=1}^{n} W_{ij}} \qquad (6)$$

where $\delta \log G_j$ is the increment in $\log G_j$ between iterations in going from the K^{th} to $K+1^{th}$ iteration, $W_{ij_k} = G_j^k \phi_{ji} / D_{ic}^k$, D_{ic}^k is the value of D_{ic} on the K^{th} iteration, R_i^k is the ratio D_{ic}/D_{im}, G_j^o is the a-priori damage coefficient in the j^{th} energy group and G_j^k is the damage coefficient for the j^{th} group after K iterations. The term $\delta (\log G_j)$ is to be identified with C_j in the usual SAND-II terminology. In actual practice, Equation (6) must be modified to include weights for both the real and conceptual damage experiments.

CONCLUSION

It has been shown that the basic SAND-II algorithm solves a least squares type problem in a log-normal statistical system for the case of an overdetermined set of equations. A method has been conceived for including the a-priori information about the energy dependence of the damage function and its associated uncertainties as part of the input data. This will result in a very small increase in computing time, while providing a basis for overdetermining the set of equations.