A MULTIGROUP, MULTIREGION, ONE-SPACE DIMENSIONAL PROGRAM USING NEUTRON DIFFUSION THEORY

DMM

ASAE-4

DISCLAIMER

A MULTIGROUP, MULTIREGION, ONE-SPACE DIMENSIONAL PROGRAM USING NEUTRON DIFFUSION THEORY

DMM

ASAE-4

Report by: Joel Franklin
and
Edward J. Leshan

Approved by: _____
David P. Herron, Manager
Reactor Engineering Department

December 1956

ATOMIC ENERGY DIVISION
American-Standard · Redwood City, California

TABLE OF CONTENTS

IV

ACKNOWLEDGEMENTS

This program is the result of a cooperative effort by Lockheed Aircraft Corporation, Missile Systems Division, represented by Jack Rosengren, Richard Friedman, Charles Humphreys, Ramon Moore, and R. J. Dickson; General Electric Company, Atomic Power Equipment Division, represented by Richard Stark and George Keierns; Electro Data Division of Burroughs Corporation, represented by Joel Franklin; and American-Standard, Atomic Energy Division, represented by Edward J. Leshan. The group was organized by Thomas Wilder, III, of Remington Rand Corporation.

The principle analysis done by Joel Franklin and Edward J. Leshan. Richard Castanias, Kenneth Rich, and Lou G. Kelly are responsible for the supervision and organization of the programming effort for the large machines. The Datatron version is being programmed by Joel Franklin and Foy Morrison. This report was edited by Carolee Walker.

The authors have drawn freely on previous work in developing this program. In particular, many ideas were obtained from the communications of G. J. Habetler, A. T. Biehl, H. Hurwitz, Jr., R. Ehrlich, and C. E. Leith.

It is a pleasure to acknowledge the practical advice of J. W. Webster and W. E. Newkirk.

I

ABSTRACT

DMM is a program using multigroup diffusion theory to calculate the reactivity or critical conditions and flux distribution of a multi-region reactor. As an essential part of this program are calculations of perpendicular buckling, fission produced xenon, and time variation due to production and depletion of isotopes. The adjoint fluxes may also be computed and the program includes the calculation of the nuclear constants from fairly simple input combined with a library of cross sections.

DMM is presently being coded for the Datatron, the Remington Rand 1103A, and the IBM 704.

Introduction

DMM is a program whose primary purpose is the calculation of the spatial and energy distribution of the flux of neutrons in a spherical, cylindrical or slab reactor and of the reactivity of this system. As auxiliary programs intimately associated with this, DMM include:

1) The calculation of multigroup cross sections using interpolated estimates of the flux within an energy group to weight what is essentially a point cross section table.

2) A burnout routine using the fluxes calculated by the main program to compute the consumption and production of reactor materials in finite time steps. After calculating the amount of an absorbing material to add in order to maintain the multiplication at unity, it computes new group constants and returns to the main program to compute new fluxes.

3) The calculation of the xenon distribution from the fluxes, recomputation of the group constants using this added material followed by return to the main program to calculate new fluxes and multiplication with the new group constants. The xenon distribution is computed for equilibrium or for a given time after shutdown.

4) Calculation of the adjoint fluxes; using the main flux program with altered nuclear constants.

5) Iteration on any one of several reactor parameters, such as, concentration of one of the elements in one of the regions, thickness of one of the regions, or density of one of the materials. This procedure obtains the value of the chosen parameter at which the reactivity is equal to a given value.

The Diffusion Equation

The principle calculation is the solution of a diffusion theory approximation for the multiplication:

$$\nabla \cdot (x,E)\, \nabla \phi\,(x,E) - \Sigma_t\,(x,E)\,\phi\,(x,E)$$

$$+ \sum_A \chi_{ir_A}\,(x,E) \int_E^{E_{max}} \Sigma_{in_A}\,(x,E')\,\nu_{in_A}\,(x,E')\,\phi\,(x,E')\,\frac{dE'}{\chi_{in_A}(x,E')}$$

$$+ \sum_{A} \int_{E}^{\min\left(E/\alpha_A, E_{max}\right)} \frac{\Sigma_{s_A}\, \phi\,(x,E')}{(1-\alpha_A)\,E'}\, dE' \qquad (1)$$

$$= -\frac{\chi_f\,(x,E)}{K} \int_{E_{min}}^{E_{max}} \Sigma_f\,(x,E')\,\nu_f\,(x,E')\,\phi\,(x,E')\,dE'$$

In addition to the diffusion approximation, several other assumptions have been made in writing Equation (1).

The second term covering inelastic scattering implies that, $\chi\,(x,E)$, the probability that a neutron absorbed at a higher energy E' will be emitted with an energy between E and $E+dE$ (if an inelastic collision takes place,) is dependant on E' only in that it is zero for $E > E'$. The divisor $\overline{\chi}_{in}\,(x,E)$ is given by

$$\overline{\chi}_{in}\,(x,E') = \int_{E_{min}}^{E} \chi_{in}\,(x,E'')\,dE'' \qquad (2)$$

and is included to normalise the truncated $\chi\,(x,E)$.

The third term covers elastic scattering and is exact except for the assumption that scattering is isotropic in the center of mass system. It is anticipated that in the rare cases where this is significant, non-isotropic components can be included by replacing the scatterer by several scatterers of different atomic weight.

The terms in Equation (1) that have not been described, may be defined as follows:

> x is the single spatial coordinate over which the integration of Equation (1) is performed in detail. In the case of a sphere, x is the distance from the center, in cylindrical calculations it is the perpendicular distance from the axis. Finally, in the case of slab geometry it is the single Cartesian coordinate being investigated.

E is the energy of the neutrons

$D(x,E)$ is the neutron diffusion coefficient

$\phi(x,E)$ is the neutron flux

$\Sigma_t(x,E)$ is the macroscopic total cross section

$\Sigma_{in_A}(x,E)$ is the macroscopic inelastic scattering cross section. An inelastic event is defined as a collision of the types; (n;n',), (n; 2n,), (n; 3n,) etc.

$\nu_{in_A}(x,E)$ is the number of neutrons emitted per inelastic collision with the A^{th} isotope.

α_A is the maximum fractional energy loss for a neutron colliding with the A^{th} isotope.

Σ_{s_A} is the macroscopic elastic scattering cross section for the A^{th} isotope.

$\chi_f(x,E)$ is the probability that a neutron emitted after a fission has energy in the range E to E+dE.

K is the multiplication.

Σ_f and ν_f are the macroscopic fission cross section and the number of neutrons emitted per fission respectively.

The Multigroup Method

If we define energy ranges E^i to E^{i+1} and integrate Equation (1) over these ranges we obtain a set of equations, one for each energy group.

$$-\nabla D^i(x)\nabla\phi^i(x)+T^i(x)\phi^i(x)=H^i(x);\ i=1,2,3\cdots I \tag{3}$$

where

$$H^i(x)=\sum_{j=1}^{i-1} T^{ij}(x)\phi^j(x)+\chi^i P(x) \tag{4}$$

and

$$P(x)=\frac{1}{K}\sum_{j=1}^{I} F^j(x)\phi^j(x) \tag{5}$$

The nuclear constants D^i, Σ^i, Σ^{ij}, χ^i, F^1, and the flux ϕ^i are related to the functions in Equation (1) by integration formulas as described in the section of this report on cross sections.

The Spatial Integration

We further define N+1 mesh points, 0, x_1, x_2 $\cdots x_N$ at which the functions of Equations (3), (4), and (5) are defined. Setting ρ=0 1 or 2 for planar, cylindrical and spherical geometries respectively. we obtain:

$$-a^i_n \phi^1_{n+1} + b^i_n \phi^i_n - c_n \phi^1_{n-1} = d^i_n \qquad (n\ 0, 1, 2 \cdots N) \tag{6}$$

$$a^i_n = \left(1 + \frac{\Delta x_n}{2x_n}\right)^\rho D^i_n \frac{\Delta x_{n-1/2}}{x_n}$$

$$b^i_n = a^i_n + c^i_n + \left(\Delta x_{n-1/2}\right)^2 T^i_n \qquad (n=1, 2 \cdots N-1) \tag{7}$$

$$c^i_n = \left(1 - \frac{\Delta x_{n-1}}{2x_n}\right)^\rho D^i_n \frac{\Delta x_{n-1/2}}{\Delta x_n}$$

$$d^i_n = \left(\Delta x_{n-1/2}\right)^2 H^1_n$$

where

$$\Delta x_n = x_{n+1} - x_n$$

$$\Delta x_{n-1/2} = 1/2 \left(x_{n+1} - x_{n-1}\right) \tag{8}$$

Boundary Conditions

At the boundaries of the mesh, x_0=0 and x_N, the equations for a^i_n, b^1_n, c^i_n, and d^i_n are modified to agree with the boundary conditions.

These are

$$-\left(1-B_o^{\,i}\right) D_o^{\,i} \frac{\phi^i - \phi_o^{\,1}}{\Delta x_o} + B_o^{\,i}\, \phi_o^{\,i} = 0$$

$$\left(1-B_N^{\,i}\right) D_N^{\,i} \frac{\phi_N^{\,i} - \phi_{N-1}^{\,i}}{\Delta x_{N-1}} + B_N^{\,i}\, \phi_N^{\,i} = 0 \qquad (9)$$

which correspond to the physical boundary conditions

$$-\left[1-B_o(E)\right] D(0,E)\, \nabla\phi(0,E) + B_o(E)\, \phi(0,E) = 0$$

$$\left[1-B_N(E)\right] D(X,E)\, \nabla\phi(X,E) + B_N(E)\, \phi(X,E) = 0 \qquad (10)$$

These equations permit solutions corresponding to flux zero at an extrapolated end point, symmetrical or non-symmetrical slabs, a control rod using the boundary conditions developed by Kushneriuk, or to an infinite, non-moderating reflector.

Equations (9) lead to the following equations for n=0 and N.

$$a_o^{\,i} = \left(1-B_o^{\,i}\right) D_o^{\,i}$$

$$b_o^{\,i} = a_o^{\,i} + B_o^{\,i}\, \Delta x_o \qquad (11)$$

$$c_o^{\,i} = 0$$

$$d_o^{\,i} = 0$$

$$a^i_N = 0$$

$$b^i_N = c^i_N + B^i_N \Delta x_{N-1}$$

$$c^i_N = \left(1 - B^i_N \quad D \cdot N \right. \tag{12}$$

$$d^i_N = 0$$

Numerical Procedure

Since the boundary conditions, Equations (11) and (12), apply at opposite ends of the mesh, the solution of Equations (6) and (7) subject to them requires some special device. We define p^i_n and q^i_n by

$$\phi^i_n = p^i_n \quad \phi^i_{n+1} + q^i_n \tag{13}$$

Substituting Equation (13) into Equation (6), gives as a solution

$$p^i_n = \frac{a^i_n}{b^i_n - c^i_n \; p^i_{n-1}}$$

$$q^i_n = \frac{d^i_n + c^i_n \; q^i_{n-1}}{b^i_n - c^i_n \; p^i_{n-1}} \tag{14}$$

$$p^i_{-1} = q^i_{-1} = 0$$

If the source H^i_n is known, it is possible to calculate the p^i_n and q^i_n beginning with n=0. Substituting these values in Equation (13) the ϕ^i_n may then be evaluated.

Since

$$T_n^{ij} = 0 \qquad i \leq j$$

H_n^1 can be calculated from P_n without any knowledge of the ϕ^i_n. Thus, if the P_n are known, the ϕ^i_n can be calculated, beginning with $i=1$ and proceeding, one group at a time, to $i=I$.

Having computed the ϕ^i_n, the P_n are computed by an equation similar to Equation (5).

$$P_n' = \sum_{j=1}^{I} F^j_n \, \phi^j_n \tag{16}$$

these are normalized by

$$P_n = \frac{1}{k} P_n'$$

$$\frac{1}{V} \sum_{n=1}^{N-1} P_n \, \Delta V_n = 1 \tag{17}$$

where

$$\Delta V_n = x_n^{\ell} \, \Delta x_n$$

$$V = \sum_{n=1}^{N} \Delta V_n \tag{18}$$

The K of Equation (17) is the reactivity computed for this iteration. Using the values of P_n computed in this way, it is possible to recalculate the ϕ^i_n and continue in this iterative fashion until the results show satisfactory convergence.

In order to accelerate the rate of convergence, one uses $\tilde{P}_n$ instead of P_n in Equation (4).

$$ {}^{t}\tilde{P}_n = \beta \left[{}^{t}P_n + \omega \left({}^{t}P_n - {}^{(t-1)}\tilde{P}_n \right) \right] $$

$$ \beta = \frac{V}{\sum_{n=1}^{N-1} \left[{}^{t}P_n + \omega \left({}^{t}P_n - {}^{(t-1)}\tilde{P}_n \right) \right] \Delta V_n} \qquad (19) $$

The presubscript, t, is the iteration number

The Nuclear Constants

The integration of Equation (1) leads to the following definitions for the functions appearing in Equations (3), (4), and (5).

$$ D^i \equiv \left[\int_{E_{i+1}}^{E_i} D(E)\, \phi(E)\, dE \right] \Big/ \left[\int_{E_{i+1}}^{E_i} \phi(E)\, dE \right] \qquad (20) $$

$$ T^i = \left[\int_{E_{i+1}}^{E_i} \left\{ \Sigma_t(E)\, \phi(E) - \sum_A \chi_{in_A}(E) \int_E^{E_i} \Sigma_{in_A}(E') \nu_{in_A}(E') \phi(E') \frac{dE'}{\chi_{in_A}(E')} \right. \right. $$

$$ \left. \left. - \sum_A \int_E^{\min\left(\frac{E}{\alpha_A}, E_i\right)} \frac{\Sigma_{s_A}(E')}{(1-\alpha_A)E'}\, \phi(E')\, dE' \right\} dE \right] \Big/ \left[\int_{E_{i+1}}^{E_i} \phi(E)\, dE \right] \qquad (21) $$

$$ -B^2 D^i $$

$$T^{ij} \equiv \left[\int_{E_{i+1}}^{E_i} \left\{ \sum_A \chi_{in_A}(E) \int_{E_{j+1}}^{E_j} \Sigma_{in_A}(E') \nu_{in_A}(E') \phi(E') \frac{dE'}{\chi_{in_A}(E')} \right. \right.$$

$$\left. \left. + \sum_A \int_{E_{j+1}}^{\mathrm{Min}(E/\alpha_A,\, E_j)} \frac{\Sigma_{s_A}(E')}{(1-\alpha_A)E'} \phi(E')\, dE' \right\} dE \left\{ \frac{E_j - E_{j+1}}{E_i - E_{i+1}} \right\} \right] \Bigg/ \left[\int_{E_{j+1}}^{E_j} \phi(E)\, dE \right] \tag{22}$$

$$\chi^i \equiv \left[\int_{E_{i+1}}^{E_i} \chi_f(E)\, dE \right] \Bigg/ \left[E_i - E_{i+1} \right] \tag{23}$$

$$F^j \equiv \left[\int_{E_{j+1}}^{E_j} \Sigma_f(E) \nu_f \phi(E)\, dE\, (E_j - E_{j+1}) \right] \Bigg/ \left[\int_{E_{j+1}}^{E_j} \phi(E)\, dE \right] \tag{24}$$

$$\phi^i \equiv \left[\int_{E_{i+1}}^{E_i} \phi(E)\, dE \right] \Bigg/ \left[E_i - E_{i+1} \right] \tag{25}$$

B^2 is the perpendicular buckling, an input constant.

The calculation of these quantities must use different equations since $\phi(E)$ is not known, though ϕ^i is, and since these equations imply a computation at each mesh point--an excessive amount of work. As a compromise, we use the function ${}^k\psi(E)$ in place of $\phi(E)$ in Equations (20) through (24). Equation (25) need not be performed at all. ${}^k\psi(E)$ is obtained from a set of ${}^i\phi_n$ by integrating the ${}^i\phi_n$ over regions of uniform composition and interpolating. Thus, if

$${}^k\phi^i \equiv \sum_{N_{k-1}}^{N_k} {}^i\phi_n\, \Delta V_n \tag{26}$$

where k indicates the k^{th} region,

$$k\psi(E) = k_{C_1^i} + k_{C_2^i}/E + k_{C_3^i} E \qquad E_{i+1} \leq E < E_i \tag{27}$$

The coefficients are defined by

$$k\phi^{i-1} = k_{C_1^i} + \frac{k_{C_2^i}}{E_{i-1/2}} + k_{C_3^i} \quad E_{i-1/2}$$

$$k\phi^{i} = k_{C_1^i} + \frac{k_{C_2^i}}{E_{i+1/2}} + k_{C_3^i} \quad E_{i+1/2} \tag{28}$$

$$k\phi^{i+1} = k_{C_1^i} + \frac{k_{C_2^i}}{E_{i+3/2}} + k_{C_3^i} \quad E_{i+3/2}$$

$$E_{i+1/2} = \frac{E_{i+1} + E_i}{2}$$

The cross sections used in the auxiliary averaging program are to be found in a library of cross sections on magnetic tape. In this library the values of the cross sections of each isotope are given at a relatively large number of energies, $E\ell_A$. The $E\ell_A$ in general are different for each isotope, being chosen as the most convenient ones to represent the behavior of that isotope's cross sections. The assumption involved is that the cross sections can be approximated by step-functions as shown in Figure 1, Page 11, if a sufficient number of points are used.

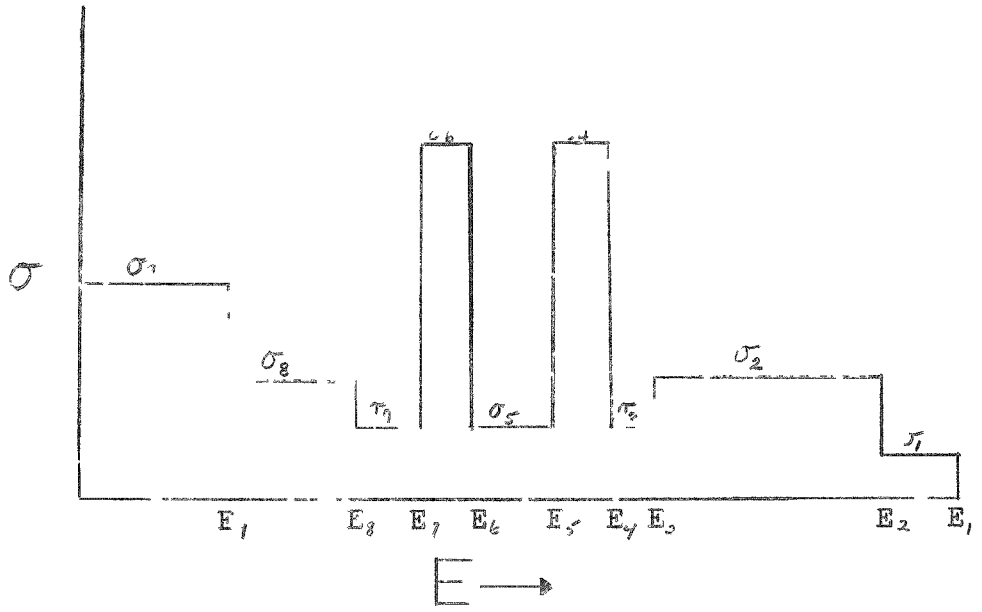

Figure 1. Cross Sections approximated by step-functions.

The information stored in the library for each isotope is the isotope identification, A, its atomic weight a_A, and α_A. For each energy value for each isotope there is also σ_t, σ_s, $\sigma_{in}\gamma_{in}$, $\sigma_f\gamma_f$, χ_{in}. At an intermediate stage in the cross section averaging program, before converting microscopic to macroscopic quantities, and before averaging over energy, there is an option of adding the combination of isotopes to the library as a new isotope.

The input to the cross section averaging program is, in addition to the library, the following data:

M	The material identification.
Set of A and N_A	The isotopes and relative number of atoms of each present in the mixture.
ϱ	The density of the mixture.
Set of E_n	The upper energy limits of the groups to be used in the main code.

Figure 2, Page 12, explains the method of combining the cross sections of the isotopes present. Here we define $\sigma\varrho_A$ as the cross section

for the A^{th} isotope at the ℓ^{th} energy value $E_{\ell A}$ at which cross sections for that isotope are specified.

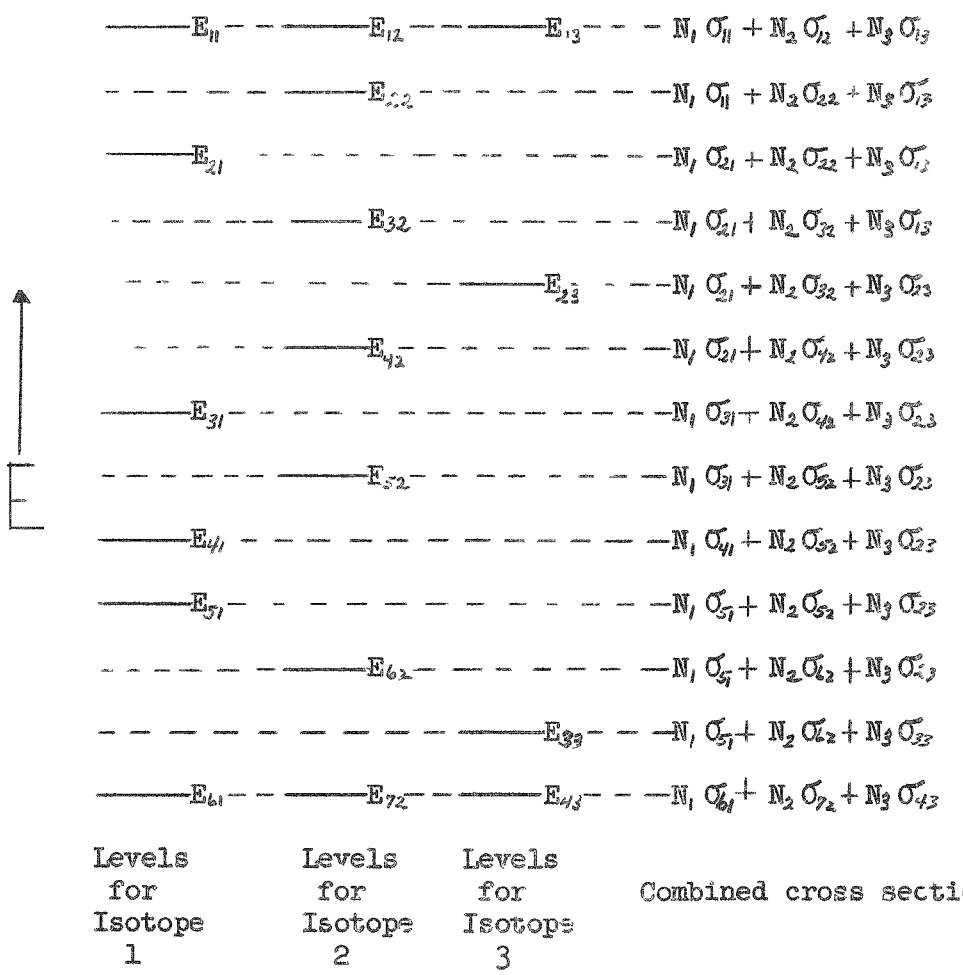

Figure 2. Method of combining the cross sections of the isotopes present.

In this way the cross sections are combined to get a set for a new "isotope" of atomic weight $N_1A_1 + N_2A_2 \cdots$. At this point, the new isotop with name equal to M can be added to the library. The cross sections are then averaged over energy according to the Equations (20) through (24). In these equations $D(E) = \frac{1}{3\Sigma_t}$.

To convert the results to macroscopic cross sections D^i is divided by C_N, and T^i, T^{ij}, and F^i are multiplied by C_N where

$$C_N = \frac{0.6023\,\rho}{\sum_A a_A N_A}$$

The $\phi^i{}_n$ used in the nuclear constant evaluation may be input to a problem or may be the result of a previous iteration or stage of the calculation. This cross section averaging routine may be transferred to at some stage of the calculation when some of the auxiliary programs have caused a change in concentration of some components. It is also possible to transfer control to this part of the program after a number of iterations when the computer feels that the flux weighting is sufficiently important and the spectrum of neutrons has changed appreciable from the initial guess.

If a problem is begun without a set of $\phi^i{}_n$ to use for flux weighting the cross sections, the program automatically replaces the ${}^k\psi(E)$ by $\frac{1}{E_i - E_{i+1}}$.

Self-Shielding

Usually this calculation will be used on a homogenized approximation to the actual reactor. For this reason, it is incorrect to mix the isotopes in a mixture according to their relative concentrations as described in the Nuclear Constants Section of this report. This difficulty can be solved by multiplying the concentrations by self-shielding factors which should be functions of energy and are the ratios of the flux densities in the regions of a lattice cell. Therefore, provision is made in the auxiliary cross section averaging program for use of these self-shielding factors. This is done by distinguishing, in the list of isotopes present in a mixture, between isotopes associated with each of three submixtures. For each of these submixtures a self-shielding factor is given for each energy range in the main program so that the input includes a set of f_{i1}, f_{i2}, and f_{i3}. The N_A in Figure 2, Page 12, are then multiplied by the appropriate f_{i1}, f_{i2}, or f_{i3} in adding-up cross sections for a mixture.

In order to minimize the work involved in input preparation, the program substitutes a value of unity for any self-shielding factor that has not been specified.

Xenon Addition

DMM contains an auxiliary program which can add xenon to the mixture initially present in a region containing fissionable material. This can only be done at a stage in the calculation when the P_n are available and the amount added is that corresponding to equilibrium or to a given time after complete shutdown.

The input required is

Δt - the time since shutdown in hours

($\Delta t = 0$ gives equilibrium xenon concentration)

I - the power density of the reactor in kw/cc

The calculation makes use of the following constants

$\mathscr{P}$ - fissions per kw sec

Y_{Xe} - xenon atoms produced per fission

Y_I - iodine atoms produced per fission

λ_{Xe} - probability per hour of decay of a xenon atom

λ_I - probability per hour of decay of an iodine atom

Then

$$k\mathfrak{N}_{Xe}(t) = k\mathfrak{N}^o_{Xe} e^{-\lambda_{Xe}\Delta t} + \mathfrak{N}^o_I \left(e^{-\lambda_I \Delta t} - e^{-\lambda_{Xe}\Delta t}\right) \qquad (30)$$

$$N^{\circ}_{Xe} = \frac{(\gamma_I + \gamma_{Y_2})}{k_{-A}\lambda_{X_e} + k\sigma_{X_e}} \; k_P$$

$$k_P = 1/2.5 \sum_{n=N_{k-1}}^{N_k} P_n \Delta V_n \qquad (31)$$

$$k\Sigma_{Xe} = L \int_{E_{min}}^{E_{max}} k\psi(E, \sigma_{a_{Xe}}(E)\, dE$$

The N_{Xe} obtained from Equation (30) is treated in exactly the same manner as the other N_A in the cross section averaging routine.* The use of $\sigma_{a_{Xe}}$ in Equation (31) implies that the cross section library contains an extra file for xenon.

Burnup

DMM contains an option to permit the calculation of the behavior of a reactor as a function of time as reactor materials are depleted and produced due to neutron absorption.

The input required is

η	-	the number of iterations to be done between each time step
$[N_A]$	-	the address of the concentration which is to be varied to keep k equal to $K_\circ$
$K_\circ$	-	the value of the reactivity which is to be maintained
δ	-	a guess as to the change in N_A necessary in each time step
ε	-	the tolerance for the deviation of K from $K_\circ$
T	-	the length of the time step to be used
L	-	the power level defined in the previous section

Note: *If $\Delta t=0$, $^{k}N_{Xe} = {}^{k}N_{Xe}^{\circ}$, the equilibrium xenon concentration. Also $^{k}\Gamma_{Xe}=0$ if $^{k}P=0$ so that this need be computed only for fissioning regions.

The calculation is based on the approximation that the neutron flux and the concentrations remain constant during each time step.

The cross section library contains with each isotope, a set of three or fewer isotope transformation cross sections, $\sigma_{A\to A'}$.

These are microscopic cross sections for the transformation of isotope A into isotope A'.

Macroscopic transformation cross sections, ${}^{k}\Sigma^{i}_{A\to A'}$, are calculated from the microscopic cross sections by the equation

$$ {}^{k}\Sigma^{i}_{A\to A'} = {}^{k}N_A \; {}^{k}C_N \left[\int_{E_{i+1}}^{E_i} \sigma_{A\to A'} \; {}^{k}\psi^{i}(E)\, dE \right] \Bigg/ \left[\int_{E_i}^{E_i} {}^{k}\psi^{i}(E)\, dE \right] \qquad (32) $$

The changes in the N_A are obtained by adding the δN_A to the old N_A.

$$ \delta {}^{k}N_A = \frac{\delta T \rho L}{{}^{k}C_N} \sum_{i=1}^{I} \left[\sum \left({}^{k}\Sigma^{i}_{A'\to A} - {}^{k}\Sigma^{i}_{A\to A'} \right) {}^{k}\phi^{i} \right] \left(E_i - E_{i+1} \right) \qquad (33) $$

Having computed the new compositions of the regions, it is necessary to adjust the concentration of isotope $\overline{A}$ in order to keep K equal to K_o. δ is added to ${}^{k}N_{\overline{A}}$ and the program transfers to the cross section averaging routine to compute new nuclear constants and thence to the main program. After $\mathcal{N}$ iterations the new value of K is compared with K_o. If $K<K_o$, $0.1\,\delta$ is added to ${}^{k}N_{\overline{A}}$. If $K>K_o$, 0.1 is subtracted from ${}^{k}N_{\overline{A}}$; in either case, the procedure beginning with the transfer to the cross section program is repeated. If we define K_i and K_{i+1} as the reactivities obtained in two successive trials.

$$ N_{\overline{A}_{i+1}} = N_{\overline{A}_i} + \left(N_{\overline{A}_i} - N_{\overline{A}_{i-1}} \right) \frac{K_o - K_i}{K_i - K_{i-1}} \qquad (34) $$

where N_{A_i} is the value of N_A used in the i^{th} trial. The procedure is repeated using Equation (34) to find the amount of control material until

$K_0 - K_i \leq \xi$. At this point all the N_A are printed out and the program transfers to the calculations of Equation (33) for the next time step.

The Adjoint Equation

At the computer's option, DMM may be used to calculate the adjoint fluxes. The equation adjoint to Equation (1) is

$$\nabla D(x,E)\nabla\phi^*(x,E) - \Sigma_t(x\ E)\phi^*(x,E)$$

$$+\sum_A \Sigma_{in_A}(x,E)\frac{\nu_{in_A}(x,E)}{\chi_{in_A}(x,E)}\int_{E_{in}}^{E}\chi_{in_A}(x,E')\phi^*(x,E')\,dE'$$

$$+\sum_A \frac{\Sigma_{s_A}(E)}{(1-\alpha_A)E}\int_{Max(E_{Min},\frac{\alpha}{A}E)}^{E}\phi^*(x,E')\,dE' \qquad (35)$$

$$=\frac{1}{K^*}\Sigma_f(x,E)\nu_f(x,E)\int_{E_{Min}}^{E_{Max}}\chi_f(x,E')\phi^*(x,E')\,dE'$$

The corresponding multigroup equations are almost identical in form to Equations (3), (4), and (5)

$$-\nabla D^{*i}(x)\nabla\phi^{*i}(x)+T^{*i}(x)\phi^{*i}(x)=H^{*i}(x) \qquad i=1,2,3\cdots I \qquad (36)$$

$$H^{*i}(x)=\sum_{j=i+1}^{I} T^{*ij}(x)\phi^{*j}(x)+\chi^{*i}P^*(x) \qquad (37)$$

$$P^{*}(x) = \frac{1}{K} \sum_{j=1}^{I} F^{*j}\, \phi^{*j}(x) \tag{38}$$

The only difference between these equations and Equations (3), (4), and (5) is that the sum in Equation (37) is over $j>i$ while in Equation (4) it is over $j<i$. The definitions of the nuclear constants, however, are quite different. The adjoint equations analogous to Equations (20) through (25) are

$$D^{*i} \equiv \left[\int_{E_{i+1}}^{E_i} D(E)\, \phi^{*}(E)\, dE \right] \Big/ \left[\int_{E_{i+1}}^{E_i} \phi^{*}(E)\, dE \right] \tag{39}$$

$$T^{*i} \equiv \left[\int_{E_{i+1}}^{E_i} \left\{ \Sigma_t(E)\, \phi(E) - \sum_{A}^{I} \Sigma_{in_A}(E) \frac{\gamma_{in_A}(E)}{\chi_{in_A}(E)} \int_{E_{i+1}}^{E} \chi_{in_A}(E')\, \phi^{*}(E')\, dE' \right. \right.$$

$$\left. \left. - \sum_{A}^{I} \frac{\Sigma_{s_A}(E)}{(1-\alpha_A)E} \int_{Max(E_{i+1}, \alpha E)}^{E} \phi^{*}(E')\, dE' \right\} dE \right] \Big/ \left[\int_{E_{i+1}}^{E_i} \phi^{*}(E)\, dE \right] \tag{40}$$

$$T^{*ij} \equiv \left[\int_{E_{i+1}}^{E} \left\{ \sum_{A}^{I} \frac{\Sigma_{in_A}(E)\, \gamma_{in_A}(E)}{\chi_{in_A}(E)} \int_{E_{j+1}}^{E_j} \chi_{in_A}(E')\, \phi^{*}(E')\, dE' \right. \right. \tag{41}$$

$$+\sum_{A}^{1}\frac{\Sigma_{s_A}(E)}{(1-\alpha_A)E}\left\{\int_{\mathrm{Max}(E_{j+1},\,E)}^{E_j}\phi^*(E')\,dE'\right\}\,dE\Bigg]\left[\frac{E_j-E_{j+1}}{E_i-E_{i+1}}\right]\Bigg/\left[\int_{E_{j+1}}^{E_j}\phi^*(E)\,dE\right] \tag{41}$$

$$\chi^{*i}\equiv\left[\int_{E_{i+1}}^{E_i}\Sigma_f(E)\,\nu_f(E)\,dE\right]\Bigg/\left[E_i-E_{i+1}\right] \tag{42}$$

$$F^*\equiv\left[\int_{E_{j+1}}^{E_j}\chi_f(E)\,\phi^*(E)\,dE\,\left(E_j-E_{j+1}\right)\right]\Bigg/\left[\int_{E_{j+1}}^{E_j}\phi^*(E)\,dE\right] \tag{43}$$

$$\phi^{*i}\equiv\left[\int_{E_{i+1}}^{E_i}\phi^*(E)\,dE\right]\Bigg/\left[E_i-E_{i+1}\right] \tag{44}$$

The integrations in Equations (39) to (43) are carried out using the functions ${}^{k}\psi^*(E)$ defined in a manner identical to the ${}^{k}\psi(E)$ of Equations (26), (27), and (28) except that adjoint fluxes are substituted for the ϕ^{i}_{n}.

Determination of Critical Conditions

By a procedure very similar to that described in the section of this report on burnout, DMM may be made to determine the conditions under which a reactor is critical. As imput, we specify the address of single number, $\mathcal{Z}$, which is to be varied, the number of iterations, $\mathcal{N}$, between variations, a first guess, δ, for the amount by which it is to be varied, K_o, the reactivity desired, and $\mathcal{E}$, the tolerance for the deviation of the final reactivity from K_o. The number to be varied might be, for example, the mesh spacing in any region, the concentration of one component in any region, the density of any region, the power level (since this effects

the equilibrium xenon concentration), or any other continuous variable of which the reactivity is a function. The equation governing the trial changes in $\mathcal{Z}$ is the same as Equation (34)

$$\mathcal{Z}_{i+1} = \mathcal{Z}_i + \left(\mathcal{Z}_i - \mathcal{Z}_{i-1}\right) \frac{K_o - K_i}{K_i - K_{i-1}} \tag{45}$$

Trials are made iteratively until $\left|k_o - k_i\right| \leq \mathcal{E}$.

Neutron Balance

DMM computes the various components of neutron transfer from and to each group and region and can, at the computer's option, print these out. These components are

1) The degradation from each group, i, in region k:

$$^{k}\mathcal{D}^{i} = \sum_{n=N_{k-1}+1}^{N_k} \sum_{j=i+1}^{I} k_T^{ji}\, \phi^{i}_{n}\, \Delta V_n \tag{46}$$

2) The degradation into each group, i, in region k:

$$^{k}\mathcal{S}^{i} = \sum_{n=N_{k-1}+1}^{N_k} \sum_{j=1}^{i-1} k_T^{ij}\, \phi^{j}_{n}\, \Delta V_n \tag{47}$$

3) The production due to fission in each group, i, in region k:

$$^{k}\mathcal{F}^{i} = \sum_{n=N_{k-1}+1}^{N_k} P_n\, \chi^{i}\, \Delta V_n \tag{48}$$

4) The loss due to neutron capture from each group, i, in region k:

$$ {}^{k}C^{i} = \sum_{n=N_{k-1}+1}^{N_k} {}^{k}T^{i}\, \phi^{i}_{n}\, \Delta V_n - {}^{h}\mathcal{D}^{i} - {}^{k}\mathcal{L}^{i}_{p} \qquad (49) $$

5) The loss due to leakage from each group, i, at the inner boundary of region k:

$$ {}^{k}\mathcal{L}^{i}_{-} = {}^{k}D^{i}\, \frac{\phi^{i}_{N_{k-1}+1} - \phi^{i}_{N_{k-1}}}{{}^{k}\Delta X} \qquad (50) $$

6) The loss due to leakage from each group, i, at the outer boundary of region k:

$$ {}^{k}\mathcal{L}^{i}_{+} = - {}^{k}D^{i}\, \frac{\phi^{i}_{N_k} - \phi^{i}_{N_{k-1}}}{{}^{k}\Delta X} \qquad (51) $$

7) The loss due to perpendicular buckling from each group, i, in region k:

$$ {}^{k}\mathcal{L}^{i}_{p} = \sum_{n=N_{k-1}+1}^{N_k} B^{2}\, {}^{k}D^{i}\, \phi^{i}_{n}\, \Delta V_n \qquad (52) $$

8) The flux in each group in region k:

$$k\phi^i = \sum_{n=N_{k-1}}^{N_k} \phi^i_n \, \Delta V_n \tag{53}$$

The program also sums these functions over i to give the total transfer in each region.

Various obvious equalities apply to these functions and sums, providing a check on the operation of the code.

Iteration on the Perpendicular Buckling

In finding the reactivity of a cylindrical or slab reactor by means of a one-space dimensional program, one is faced with the problem of accounting for leakage in the direction perpendicular to that being studied. This is usually done by specifying a perpendicular buckling, B^2, and adding B^2D to the total cross section. If the reactor is unreflected in the perpendicular direction the evaluation of B^2 is simple; otherwise it is necessary to determine a value of the reflector savings. If the leakage from the core in the perpendicular direction $\mathcal{L}_{cp}$ is known, the perpendicular buckling is

$$B^2 = \frac{\mathcal{L}_{cp}}{D\phi}$$

The method by which this can be accomplished in DMM is as follows: The input for two reciprocal versions of the reactor (normally a cylindrical and a slab version) are read in at one time, an initial guess being given for the perpendicular buckling in one of these. A calculation is then performed on the first version and the k_{eff}, $k\mathcal{L}_-^i$, and $k\mathcal{L}_+^i$ are recorded. If the regions from k_{c_1} to k_{c_2} have been specified as the core region, the program determines ${}_cB^{i\,2}$ from the equation

$${}_cB^i = \frac{k_{c_2}\mathcal{L}_+^i + k_{c_1}\mathcal{L}_-^i - k_{c_2}\mathcal{L}_-^i - k_{c_1}\mathcal{L}_+^i}{\sum_{k=k_{c_1}}^{k_{c_2}} k\phi^i \; k_D^i} \tag{54}$$

$^{o}\mathcal{L}_{+}^{i}$ is assumed to be zero and the boundary conditions require that $'\mathcal{L}_{-}^{i}$ be zero (when the reactor is symetric) so that if, as is normal, the core includes the central region, these two terms will not effect Equation (54). The values of $^{c}B^{i}$ obtained in this manner are then used in the reciprocal problem as the perpendicular buckling.

After performing the reciprocal calculation a set of $^{c}B^{i}$ are computed from its results in the same manner and used in the initial problem. This process is continued iteratively with η flux iterations performed on each problem between calculations of the $^{c}B^{i}$. When the two reactivities agree within a tolerance, ε, the process is terminated.

Note: This program uses a different value of the perpendicular buckling in each energy group and accomodations must be made for this in the calculation of the $^{k}T^{i}$.