

Library Information (Lucas) TRN AUS206090

AAEC/E539

AAEC/E539



AUSTRALIAN ATOMIC ENERGY COMMISSION  
RESEARCH ESTABLISHMENT

LUCAS HEIGHTS RESEARCH LABORATORIES

ANAUSN - A ONE-DIMENSIONAL MULTIGROUP SN TRANSPORT  
THEORY MODULE FOR THE AUS REACTOR NEUTRONICS SYSTEM

by

B.E. CLANCY

May 1982

ISBN 0 642 59744 8

AUSTRALIAN ATOMIC ENERGY COMMISSION  
RESEARCH ESTABLISHMENT

LUCAS HEIGHTS RESEARCH LABORATORIES



ANAUSN - A ONE-DIMENSIONAL MULTIGROUP SN TRANSPORT  
THEORY MODULE FOR THE AUS REACTOR NEUTRONICS SYSTEM

by

B.E. CLANCY

ABSTRACT

ANAUSN is a general purpose, one-dimensional discrete ordinate transport theory program which has access to AUS datapools. Fixed source, reactivity and a variety of criticality search calculations can be performed. The program can be operated as a module in the AUS scheme or as a stand-alone program.

National Library of Australia card number and ISBN 0 642 59744 8

The following descriptors have been selected from the INIS Thesaurus to describe the subject content of this report for information retrieval purposes. For further details please refer to IAEA-INIS-12 (INIS: Manual for Indexing) and IAEA-INIS-13 (INIS: Thesaurus) published in Vienna by the International Atomic Energy Agency.

A CODES; COMPUTER CALCULATIONS; DISCRETE ORDINATE METHOD; MULTIGROUP THEORY;  
NEUTRON TRANSPORT THEORY

## CONTENTS

1. INTRODUCTION	1
2. ZONING AND PROBLEM SPECIFICATION	2
3. INPUT	2
3.1 Data Block 1	3
3.2 Data Block 2	3
3.3 Data Block 3	5
3.4 Data Block 4	8
3.5 Data Block 5	12
3.6 Minimal Input	12
4. OUTPUT	13
4.1 AUS Datapools	13
4.2 Printed Output	13
5. CORE MEMORY REQUIREMENTS AND CODE STRUCTURE	15
6. REBALANCE PROCEDURES	16
7. INVOCATION OF ANAUSN	19
7.1 ANAUSN Catalogued Procedure	20
8. ACKNOWLEDGEMENTS	20
9. REFERENCES	20

## 1. INTRODUCTION

The ANAUSN code is an implementation of Carlson's discrete ordinate  $S_N$  method for the solution of multi-group transport problems in one-dimensional geometries - plane, cylindrical and spherical [Carlson 1963]. As a module in the AUS system [Robinson 1975], it serves two functions:

- (a) In large heterogeneous reactor systems, ANAUSN treats the single cell multiplication problem to produce tentative group fluxes which can be used to homogenise the cell and to provide the few group cross sections for use with multi-dimensional diffusion theory codes. The ANAUSN calculation produces a solution of an eigenvalue problem, with the eigenvalue being the cell fission multiplication factor  $k_{\infty}$ .
- (b) For systems which can be adequately represented in one dimension but for which the diffusion theory approximation is unacceptable, ANAUSN provides the necessary transport theory solution. The calculation may be the eigenvalue problem equivalent to that carried out for a cell calculation on a determination of the system fission multiplication factor  $k_{eff}$ . Optionally, the code may be asked to vary some system parameter and search for a value which makes the multiplication factor take some specified value - usually 1.0. A quite separate calculation is that for source problems, in which a fixed source (of neutrons) is being released in a sub-critical system and the determination of steady state group fluxes is required. In all cases, either the normal or adjoint transport equation can be solved.

The basic solution strategy is that laid down in the original  $S_N$  codes of Carlson [Carlson and Bell 1958; Carlson et al. 1960] and followed essentially unchanged in all  $S_N$  codes since that time. Consequently, it is inappropriate to rehearse the theory of the  $S_N$  treatment and the terminology and nomenclature in the Carlson [1963] account are retained, assuming that the reader is familiar with them.

The task of developing a new code was undertaken for two reasons: to provide efficient access to the AUS scheme datapools and to have the opportunity of implementing improvements in numerical techniques for producing converged solutions.

Development of the code was begun in 1972 and a first version was installed in the AUS scheme in 1973. A number of revisions have been made as experience with the code was accumulated. The version reported here was installed in the AUS scheme in September 1981. Except for the SCAN data reading subroutine of Bennett and Pollard [1967] and the VARRAY core memory allocation routine, a modification of the DARRAY routine of Cox [1971], the code is written entirely in FORTRAN IV.

## 2. ZONING AND PROBLEM SPECIFICATION

Specification of the material layout through user-supplied input data and most of the acceleration procedures depend on the concept of a material zone; this is simply a set of contiguous mesh intervals which are occupied by a single material. When defining the material layout, the user's data must assign each interval to a particular zone and then provide a material number to be associated with each of the zones.

Depending on the problem type, the number of mesh intervals and the number of material zones, ANAUSN may subdivide the zones even further. For example, a material zone with, say, fifteen mesh intervals may be subdivided into three or four zones for rebalance purposes even though only one material is present throughout the fifteen mesh intervals. Except for cell calculations, ANAUSN attempts to make the average number of mesh intervals per zone equal to four.

The rebalance techniques take the flux iterates calculated as a result of inner iterations and multiply these by scaling factors computed in an attempt to accelerate the iteration procedure. A single scale factor is sometimes computed for each zone, and all fluxes in that zone are multiplied by that factor. With very many mesh intervals per zone, the acceleration procedure is less efficient whereas too few mesh intervals per zone may produce instabilities in the acceleration factors and result in wasted computations.

## 3. INPUT

Except for the title card, all card input is read by SCAN which searches for approved keywords. These are used to identify any string of numeric data items which follow the keyword. The data to be provided for a particular case

fall logically into five data blocks. Unless specifically stated, the ordering of data items within blocks is immaterial.

### 3.1 Data Block 1

One title card with the requested maximum CPU time for the case in integer minutes occupying columns 69-72. If this time is left blank or is greater than the remaining job time the maximum available job time is used.

### 3.2 Data Block 2

This block specifies a set of prelude numbers which determine the problem type and size. They are used to determine allocation of storage arrays for the case. One or other of two keywords must begin this data block. The first option is

PREL (short for PRELUDE) Here and for all keywords only the first four letters are significant. If given, this keyword must be followed by the following 16 integer parameters:

ID        The case number.

IADJ      0 for normal transport calculation, 1 for an adjoint calculation and 2 if both normal and adjoint calculations are to be performed.

NL        0 if isotropic scattering theory is wanted, 1 if  $P_1$  scattering is wanted, 2 if  $P_2$  scattering is wanted, etc.

NSN      The order of  $S_N$  treatment and so must be even.

IEVT      The type of problem:  
           0 for fixed source calculations;  
           1 for  $k_{\text{eff}}$  calculations;  
           2 for eigenvalue search on  $\alpha/v$  absorption, where  $\alpha$  is the eigenvalue,  $v$  is the group velocity;  
           3 for eigenvalue search with whole reactor radius varied, all intervals being varied uniformly - the reactor radius is the eigenvalue;  
           4 for eigenvalue search with the size of material zones being varied individually;

5 for an eigenvalue search on material concentrations;  
 6 for an eigenvalue search for criticality by varying a single transverse buckling  $B^2$  and adding a pseudo-absorption of  $D.B^2$  in each interval of each group - here  $D$  is the material group diffusion coefficient and the eigenvalue is the value of  $B^2$ .

- NMIX The number of entries to be entered in each mixing table for cross sections.
- JOM The geometry number, 0 = slab (plane) geometry, 1 = cylindrical geometry, 2 = spherical geometry.
- IBL The left or inner boundary condition on the angular fluxes; 0 implies a free (vacuum) boundary, 1 implies a perfect specular reflection boundary, 2 implies the periodic boundary condition and 3 implies a white reflective boundary.
- IBR The right or outer boundary condition with the same meanings as IBL.
- NZ The number of material zones.
- NMAT The total number of materials for which cross sections can be supplied.
- NI The number of mesh intervals.
- NQV The number of mesh intervals which contain a fixed isotropic volume source.
- NQA The number of mesh intervals into which fixed angular shell sources are directed.
- NG The number of groups for the problem.
- MAXOUT The maximum number of outer (power) iterations to be used to achieve convergence.

After reading these 16 values, ANAUSN goes on to data block 3.



As an alternative, the second option may be invoked to determine the prelude numbers; here the user enters the keyword AUS instead of PREL. This causes the code to search an AUS geometry datapool GM1 to determine values for the parameters

JOM, IBL, IBR, NZ, NMAT and NI

while a cross section datapool XS1 is searched to determine values for the parameters NL and NG. Default values for the remaining parameters are assigned as

ID = 1 for first case within the task, 2 for second case etc.,  
 IADJ = 0 normal transport calculation,  
 NSN = 4  $S_4$  treatment,  
 IEVT = 1 a  $k_{eff}$  calculation, and  
 NMIX = 0 hence no cross section mixing tables.

Any of these default values can be overridden by entering the parameter name as a keyword and following it with the new value. Thus to specify an eigenvalue search on outer radius using  $S_{16}$  theory, the data for this block could be

AUS IEVT 3 NSN 16

This explicit overriding of default values is essential for a job in which the AUS datapools GM1, XS1 have not been created earlier.

### 3.3 Data Block 3

This block of data may be used to specify certain parameters which further define the problem or else give directives to the code. Again the input procedure consists of entering the keyword which may have to be followed by an input value. Default conditions have been set for all quantities which may be specified here and, in most cases, the block may be omitted. The keywords, corresponding data items, their function and the default values are listed below.

EV followed by 1 real number. First guess for eigenvalue searches.

EV2 followed by 1 real number. Second guess for eigenvalue searches.

The default values selected by the code depend on the type of search.

EPS followed by 1 real. The accuracy in  $k_{\text{eff}}$  and/or the eigenvalue; default is  $5 \times 10^{-4}$ .

EPSP followed by 1 real. The accuracy required in all scalar fluxes; default is  $5 \times 10^{-4}$ .

REQK followed by 1 real. The  $k_{\text{eff}}$  value which searches try to achieve; default is 1.0.

ALFA followed by 1 real. An ALFA/v absorption is added to all group cross sections; default is ALFA = 0.0 .

BSQ followed by 1 real. An absorption  $\text{BSQ}/3\Sigma_{\text{tr}}$  is added; default is BSQ = 0.0 .

MAXI followed by 1 integer. The maximum number of inner iterations allowed per group per outer iteration; default is 100.

INSC followed by 1 integer. Controls the type of rebalance acceleration used after each inner iteration. If INSC = 0, no rebalance is applied, if INSC = 1, Bell's simple scaling, reported by Carlson and Bell [1958], is used so that the reactor fluxes are rebalanced as a whole after each inner, and if INSC > 1 then for INSC inner iterations, a region rebalance will be applied every third iteration and simple scaling used on any other inner iterations. Default value is 30.

IGSC followed by 1 integer. Controls the method of group rebalance used in problems with upscatters. If IGSC = 0, no group rebalance is applied; if IGSC = 1, the fluxes in the thermal upscatter groups are rebalanced globally after each outer iteration using a different scale factor for each upscatter group but applying it uniformly throughout the reactor. If IGSC > 1, the global rebalance is used after most outer iterations, but periodically a separate factor is evaluated for each rebalance zone and each upscatter group. The frequency of application of

this is determined by the next parameter.

MAKS followed by 1 integer. The frequency of application of group and zone rebalance in upscatter problems. Default value is 3. For IGSC the default value is 2 unless a cell calculation is being performed when the default is 1. For problems with no upscatters, these last two parameters are irrelevant.

The remaining keywords in this block serve as directives to the code and no numeric data items follow them.

DIAM forces the code to use the 'diamond difference' scheme when calculating angular fluxes, even when this produces negative angular fluxes, whereas

STEP forces the code always to use the 'step difference' scheme. Step and diamond schemes were defined by Carlson [1963]. The default scheme for the code is to use the diamond scheme except where it results in a negative angular flux, in which case the step scheme is used for that mesh interval and angular direction.

PN directs the code to choose the angular segmentation weights and ordinates to be identical with those corresponding to Legendre quadrature integration over the angular range  $-1 < \mu < 1$ , whereas

DPN directs the code to use weights and ordinates corresponding to separate Legendre quadratures over the angular intervals  $-1 < \mu < 0$  and  $0 < \mu < 1$ . The default scheme uses  $P_N$  quadrature for spherical quadrature,  $DP_N$  quadrature for plane geometry (so as to mitigate any thin slab effect in that geometry) and to use the equal weight quadratures of Brissenden as implemented in the WDSN code [Francescon 1963] for cylindrical geometry.

The next set of directive keywords in this block each direct the code to write a corresponding AUS datapool appropriate to the reactor as defined at the end of the case. Thus

WGM1		GM1	
WXS1		XS1	
WXS2	request datapool	XS2	to be written
WXS3		XS3	
WFL2		FL2	

In a default condition none of the datapools are written unless the keyword AUS is detected anywhere in the input stream. In this event the FL2 datapool is written.

The final set of directive keywords either causes certain arrays to be printed or else stops them from being printed. The general default philosophy is that the user who specified PREL as the first keyword will have all the arrays printed (with the exception of the angular fluxes) whereas the user who specified AUS will get none of the arrays. The keywords are

WRXS to have the cross sections printed, NOXS to delete this

WRCH to have the fission spectra printed, NOCH to delete this

WRED to have input vectors printed, NOED to delete this

WRBA to have zone balance table printed, NOBA to delete this

WRGB to have each group balance table printed, NOGB to delete this

WRAF to have group angular fluxes printed, NOAF to delete this

WRSF to have scalar fluxes printed, NOSF to delete this

### 3.4 Data Block 4

Data within this block consist of keywords, each of which is followed by arrays of data items. The keywords together with the length, type and meaning of the arrays are given below:

R followed by  $NI + 1$  real numbers. The positions  $r_i$  of the reactor meshpoints,  $r_1$ , is the left or inner boundary and  $r_{NI+1}$  is the right or outer boundary. Instead of supplying the mesh point position in this form the user may give

- DR followed by NI reals. The mesh interval spacings, in this case  $r_1$ , is set to zero. Units for R or DR are cm.
- MZI followed by NI integers. The zone material zone numbers for each mesh interval starting at the left-most one.
- MNZ followed by NZ integers. The material numbers for each of the NZ zones.
- MLZ followed by NZ integers. The order of  $P_N$  scattering which applies for each zone in turn.
- CHI followed by NG reals. The fission spectrum for all materials.
- VEL followed by NG reals. The group velocities. Normal units are cm/shake ( $\text{cm}/10^{-8} \text{ s}$ ). This vector should be entered if IEVT = 2 or if ALFA  $\neq$  0.0.
- XS followed by any number of cross section blocks. To enter a block of data for the  $P_L$  component of material number M, the data consist of the integers M,L followed by the cross sections in modified WDSN layout. As many cross section blocks as desired can be entered here in any order; the code will continue to read them until a new keyword is sensed. Unless modified by the mixing tables, these are nominally macroscopic cross sections and the usual units are  $\text{cm}^{-1}$ . The WDSN layout used for a  $P_0$  component consists of NG sets of numbers - one set for each group starting with the highest energy group. The set for group g, say, would consist of the following:
- (i) Two integers LSS and LV : the position in the group vector of the self-scatter cross section and the length of the group vector.
  - (ii) Two reals,  $\Sigma_{tr}$  and  $v\Sigma_f$  : the group transport cross section and the fission emission cross section.
  - (iii) the group vector containing LV reals : the first of which is the group absorption cross section and the remainder being the group outscatter cross sections - one of which is the group self-scatter cross section. The group transport

cross section read-in is only used in problems for which a transverse leakage is significant, i.e. BSQ  $\neq$  0.0 or IEVT = 6 and, in either case, this transverse leakage is modelled in the code by momentarily adding a term BSQ/(3 $\Sigma_{tr}$ ) to each group total cross section and hence to the implied group absorption cross section.

For a  $P_L$  component cross section block with  $L > 0$ , the block layout is identical except that  $\Sigma_{tr}$ ,  $\Sigma_{abs}$  and  $\nu\Sigma_f$  must be zero.

The  $P_L$  cross section components,  $\sigma_{gg'}^{\ell}$ , are defined so that the angular scattering source in group  $g$ , direction  $\mu$  is

$$S_g(\mu) = \frac{1}{2} \sum_{g'} \sum_{\ell=0}^{NL} (2\ell+1) \sigma_{gg'}^{\ell} P_{\ell}(\mu) \int_{-1}^1 P_{\ell}(\mu') \phi_{g'}(\mu') d\mu'$$

and ANAUSN inserts the  $(2\ell+1)$  factor when computing this source.

- DC followed by NM reals. The direction cosines for the discrete directions of the  $S_N$  method. Here  $NM = NSN+1$  for plane and spherical geometry, but  $NM = NSN*(NSN+4)/4$  for cylindrical geometry.
- W followed by NM reals. The corresponding  $S_N$  direction weights. The weights entered as data are normalised again by the code. Default values for the DC and W arrays are available.
- FD followed by NI reals. A first guess for the fission density in each interval. Default values are generated if this array is not given explicitly.
- SF followed by NI\*NG reals. A first guess for the group scalar fluxes. Again, the code will generate a default array if necessary.
- MQV followed by NQV integers. Only entered for source problems with  $NQV > 0$ . The array lists the interval numbers which contain a fixed isotropic source.

- QV followed by  $NG \cdot NQV$  reals. The fixed isotropic source density spectra for each of the  $NQV$  source intervals in turn. Units are particles/cm<sup>3</sup>/s.
- MQA followed by  $NQA$  integers. Only entered in  $NQA > 0$ . The array less the interval numbers into which fixed angular dependent shell sources are directed.
- QA followed by  $NM \cdot NG \cdot NQA$  reals. The shell source densities ordered by  $S_N$  directions, by group and by interval. If, for example, the first element of the MQA array is  $MQA(1) = 3$ , a non-zero element of the QA array such as  $QA(m,g,1) = q$  means that the group  $g$  angular flux in the discrete direction  $m$  is increased by an amount  $q$  over that leaving the 'previous' interval. If the direction cosine  $\mu_m$  is positive, this previous interval will be interval 2, but if  $\mu_m$  is negative, the previous interval will be interval 4.
- ZMOD followed by  $NZ$  reals. Only entered if  $IEVT = 4$ ; in this case the material zone thicknesses will be expanded or contracted according to this array of zone modifiers. Thus if mesh interval  $i$  (of thickness  $\delta_i$  according to the original  $R$  or  $DR$  array) lies in zone  $j$ , its thickness when the eigenvalue has a value  $E$  will be set to  $\delta_i[1+E \cdot ZMOD_j]$ .
- MTE followed by  $NMIX$  integers,
- MTC followed by  $NMIX$  integers, and
- ATD followed by  $NMIX$  reals. The mixing tables which can only be entered if  $NMIX > 0$  and must be entered if  $IEVT = 5$ . The tables are interpreted by ANAUSN before iterations start and, if  $IEVT = 5$ , again after each estimate of  $k_{eff}$  has been made with the current eigenvalue. The three mixing rules are shown below, where  $\ell$  and  $m$  are positive integers and  $x$  is a real:

MTE	MTC	ATD	
0	$\pm m$	x	means multiply all cross sections for material m by x.
m	$\pm m$	0.0	means multiply all cross sections for material m by the current eigenvalue.
$\ell$	$\pm m$	x	means add a multiple x of the cross sections for material $\ell$ to those for material m.

All cross section mixing can be done with these rules. A negative entry in the MTC array means that the corresponding operation is only performed on the first interpretation of the mixing table. This provides a means of converting microscopic data to macroscopic form or, alternatively, of using a standard material cross section block but using a density or voidage factor with it. If the keyword

AUS was given in data block 2, the code will remember this fact and search the appropriate datapools to find the eight arrays R, MZI, MNZ, MLZ, CHI, VEL, SF and XS before interpreting any keywords in the present block. However, the keyword AUS can also be given in the present data block; in this case the datapools in which they exist will be searched.

### 3.5 Data Block 5

This consists simply of the keyword END and signals the completion of data for the case.

### 3.6 Minimal Input

A special option is provided for the AUS user who has set up cross section and geometry datapools XS1 and GM1 to define the system and who wishes simply to have a calculation of multiplication factor  $k_{eff}$  made for that system. If the user supplies an empty data set as input, the code senses this and generates all necessary input. The same result could have been achieved by supplying as data the following cards:



TITLE CARD FOR THE CASE  
AUS END

#### 4. OUTPUT

##### 4.1 AUS Datapools

If the code is executed within the AUS scheme, a WDSN flux dump is always written at the termination of a case - whether the case has been converged or not. Termination without convergence will occur if the maximum allowable number of outer iterations has been reached, or if the maximum allowable time for the job seems likely to be exceeded. In the latter situation, the flux dump can be used as a flux guess to restart the calculation during another job. Datapools for geometry and cross sections will be written if the case input requests them.

##### 4.2 Printed Output

Printed output for a case which appears on FORTRAN logical unit 3 can be brief or very full.

###### 4.2.1 Minimal printed output

The minimum output consists of a marked listing of the case input (the marking being a statement of the keywords processed), a description of any AUS datapools from which data are taken, a monitor print for each outer iteration, a statement of any AUS datapools written after the case is terminated, and case timing information which is self-explanatory.

The monitor line is of interest mainly with problems for which convergence is difficult to achieve. The items printed in the monitor line are

- . the outer iteration number;
- . the cumulative count of inner iterations;
- . the present eigenvalue estimate;

- the total fission rate in the system for this outer iteration (For eigenvalue problems the fluxes are always normalised so that there is exactly one fission neutron released; this monitor item is then the current estimate of the fission multiplication factor  $k_{\text{eff}}$ .);
- the overall neutron balance for the system which should be exactly 1.0;
- $\lambda$  which, for eigenvalue problems, is the ratio of successive fission rates, but for fixed source problems is the ratio of successive estimates of fixed source plus fissions plus upscatter reactions (In both cases this should be 1.0 for a converged problem.);
- the maximum proportional change in any group scalar flux in any interval during the last outer iteration;
- the group and the interval in which this flux change occurred;
- an estimate of the dominance ratio when Chebychev acceleration of the fission density is being used;
- the maximum proportional change in fission density during the last outer iteration;
- the maximum and minimum scale factors used in the group rebalance for this outer iteration; and
- an indicator to show the form of rebalance applied on this outer iteration, namely, group and zone rebalance, simple group rebalance or no rebalance at all.

#### 4.2.2 Other printed output

A variety of other quantities that are descriptive of the case can be printed out. The presence or absence of these items from the case output is determined by the user's input, as described in Section 4.3. Heading information is supplied with each item to make it self-descriptive.

## 5. CORE MEMORY REQUIREMENTS AND CODE STRUCTURE

For a transport theory code to be capable of managing large problems, it was once necessary for the code to make provision for saving much of the information in its work areas on subsidiary storage, i.e. magnetic tape or disk. The ANISN code [Engle 1967], the best known of such codes, incorporated efficient and elegant methods for achieving this result. From the time when development of ANAUSN was started, it was reasonably clear that the available size of core memory would increase dramatically. ANAUSN was therefore designed on the basis that all information required for the calculation would be held in core. For a time this limited the size of problem which could be handled, but the parallel development of multi-programming operating systems with essentially unlimited virtual memory allocation has alleviated this problem.

When execution of ANAUSN begins, the MAIN routine starts an internal clock, determines the amount of core memory available and passes this to the subroutine CONTRL which supervises the calculations, returning to the main routine only at the completion of the task. CONTRL reads as much of the input data as is necessary to determine the problem size before computing address pointers for the different arrays necessary for solution of the problem. The remaining input data are then read into the appropriate array positions before the subroutine RUN is called. This and other subroutines called use variably dimensional FORTRAN arrays with dimensions set from the routine CONTRL.

The main task of subroutine RUN is to handle eigenvalue search problems if these have been requested. For each possible reactor configuration during the search, a call is made to the subroutine POWER to determine the multiplication factor  $k_{\text{eff}}$  of the configuration but for non-search problems, a single call to POWER is sufficient.

The POWER subroutine actually supervises the iterative solution of the multi-group transport equations. At the beginning of an iteration, a set of group scalar fluxes will be available as well as the Legendre polynomial weighted integrals of the group angular fluxes necessary for the solution of problems with anisotropic scattering. From the fluxes, the spatial dependence of the fission density is calculated, then for each group in turn a call is made to an INNER subroutine to determine the angular and scalar fluxes in each reactor interval for that group. From the results, the POWER subroutine can determine whether the solution has converged or whether further iterations are

necessary. For thermal problems in which the group structure allows upscatters to occur, the POWER subroutine also applies zone and/or group rebalance techniques to accelerate the convergence. When appropriate, a standard Chebychev acceleration of the fission density is also applied. The rebalance/acceleration techniques are discussed in more detail in Section 6.

It was advantageous to provide two versions of the INNER subroutines. The normal version handles the solution of the transport equation in an individual group by techniques which are now standard - see for example the discussion by Carlson [1963] and the description of the Winfrith DSN code by Francescon [1963]. Implementation of these techniques in FORTRAN means that before computing an angular flux in any direction in any mesh interval, it is necessary to test whether anisotropic scattering applies and whether fixed sources are present. For an important class of problems - some reactor cell calculations for example - neither of these situations will apply and a special truncated INNER subroutine called FASTIN is accessed in which these tests need not be made. Acceleration of the inner iterations is carried out by means of zone rebalance.

## 6. REBALANCE PROCEDURES

Acceleration of both inner and outer iteration loops in ANAUSN is achieved by use of rebalance procedures which are generalisations of the scaling procedure first proposed by Carlson and Bell [1958]. The inner iteration rebalance is essentially that developed by Engle and Mynatt [1968] who calculated and used a different rebalance factor for each interval. The ANAUSN modification requires the use of only one rebalance factor for each rebalance zone which typically will contain about four mesh intervals. This is less subject to instabilities.

For the outer iteration loop let us use the group and zone rebalance scheme alluded to previously. Suppose that we are performing the (n)-th outer iteration and have just converged the fluxes for group g. The source term for group g will have included inscatters from other groups and these inscatters will have been calculated using the most recent flux estimates in the groups. Some of these estimates will be those obtained during the present (n)-th iteration and others will be those obtained from the previous (n-1)-th iteration. If the volume integrated group g flux in zone i found during iteration n is denoted by  $\phi_{gi}^n$ , these zone fluxes will satisfy the equation

$$\begin{bmatrix} Q_{gi} + R_{g,i-1}^n + \sum_{g' < g} \sigma_{gg'i} \phi_{g'i}^n \\ + L_{gi+1}^n + \sum_{g' > g} \sigma_{gg'i} \phi_{g'i}^{n-1} \end{bmatrix} = \begin{bmatrix} L_{gi}^n + (\sigma_{gi} - \sigma_{ggi}) \phi_{gi}^n \\ + R_{gi}^n \end{bmatrix} \quad (1)$$

where

$\sigma_{gg'i}$  is the macroscopic scattering cross section in zone  $i$  for scattering from group  $g'$  into group  $g$ ;

$\sigma_{gi}$  is the total cross section in zone  $i$  group  $g$ ;

$Q_{gi}$  is the total non-scattering source into zone  $i$  group  $g$  including fission sources if present;

$R_{gi}^n$  is the flow of particles leaving zone  $i$  in group  $g$  travelling to the right; and

$L_{gi}^n$  is the corresponding flow travelling to the left, both flows being calculated during the present iteration ( $n$ ).

So  $R_{gi}^n - L_{gi+1}^n$  is the net current to the right at the interface between zones  $i, i+1$ .

In Equation (1), the presence of the  $\phi_{gi}^{n-1}$  terms which are not identical with  $\phi_{gi}^n$  shows that the solution is not yet converged. The rebalance scheme assumes that if the  $\phi_{gi}^n$  terms were multiplied by appropriate scale factors  $F_{gi}$ , these results would be the final correct solutions and would satisfy the equation

$$\begin{bmatrix} Q_{gi} + R_{g,i-1}^n F_{g,i-1} + \sum_{g' < g} \sigma_{gg'i} \phi_{g'i}^n F_{g'i} \\ + L_{gi+1}^n F_{g,i+1} + \sum_{g' > g} \sigma_{gg'i} \phi_{g'i}^n F_{g'i} \end{bmatrix} = \begin{bmatrix} L_{gi}^n F_{gi} + (\sigma_{gi} - \sigma_{ggi}) \phi_{gi}^n F_{gi} \\ + R_{gi}^n F_{gi} \end{bmatrix} \quad (2)$$

the particle current terms having been multiplied by the appropriate scale factors.

If Equation (1) is multiplied by  $F_{gi}$  and Equation (2) subtracted from the result, we recover the equation



After zone-group rebalance has been disabled, some problems may still be slow to converge if fissions are present. For this reason, a simple Chebychev acceleration of the fission density is initiated.

## 7. INVOCATION OF ANAUSN

ANAUSN may be executed on the AAEC's central IBM3033 computer either as a stand-alone program or as a module within the AUS system. In stand-alone mode, it can be invoked through a catalogued procedure, also named ANAUSN. This procedure, listed below, specifies all data sets necessary to execute the code and allows the results of calculations to be saved in AUS datapools. As a module within the AUS scheme the code can be invoked within the AUSYS supervisor [Robinson 1975] by the statement

```
LINK ANAUSN
```

which is an abbreviation of the standard linkage statement

```
LINK ANAUSN(FT01F001,DD2),(FT02F001,DD12),(FT03F001,DD13),
(FT08F001,DD21),(FT09F001,DD33),(FT10F001,DD33),(FT11F001,DD35),
(FT11F001,DD37),(FT12F001,DD22),(FT23F001,DD40).
```

This linkage is appropriate if the ANAUSN data are in the dataset DD2. If the data are in another dataset DDn, the appropriate AUSYS statement is

```
LINK ANAUSN(1,DDn)
```

7.1 ANAUSN Catalogued Procedure

```

/** ANAUSN QUERIES: R.F.CIANCY
//ANAUSN PROC PRG=PROGPGM,
// PQ=1,SQ=10,P=1,S=1,
// XS1='&&XS1',DISPXS1=NEW,
// XS2='&&XS2',DISPXS2=NEW,
// XS3='&&XS3',DISPXS3=NEW,
// GM1='&&GM1',DISPGM1=NEW,
// FL2='&&FL2',DISPFL2=NEW
//ALLOC EXFC PGM=NEWEOF,REGION=10K,
// PARM=(&DISPXS1,&DISPXS2,&DISPXS3,&DISPFL2,&DISPGM1,OLD)
//STEPLIR DD DSN=AUS.SYS,DISP=SHR
//ALL1 DD DSN=&XS1,DISP=(&DISPXS1,CATLG),SPACE=(TRK,(&PQ,&SQ)),
// UNIT=SYSDA,DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//ALL2 DD DSN=&XS2,DISP=(&DISPXS2,CATLG),SPACE=(TRK,(&PQ,&SQ)),
// UNIT=SYSDA,DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//ALL3 DD DSN=&XS3,DISP=(&DISPXS3,CATLG),SPACE=(TRK,(&PQ,&SQ)),
// UNIT=SYSDA,DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//ALL4 DD DSN=&FL2,DISP=(&DISPFL2,CATLG),SPACE=(TRK,(&PQ,&SQ)),
// UNIT=SYSDA,DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//ALL5 DD DSN=&GM1,DISP=(&DISPGM1,CATLG),SPACE=(TRK,(&PQ,&SQ)),
// UNIT=SYSDA,DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//ALL6 DD DUMMY
//SYSPRINT DD SYSOUT=A
//GO EXEC PGM=&PRG,REGION=
//STEPLIR DD DSN=REC.ANAUSNEW.PROGRAM,DISP=SHR
//FT01F001 DD DDNAME=SYSIN
//FT02F001 DD SYSOUT=R
//FT03F001 DD SYSOUT=A
//FT08F001 DD UNIT=SYSDA,SPACE=(CYL,(&P,&S)),
// DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//FT09F001 DD DSN=&XS1,DISP=OLD,SPACE=(TRK,(&PQ,&SQ),RLSE)
//FT10F001 DD DSN=&GM1,DISP=OLD,SPACE=(TRK,(&PQ,&SQ),RLSE)
//FT11F001 DD DSN=&FL2,DISP=OLD,SPACE=(TRK,(&PQ,&SQ),RLSE)
//FT12F001 DD UNIT=SYSDA,SPACE=(CYL,(&P,&S)),
// DCR=(RECFM=VRS,RLKSIZE=4584,LRFCL=X)
//FT22F001 DD DSN=&XS2,DISP=OLD,SPACE=(TRK,(&PQ,&SQ),RLSE)
//FT23F001 DD DSN=&XS3,DISP=OLD,SPACE=(TRK,(&PQ,&SQ),RLSE)
/** END OF PROCEDURE "ANAUSN"

```

8. ACKNOWLEDGEMENTS

To record the names of all the people who, through their advice and suggestions, have made valuable contributions to the development history of ANAUSN would make this section ridiculously long; the author's thanks are extended to them all. It would, however, be improper not to note the special contributions made by G. Doherty and G.S. Robinson. Without their advice, ideas and assistance, the code would not have reached its present status.

9. REFERENCES

Bennett, N.W. and Pollard, J.P. [1967] - SCAN - A Free Input Subroutine for the IBM360. AAEC/TM399.



- Carlson, B.G. [1963] - The Numerical Theory of Neutron Transport. In Methods of Computational Physics (Alder, B., Fernbach, S. and Rotenberg, M. Eds.). Vol.1, Academic Press, New York, p.1.
- Carlson, B. and Bell, G. [1958] - Solution of the Transport Equation by the  $S_N$  Method. Proc. 2nd UN Conf. on the Peaceful Uses of Atomic Energy, Geneva, 16:535.
- Carlson, B., Lee, C. and Worlton, J. [1960] - The DSN and TDC Neutron Transport Codes. LAMS-2346.
- Clancy, B.E. and Donnelly, I.J. [1970] - Outer Iteration Scaling in Neutron Transport Codes. Nucl. Sci. Eng., 39:398.
- Cox, G.W. [1971] - DARRAY. AAEC Unpublished Report.
- Engle, W.W. Jr. [1967] - A Users' Manual for ANISN. Union Carbide Corporation, Nuclear Division. K-1693.
- Engle, W.W. Jr., and Mynatt, F.R. [1968] - A Comparison of Two Methods of Inner Iteration Convergence Acceleration in Discrete Ordinate Transport Codes. Trans. Am. Nucl. Soc., 11:193.
- Francescon, S. [1963] - The Winfrith WDSN Program. AAEW-R273.
- Robinson, G.S. [1975] - AUS - The Australian Modular Scheme for Reactor Neutronics Computations. AAEC/E369.