

11

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

DMM

ASAE-4

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

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

III

TABLE OF CONTENTS

Acknowledgements	iii
Abstract	a-iii
Introduction	1
The Diffusion Equation	1
The Multigroup Method	3
The Spatial Integration	4
Boundary Conditions	4
Numerical Procedure	6
The Nuclear Constants	8
Self-Shielding	13
Xenon Addition	14
Burnup	15
The Adjoint Equation	17
Determination of Critical Conditions	19
Neutron Balance	20
Iteration on the Perpendicular Buckling	22
Figure 1. Cross Sections approximated by step-functions	11
Figure 2. Method of combining the cross sections of the isotopes present	12

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

A B S T R A C T

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, re-computation 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:

$$\nabla \cdot (\mathbf{v}, E) \nabla \phi (x, E) - \sum_t (x, E) \phi (x, E) \\ + \sum_{A1} \sum_{ir_A} (\mathbf{v}_{ir_A} (x, E) \int_E^{E_{max}} \sum_{in_A} (\mathbf{v}_{in_A} (x, E')) \chi_{in_A} (x, E') \phi (x, E') \frac{dE'}{\chi_{in_A} (x, E')})$$

$$+ \sum_A \int_E^{\min(E/\sigma_A, E_{\max})} \frac{\sum_{\sigma_A} \phi(x, E')}{(1-\sigma_A) E'} dE' \quad (1)$$

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

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 $\chi_{in}(x, E)$ is given by

$$\chi_{in}(x, E') = \int_{E_{\min}}^E \chi_{in}(x, E'') dE'' \quad (2)$$

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

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

$\sum_{in_A}(x, E)$ 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.

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

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

\sum_{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.

\sum_f and γ_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.

$$-\nabla D^i(x) \nabla \phi^i(x) + T^i(x) \phi^i(x) = H^i(x); i=1, 2, 3 \dots I \quad (3)$$

where

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

and

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

The nuclear constants D^i , m^i , m^{ij} , χ^i , F^i , 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, $0, x_1, x_2 \dots x_N$ at which the functions of Equations (3), (4), and (5) are defined. Setting $\ell=0$ 1 or 2 for planar, cylindrical and spherical geometries respectively. we obtain:

$$-a_n^i \phi_{n+1}^i + b_n^i \phi_n^i - c_n^i \phi_{n-1}^i = d_n^i \quad (n = 0, 1, 2 \dots N) \quad (6)$$

$$a_n^i = \left(1 + \frac{\Delta x_n}{2x_n} \right)^\ell D_n^i \frac{\Delta x_{n-1/2}}{x_n}$$

$$b_n^i = a_n^i + c_n^i + \left(\Delta x_{n-1/2} \right)^2 T_n^i \quad (n = 1, 2 \dots N-1) \quad (7)$$

$$c_n^i = \left(1 - \frac{\Delta x_{n-1}}{2x_n} \right)^\ell D_n^i \frac{\Delta x_{n-1/2}}{\Delta x_n}$$

$$d_n^i = \left(\Delta x_{n-1/2} \right)^2 H_n^i$$

where

$$\Delta x_n = x_{n+1} - x_n \quad (8)$$

$$\Delta x_{n-1/2} = 1/2 (x_{n+1} - x_{n-1})$$

Boundary Conditions

At the boundaries of the mesh, $x_0=0$ and x_N , the equations for a_n^i , b_n^i , c_n^i , and d_n^i are modified to agree with the boundary conditions.

These are

$$-\left(1-B_O^i\right) D_O^i - \frac{\phi^i - \phi_O^i}{\Delta x_O} + B_O^i \phi_O^i = 0 \quad (9)$$

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

which correspond to the physical boundary conditions

$$\begin{aligned} -\left[1-B_O(E)\right] D(0,E) \nabla \phi(0,E) + B_O(E) \phi(0,E) &= 0 \\ \left[1-B_N(E)\right] D(X,E) \nabla \phi(X,E) + B_N(E) \phi(X,E) &= 0 \end{aligned} \quad (10)$$

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 .

$$\begin{aligned} a_O^i &= \left(1-B_O^i\right) D_O^i \\ b_O^i &= a_O^i + B_O^i \Delta x_O \\ c_O^i &= 0 \\ d_O^i &= 0 \end{aligned} \quad (11)$$

$$a^i_N = 0$$

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

$$c^i_N = (1 - B^i_N) D_N$$

$$d^i_N = 0$$

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

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

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

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

$$q^i_n = \frac{d^i_n + c^i_n q^i_{n-1}}{b^i_n - c^i_n p^i_{n-1}} \quad (14)$$

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

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

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

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

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

$$P_n' = \sum_{j=1}^I F_n^j \phi_n^j \quad (16)$$

these are normalized by

$$\begin{aligned} P_n &= \frac{1}{k} P_n' \\ \frac{1}{V} \sum_{n=1}^{N-1} P_n \Delta v_n &= 1 \end{aligned} \quad (17)$$

where

$$\begin{aligned} \Delta v_n &= x_n^\ell \Delta x_n \\ V &= \sum_{n=1}^N \Delta v_n \end{aligned} \quad (18)$$

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_n^i 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).

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

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

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).

$$D^i = \left[\int_{E_{i+1}}^{E_i} D(E) \phi(E) dE \right] \quad (20)$$

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

$$- \sum_A \left[\int_E^{\min(\frac{E}{\alpha_A}, E_i)} \left\{ \frac{\sum s_A(E')}{(1-\alpha_A)E'} \phi(E') dE' \right\} dE \right] \quad (21)$$

$$\frac{-B^2 D^i}{}$$

$$T^{ij} = \left[\int_{E_{i+1}}^{E_i} \left\{ \sum_A \chi_{in_A}(E) \right\} \sum_{in_A}^{E_j} (E') \psi_{in_A}(E') \phi(E') \frac{dE'}{\sum_{in_A}^{E_i}(E')} \right] (22)$$

$$+ \sum_A \left[\int_{E_{j+1}}^{\min(E_{i+1}, E_j)} \left\{ \frac{\sum s_A(E')}{(1-\phi_A)E'} \phi(E') dE' \right\} dE \left\{ \frac{E_j - E_{j+1}}{E_i - E_{i+1}} \right\} \right] \left[\int_{E_{j+1}}^{E_j} \phi(E) dE \right]$$

$$\chi_i = \left[\int_{E_{i+1}}^{E_i} \chi_f(E) dE \right] \left[\frac{E_i - E_{i+1}}{E_i - E_{i+1}} \right] (23)$$

$$F^j = \left[\int_{E_{j+1}}^{E_j} \sum_f (E) \psi_f \phi(E) dE (E_j - E_{j+1}) \right] \left[\int_{E_{j+1}}^{E_j} \phi(E) dE \right] (24)$$

$$\phi^i = \left[\int_{E_{i+1}}^{E_i} \phi(E) dE \right] \left[\frac{E_i - E_{i+1}}{E_i - E_{i+1}} \right] (25)$$

B^λ 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 ϕ_n^i by integrating the ϕ_n^i over regions of uniform composition and interpolating. Thus, if

$$k\phi^i = \sum_{n=1}^{N_k} \phi_n^i \Delta V_n (26)$$

where k indicates the k^{th} region,

$$k\psi(E) = k_{C_1}^{i-1} k_{C_2}^i / E + k_{C_3}^i E \quad E_{i+\frac{1}{2}} \leq E < E_i \quad (27)$$

The coefficients are defined by

$$\begin{aligned} k_{\phi}^{i-1} &= k_{C_1}^i + \frac{k_{C_2}^i}{E_{i-\frac{1}{2}}} + k_{C_3}^i E_{i-\frac{1}{2}} \\ k_{\phi}^i &= k_{C_1}^i + \frac{k_{C_2}^i}{E_{i+\frac{1}{2}}} + k_{C_3}^i E_{i+\frac{1}{2}} \\ k_{\phi}^{i+1} &= k_{C_1}^i + \frac{k_{C_2}^i}{E_{i+\frac{1}{2}}} + k_{C_3}^i E_{i+\frac{1}{2}} \\ E_{i+\frac{1}{2}} &= \frac{E_{i+1} + E_i}{2} \end{aligned} \quad (28)$$

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/λ . The E/λ 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.

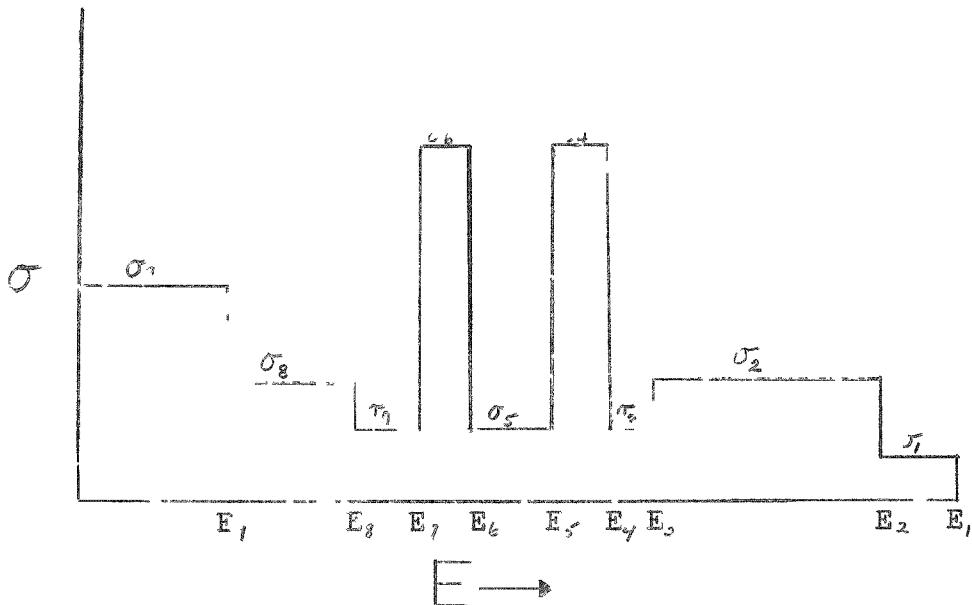


Figure 1. Cross Section: approximated by step-functions.

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 , σ_{in} , γ_1 , σ_f , γ_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_i The upper energy limits of the groups to be used in the main code.

Figure 2, Page 2, explains the method of combining the cross sections of the 10 types present. Here we define σ_A as the cross section

15

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

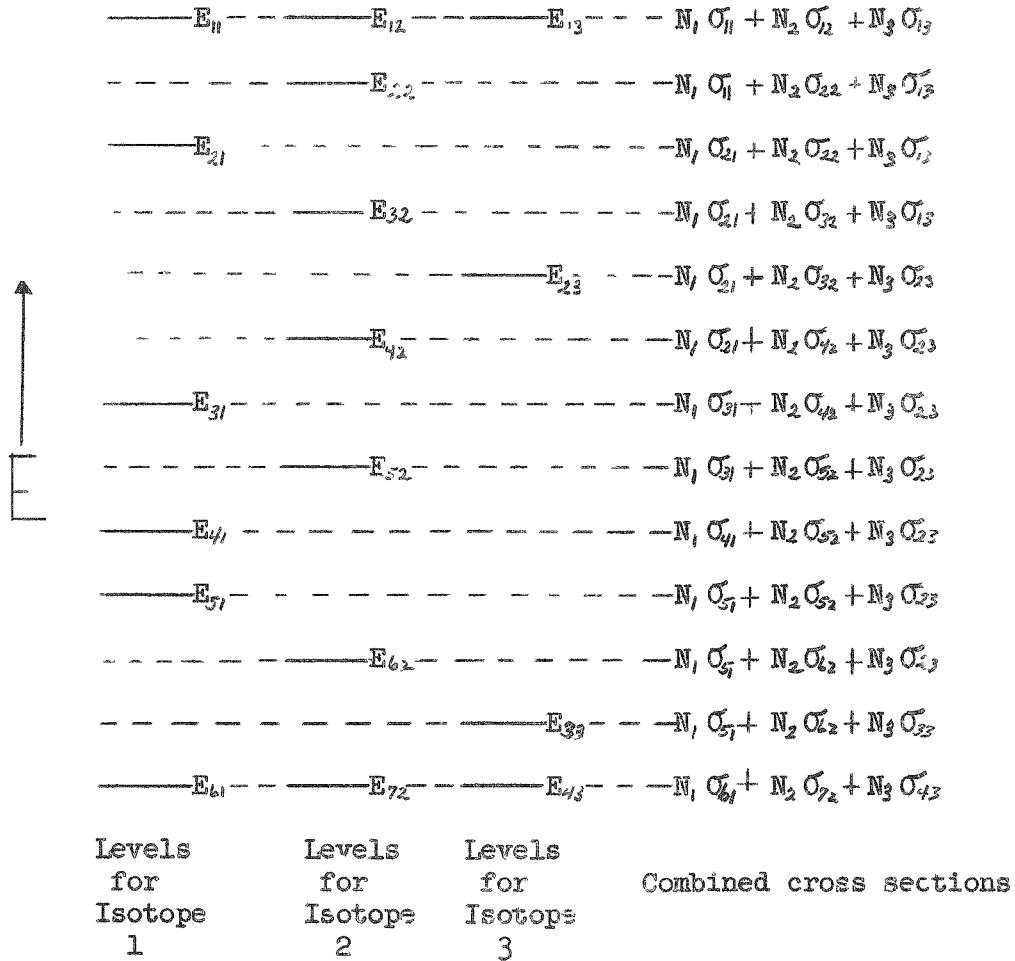


Figure 2. Method of combining the cross sections of the isotopes present.

In this way the cross sections are combined to get a set for a new "isotope" of atomic weight $N_1 A_1 + N_2 A_2 + \dots$. At this point, the new isotope 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 $D(E) = \frac{1}{3 \sum_i}$.

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

$$C_N = \frac{0.6023 e}{\sum_A \alpha_A N_A}$$

The ϕ_n^i 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 ϕ_n^i 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

ϕ - 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

$$k_{N_{Xe}}(t) = k_{N_{Xe}^0} \frac{\Delta t}{e^{-\lambda_{Xe} \Delta t}} + N_{I^0} (e^{-\lambda_I \Delta t} - e^{-\lambda_{Xe} \Delta t}) \quad (30)$$

$$N_{xe} = \frac{\int_{E_{min}}^{E_{max}} \Sigma_{xe} \cdot \frac{N_A}{k_A \cdot \lambda_{xe} + k_{xe}}}{k_p} \cdot k_p$$

$$k_p = 1/2.5 \quad \frac{N_A}{\lambda_{xe} + k_{xe}} \quad P_n \Delta V_n$$

(31)

$$k_{xe} = L^P \int_{E_{min}}^{E_{max}} \psi(E, \sigma_{xe}(E)) dE$$

The Σ_{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 Σ_{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_0

K_0

- 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_0

T

- the length of the time step to be used

L

- the power level defined in the previous section

Note: *If $\Delta t=0$, $k_{N_{xe}} = k_{N_{xe}}^2$, the equilibrium xenon concentration. Also $k_{N_{xe}} = 0$ if $k_p = 0$ so that unit 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, $\sigma_{A \rightarrow A'}$.

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

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

$$k \sum_{A \rightarrow A'}^i = k_{N_A} k_{C_N} \left[\int_{E_{i+1}}^{E_i} \sigma_{A \rightarrow A'} k \psi^i(E) dE \right] / \left[\int_{E_1}^{E_i} k \psi^i(E) dE \right] \quad (32)$$

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

$$\delta N_A = \frac{\delta T \rho_L}{k_{C_N}} \sum_{i=1}^I \left[\sum_{A' \rightarrow A} \left(k \sum_{A \rightarrow A'}^i - k \sum_{A \rightarrow A'}^i \right) k \phi^i \right] (E_i - E_{i+1}) \quad (33)$$

Having computed the new compositions of the regions, it is necessary to adjust the concentration of isotope A in order to keep K equal to K_0 . δ is added to k_{N_A} and the program transfers to the cross section averaging routine to compute new nuclear constants and thence to the main program. After N iterations the new value of K is compared with K_0 . If $K < K_0$, 0.1δ is added to k_{N_A} . If $K = K_0$, 0.1δ is subtracted from k_{N_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.

$$N_{A_{i+1}} = N_{A_i} + \left(N_{A_i} - N_{A_{i-1}} \right) \frac{K_0 - K_i}{K_i - K_{i-1}} \quad (34)$$

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

$K_0 - K_1 \leq \epsilon$. 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

$$\begin{aligned}
 & \nabla D(x, E) \nabla \phi^*(x, E) - \sum_t (x, E) \phi^*(x, E) \\
 & + \sum_A \sum_{in_A} (x, E) \frac{\nu_{in_A}(x, E)}{\chi_{in_A}(x, E)} \int_{E_{min}}^E \chi_{in_A}(x, E') \phi^*(x, E') dE' \\
 & + \sum_A \sum_{s_A(E)} \frac{s_A(E)}{(1-\alpha_A)E} \int_{\max(E_{min}, \frac{E}{A})}^E \phi^*(x, E') dE' \\
 & = \frac{1}{K^*} \sum_f (x, E) \nu_f(x, E) \int_{E_{min}}^{E_{max}} \chi_f(x, E') \phi^*(x, E') dE' \tag{35}
 \end{aligned}$$

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

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

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

$$P^*(x) = \frac{1}{K} \sum_{j=1}^J F^{*j} \phi^{*j}(x) \quad (38)$$

The only difference between these equations and Equations (3), (4), and (5) is that the sum in Equation (7) 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

$$D^{*i} = \left[\int_{E_{i+1}}^{E_i} D(E) \phi^*(E) dE \right] \quad (39)$$

$$T^{*i} = \left[\int_{E_{i+1}}^{E_i} \left\{ \sum_{A} t_A(E) \phi^*(E) - \sum_{A} \sum_{in_A} \frac{\gamma_{in_A}(E)}{\chi_{in_A}(E)} \int_{E_{i+1}}^E \chi_{in_A}(E') \phi^*(E') dE' \right\} dE \right] \quad (40)$$

$$- \sum_{A} \frac{\sum_{in_A}(E)}{(1-\alpha_A)E} \int_{\max(E_{i+1}, \sim E)}^E \phi^*(E') dE' \quad (40)$$

$$T^{*ij} = \left[\int_{E_{i+1}}^E \left\{ \sum_{A} \frac{\sum_{in_A}(E) \gamma_{in_A}(E)}{\chi_{in_A}(E)} \right\} \chi_{in_A}(E') \phi^*(E') dE' \right] \quad (41)$$

$$+ \sum_A \frac{\sum s_A(E)}{(1-\alpha_A)E} \left[\int_{\max(E_{j+1}, E)}^{E_j} \phi^*(E') dE' \right] dE \left[\frac{E_j - E_{j+1}}{E_i - E_{i+1}} \right] \left[\int_{E_{j+1}}^{E_j} \phi^*(E) dE \right] \quad (41)$$

$$\chi^{*i} = \left[\int_{E_{i+1}}^{E_i} \sum_f (E) \psi_f (E) dE \right] \left[\frac{E_i - E_{i+1}}{E_i - E_{i+1}} \right] \quad (42)$$

$$F^* = \left[\int_{E_{j+1}}^{E_j} \chi_f (E) \phi^*(E) dE \left(E_j - E_{j+1} \right) \right] \left[\int_{E_{j+1}}^{E_j} \phi^*(E) dE \right] \quad (43)$$

$$\phi^{*i} = \left[\int_{E_{i+1}}^{E_i} \phi^*(E) dE \right] \left[\frac{E_i - E_{i+1}}{E_i - E_{i+1}} \right] \quad (44)$$

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^{in} .

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 input, we specify the address of single number, Z , which is to be varied, the number of iterations, \mathcal{N} , between variations, a first guess, \mathcal{J} , for the amount by which it is to be varied, K_0 , the reactivity desired, and ξ , the tolerance for the deviation of the final reactivity from K_0 . 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 \bar{Z} is the same as Equation (34),

$$\bar{Z}_{i+1} = \bar{Z}_i + (\bar{Z}_i - \bar{Z}_{i-1}) \frac{K_o - K_i}{K_i - K_{i-1}} \quad (45)$$

Trials are made iteratively until $|K_o - K_i| \leq \xi$.

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 :

$$k \mathcal{D}^i = \sum_{n=N_{k-1}}^{N_k} \sum_{j=1}^I k_{Tji} \phi_n^i \Delta v_n \quad (46)$$

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

$$k \mathcal{S}^i = \sum_{n=N_{k-1}}^{N_k} \sum_{j=1}^{i-1} k_{Tij} \phi_n^j \Delta v_n \quad (47)$$

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

$$k \mathcal{F}^i = \sum_{n=N_{k-1}}^{N_k} p_n \chi^i \Delta v_n \quad (48)$$

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

$$k_C^i = \sum_{n=N_{k-1}}^{N_k} k_T^i \phi_n^i \Delta v_n - h_D^i = k_L^i \quad (49)$$

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

$$k_L^i = k_D^i \frac{\phi_{N_{k-1}+1}^i - \phi_{N_{k-1}}^i}{k \Delta x} \quad (50)$$

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

$$k_L^i = -k_D^i \frac{\phi_{N_k}^i - \phi_{N_{k-1}}^i}{k \Delta x} \quad (51)$$

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

$$k_L^i = \sum_{n=N_{k-1}}^{N_k} B^2 k_D^i \phi_n^i \Delta v_n \quad (52)$$

8) The flux in each group in region k:

$$k\phi^i = \sum_{n=N_{k-1}}^{N_k} \phi_n^i \Delta v_n \quad (53)$$

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 $B^2 D$ 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} , \mathcal{L}_+ , and \mathcal{L}_- are recorded. If the regions from k_{c_1} to k_{c_2} have been specified as the core region, the program determines c_{Bi}^2 from the equation

$$c_{Bi}^2 = \frac{k_{c_2} \mathcal{L}_+ + k_{c_1} \mathcal{L}_- - k_{c_2}^i \mathcal{L}_+^i - k_{c_1}^i \mathcal{L}_-^i}{\sum_{k=k_{c_1}}^{k_{c_2}} k \phi^i D^i} \quad (54)$$

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

After performing the reciprocal calculation a set of cB_i are computed from its results in the same manner and used in the initial problem. This process is continued iteratively with \mathcal{N} flux iterations performed on each problem between calculations of the cB_i . When the two reactivities agree within a tolerance, $\{$, 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_{T1} .