Contract No. W-7405-eng-26

CHEMICAL TECHNOLOGY DIVISION

MRPP – MULTIREGION PROCESSING PLANT CODE

C. W. Kee and L. E. McNeese

SEPTEMBER 1976

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37830
operated by
UNION CARBIDE CORPORATION
for the
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION

OT W-7405-ENG-26

TABLE OF CONTENTS

MRPP - MULTIREGION PROCESSING PLANT CODE

C. W. Kee and L. E. McNeese

ABSTRACT

This report describes the machine solution of a
large number ($\sim$ 52,000) of simultaneous linear algebraic
equations in which the unknowns are the concentrations
of nuclides in the fuel salt of a fluid-fueled reactor
(MSBR) having a continuous fuel processing plant. Most
of the equations define concentrations at various points
in the processing plant. The code allows as input a
generalized description of a processing plant flowsheet;
it also performs the iterative adjustment of flowsheet
parameters for determination of concentrations throughout
the flowsheet, and the associated effect of the specified
processing mode on the overall reactor operation.

---

1. INTRODUCTION

In a reactor such as a Molten-Salt Breeder Reactor for which contin-
uous processing is planned, the ability to compare alternate processing
methods is essential in determining the effect of small changes in proces-
sing method on the steady-state operation of a reactor and processing plant.
Because of the degree of interaction between the reactor performance and
that of the processing plant, it is necessary to consider the reactor and
the processing system simultaneously. The short cooling times resulting
from continuous processing cause appreciable decay of nuclides in the
processing system and result in high decay heating rates and time-dependent
chemical compositions. Most processing plant flowsheets of interest, when
coupled closely with a reactor, produce numerous feedback streams that

complicate material balance calculations and lead to accumulation of materials over long time periods. To obtain an accurate representation of the performance of the reactor and processing plant in such cases, a flowsheet must be represented in detail, and a larger number of nuclides than considered previously must be taken into account.[1]

Computer programs have been developed and are discussed that allow a processing plant and the associated reactor to be represented in such a manner that the programs can be used independently to represent either the reactor or processing plant in detail or in combination to obtain a detailed representation of the reactor and processing plant system. Although each code solves a large system of simultaneous linear algebraic equations, because of the different characteristics of the two systems, different methods of solution are used; both methods were adapted for use with sparse matrices. The reactor code uses a Gauss–Seidel iteration to solve approximately 750 equations. The processing plant code solves approximately 52,500 equations. However, a special ordering scheme allows a direct solution in which the equations are considered in blocks of 70 equations each.

The calculations are carried out in an iterative manner between the reactor and the processing plant for determination of flowsheet parameters. These parameters include the molar density of phases in each holdup volume and the distribution coefficients for each mass transfer operation in the flowsheet; this feature allows the simulation of complex steps in a flowsheet by using either an equilibrium or first-order rate mechanism.

Sections 2 through 4 of this report discuss the general problem and the set of equations employed in its solution, the method used for solution of the equations, and the extensions of the code to similar problems. The appendixes essentially form a user's manual and include a description of the subroutines, the input, and the output from both the reactor and processing plant codes. A code listing is available from the authors.

2. EQUATIONS AND ASSUMPTIONS

The multiregion processing plant code is based on a model in which a MSBR and the associated processing plant are represented by separate regions of uniform composition; allowance is made for continuous flow between regions, radioactive decay of materials in each region, and transfer of materials between regions in order to represent common mass transfer effects. Allowance is also made for fission and neutron capture reactions in regions representing the reactor. The equations which describe regions in the processing plant are similar to those for the reactor; however, the reactor is represented by a separate code, MATADOR,[2] which receives input from a single region of the processing plant (the region from which processed salt is returned to the reactor). The equations for the processing plant are described separately from those for the reactor to accommodate the different assumptions that are required.

2.1 Model for the Reactor

A previously developed computer program, MATADOR, is used for cal-
culation of the concentrations of nuclides in the primary circuit of a
MSBR under steady-state conditions in which fuel salt is continuously
circulated between the reactor and a processing system. As such, the pro-
gram serves as a subroutine that calculates the input for the processing
plant program (in the form of concentrations and flow rate of the salt to
be processed) based on output from the processing plant (in the form of
concentrations and flow rate of the processed salt that is returned to
the reactor). From the standpoint of the processing plant program, the
reactor is treated as a single region; however, the reactor program treats
the reactor as a multiregion system. The equations permit the use of
terms which are not necessary for every region; for example, flow of
material between any two regions can be specified, but this seldom occurs.
The equations have been described previously,[2] although much of that
description is repeated here.

The accumulation rate of species i in the fuel salt in the reactor
is the input rate of species i by feed, fission, radioactive decay, and
neutron capture in the fuel salt, graphite, and circulating bubbles minus
the disappearance rate of species i due to radioactive decay, neutron
capture, deposition on exchanger surfaces, and chemical processing. At
steady state, this rate of accumulation is zero, so that:

$$0 = \sum_j \left(Ve_{ij} \lambda_j + \phi V_c f_{ij} \sigma_j + A_g g_{ij} + A_b h_{ij} + s_{ij} + \phi V_c y_{ij} \sigma_{fj} \right) c_j$$

$$+ Fc_{io} - \left(\lambda_i V + \sigma_i \phi V_c + F + A_g G_i + A_b H_i + S_i + P_i \right) c_i, \qquad (1)$$

where

A_b = surface area of circulating bubbles, cm^2,

A_g = surface area of graphite, cm^2,

F = volumetric flow rate of fuel salt to the reactor, cc/sec,

G_i = coefficient for loss of species i by diffusion into graphite, cm/sec,

H_i = coefficient for loss of species i by migration to bubbles, cm/sec,

P_i = effective chemical processing rate for species i, cc/sec,

S_i = rate at which species i plates on the heat exchanger surfaces, cm/sec,

V = volume of fuel salt, cc,

V_c = volume of fuel salt in core, cc,

c_i = concentration of species i, moles/cc,

c_{io} = feed concentration of species i, moles/cc,

e_{ij} = fraction of disintegrations by species j which leads to formation of species i,

f_{ij} = fraction of neutron captures by species j which leads to formation of species i,

g_{ij} = coefficient for production of species i by diffusion of species j into graphite, cm/sec,

h_{ij} = coefficient for production of species i by migration of species j to gas bubbles, cm/sec,

s_{ij} = rate of production of species i from species j plated on the heat exchanger surfaces, cc/sec,

y_{ij} = fission yield of species i from fission of species j,

$$\lambda_i \quad = \quad \text{radioactive disintegration constant of species i, sec}^{-1},$$

$$\sigma_i \quad = \quad \text{average neutron-capture cross section of species i, cm}^2,$$

$$\sigma_{fi} \quad = \quad \text{average fission cross section of species i, cm}^2,$$

$$\phi \quad = \quad \text{average neutron flux, cm}^{-2} \text{ sec}^{-1}.$$

Thus, for I nuclides, this equation is a system of I simultaneous algebraic equations and I unknowns. Two other sets of I equations and I unknowns are considered by: (1) allowing for a volume of gas bubbles and a holdup for materials plated out in the reactor flux, and (2) allowing a region in which the evolving gas bubbles and the materials plated out outside the reactor core are held in contact with fuel salt so that soluble decay products may be returned to the reactor.

All three sets of I equations are solved by the Gauss-Seidel successive substitution algorithm, with iteration occurring over each of the successive sets of I equations. Because of the size of the diagonal terms, the system of equations converges rapidly. Direct solutions were not used because of the storage required for remembering a coefficient matrix.

The treatment of diffusion of noble gases into and decay products out of the graphite uses a model developed by Kedl and Houtzeel.[3] This model assumes that the graphite moderator is replaced by semi-infinite solid cylinders with the same surface-to-volume ratio, and that the graphite is coated with a low permeability material to a depth of 1 mil. Diffusion of noble gases into the graphite occurs through a liquid film and the coating, and is simulated by a lumped resistance model. The model developed for the migration of noble gases and noble metals to circulating helium bubbles is an extension of the model proposed by Kedl and Houtzeel.[3] Both of these models are described in ref. 2.

Once the set of concentrations is obtained, it is possible to cal-
culate heat generation rates, fission product poisoning, production rates
of the materials in the reactor, inventory in moles of the material in
the reactor, the uranium mole fraction, and the breeding ratio. The
breeding ratio is calculated by assuming that it varies linearly with
small changes in the fission product poisoning. Thus, the breeding ratio
is determined from the difference between the reference fission product
poisoning (from ROD[4] calculations) and the calculated fission product
poisoning.

In calculating the fission product poisoning, the poisoning from
^{135}Xe is excluded because it was also excluded in the reactor design code,
ROD, where it was always assumed to be one-half of 1%. While MATADOR
calculations of fission product poisoning include absorptions by the
noble gases and noble metals in the gas bubbles, noble metals plated
onto surfaces outside the reactor core are not assumed to absorb neutrons.
Neptunium absorptions are included, because the reference fission product
poisoning used by ROD from previous calculations included the neptunium
absorptions with fission product poisoning.

2.2 Model for the Processing Plant

The processing plant code calculates steady-state concentrations
based on material balance equations, and constant reactor effluent concen-
trations as determined by the last MATADOR calculation. The multiregion
code assumes that the processing plant is composed of a number of regions
(or holdup volumes) connected by flowing streams. A region consists of
a well-mixed volume containing one or two phases in equilibrium. The

equilibrium is specified by a proportionality constant that varies with atomic number and may be changed between successive flowsheet calculations. Any two regions may be linked by flowing streams or by rate expressions, and each region may have feeds or discards. (These are streams which do not connect two regions in the plant.)

2.2.1 Material balance equations

The rate of accumulation of nuclide i in a region (0 at steady state) is the rate of appearance of i in the region from feed streams, from flowing streams from other regions, from mass transfer from other regions, and from production of other nuclides by radioactive decay minus the rate of disappearance of i from the region by flow out, by mass transfer to other regions, and by radioactive decay. Thus, at steady state in region n:

$$0 = \sum_{\substack{j \\ j \neq i}} \lambda_j e_{ij} (V_{S,n} + K_{j,n} V_{B,n}) c_{j,n} \qquad (2)$$

$$+ \sum_{\substack{m \\ m \neq n}} F_{Sm,n}\, c_{i,m} + \sum_{\substack{m \\ m \neq n}} F_{Bm,n}\, K_{i,m}\, c_{i,m} + F_{Ii,n}$$

$$+ \sum_{\substack{m \\ m \neq n}} \left[(k_i a)_{m,n}\, c_{i,m} - (k_i a)_{n,m}\, c_{i,n} \right]$$

$$- \lambda_i (V_{S,n} + K_{i,n}\, V_{B,n})\, c_{i,n}$$

$$- c_{i,n} \left(\sum_{\substack{m \\ m \neq n}} F_{Sn,m} + D_{Sn} + K_{i,n} \sum_{\substack{m \\ m \neq n}} F_{Bn,m} + K_{i,n} + D_{Bn} \right)$$

where

D_{Sn} = discard rate of phase S from region n, cm^3/sec,

$F_{Sm,n}$ = flow rate of phase S from region m to region n, cm^3/sec,

$F_{Ii,n}$ = feed rate of nuclide i to region n, moles/sec,

$K_{i,n}$ = distribution ratio for nuclide i in region n,

V_{Sn} = volume of phase S in region n, cc,

$c_{i,n}$ = concentration of nuclide i in first phase of region n, moles/cc,

e_{ij} = fractions of disintegrations of species j which lead to formation of species i,

$(k_i a)_{m,n}$ = mass transfer rate constant for transfer of species i from region m to region n, cm^3/sec [the ratio of $(k_i a)_{m,n}$ and $(k_i a)_{n,m}$ is a distribution function, as discussed in Sect. 2.2.2],

λ_i = decay constant for nuclide i, sec^{-1}, and

subscripts B and S = phases

i and j = nuclides

m and n = regions.

Hence, there is one equation and one unknown for each nuclide in each region. There are 52,500 equations for 750 nuclides in 70 regions. Because of the number of equations and the number and size of recycle streams present in a flowsheet, the Gauss-Seidel iteration is too time consuming. Since neutron captures are neglected, it is possible to arrange the nuclides so that each nuclide precedes all of its decay daughters. For N regions, there is one set of N equations with N unknowns

for each nuclide. By arranging the nuclides in the proper order and solving each set of equations, a direct solution for the entire set of equations is obtained. Solutions in the processing plant are alternated with calls to MATADOR to obtain a converged solution for a reactor coupled with continuous processing.

The program used has been designed to solve a large system of equations in which the nontriangular coefficient matrix may be expressed as a lower triangular coefficient matrix whose elements are matrices. Each matrix on the diagonal is solved directly, with substitutions being made for unknowns that were calculated previously. In this case, the lower triangular matrix has off-diagonal terms in row i and column j of the form:

$$- \lambda_j \, e_{ij} \, \underset{\approx}{V}_j \, ,$$

where

$\underset{\approx}{V}_j$ = diagonal matrix with the nth term on the diagonal being

$$V_{Sn} + K_{j,n} \, V_{Bn} \, .$$

The terms on the diagonal (i=j) are matrices of the form:

$$\underset{\approx}{R}_i + \lambda_i \, \underset{\approx}{V}_i \, .$$

$\underset{\approx}{R}_i$ is a matrix whose element in row n and column m is

$$(\delta_{n,m} - 1) \left[F_{Sm,n} + K_{im} \, F_{Bm,n} + (k_i a)_{m,n} \right] + \left\{ \underset{m}{\Sigma} \, \delta_{n,m} \left[F_{Sn,m} + K_{i,n} \, F_{Bn,m} \right. \right.$$

$$\left. \left. + (k_i a)_{n,m} \right] + D_{Sn} + K_{i,n} \, D_{Bn} \right\} \, ,$$

where

$\delta_{n,m}$ = 1 if n=m, and

$\delta_{n,m}$ = 0 if n≠m.

The resulting system of equations, when used with a constant vector indicating feed streams, is the set of Eqs. (2).

2.2.2 Mass transfer coefficients

Three models were used for mass transfer between liquid and gas phases. For transfer of noble gases to gas bubbles, the overall liquid mass transfer coefficient was estimated from the Hughmark correlation given by Schaftlein and Russell.[5] This is substituted into the relation for mass transfer:

$$N_i = K_{OL}A_B(c_\ell - Hc_g) = K_{OL}A_B c_\ell - K_{OL}A_B H\ c_g,$$

where

A_B = bubble surface area, cm^2,

H = Henry's law constant, moles/cc (liq) per mole/cc (gas),

K_{OL} = overall liquid mass transfer coefficient, cm/sec,

N_i = rate of mass transfer, moles/sec,

c_ℓ and c_g = concentrations in liquid and gas, moles/cc.

This equation is broken into two contributions to matrix terms analogous to a flow rate so that:

$$(k_i a)_{n,m} = K_{OL}\ A_B$$

$$(k_i a)_{m,n} = K_{OL}\ A_B\ H = H(k_i a)_{n,m},$$

where region n is the liquid phase, and region m the gas phase.

In addition to this transfer, the rate of mass transfer at the surface between the liquid and cover gas must also be considered. For volatile components, the mass transfer resistance is assumed to be across a liquid film of thickness X_ℓ:

$$N_i = \frac{D_\ell A_s}{X_\ell} (c_\ell - Hc_g) = \frac{D_\ell A_s}{X_\ell} c_\ell - \frac{D_\ell A_s}{X_\ell} H c_g \, , \tag{5}$$

where

D_ℓ = diffusivity in liquid, cm^2/sec,

A_s = surface area, cm^2;

thus,

$$(k_i a)_{n,m} = (D_\ell A_s)/X_\ell$$

$$(k_i a)_{m,n} = (D_\ell A_s H)/X_\ell = H(k_i a)_{n,m} \, . \tag{6}$$

The same model is used to describe transfer of nonvolatile decay products of volatile fission products across a gas film of thickness X_g; however, since the transfer from liquid to gas is zero, there is only one term. All of these terms must be added to obtain the matrix coefficients that are used.

In both models the design criterion is based on some consideration other than the removal of volatile nuclides. For example, the gas rate might be based on the amount of reductant necessary for a hydrofluorinator or a hydrogen-sparged vessel, or on the amount of argon needed for a level probe in a surge tank. In a fluorinator, however, the design is chosen to achieve a specified performance such as per cent removal of uranium. For zero inlet concentration in the gas, a material balance gives:

$$F_L c_I = F_G c_G + F_L c_L = P_R F_L c_I + F_L c_L,$$

or

$$c_L = (1 - P_R) c_I; \quad c_I = \frac{c_L}{(1 - P_R)} \, , \tag{7}$$

where

F_L, F_G = flow rates in liquid and gas, cc/sec,

c_I, c_L, c_G = concentrations in inlet liquid, outlet liquid, and outlet gas, moles/cc,

P_R = percent removal;

thus, the transfer across the interface is $P_R F_L c_I$. Substituting for c_I:

$$N_i = F_L \frac{P_R}{(1 - P_R)} c_L. \tag{8}$$

In the simulation, F_L was included as a parameter independent of the element, and $P_R/(1 - P_R)$ was listed as a constant that depended on the element number.

3. NUMERICAL METHODS EMPLOYED

3.1 Solution of Reactor Material Balance Equations

The number of equations to be solved by the reactor code is equal to the number of nuclides, ≈ 739. The coupling of the equations used for the salt in the drain tank with those used for the gas bubbles adds two additional sets of the same number of equations. The direct solution of such a set of equations would require a method that would take advantage of sparseness and might need only limited pivoting to minimize fill. These problems are eliminated by the use of an iterative solution, providing the solution is obtained in a reasonable number of iterations. The number of iterations has always been less than 30.

3.2 Solution of the Processing Plant Material Balance Equations

The number of equations and unknowns for a flowsheet of 70 regions
and a nuclear library of 739 nuclides is about 52,000. Storage of the
coefficient matrix of this system of equations is completely impractical,
and storage of even the nonzero terms could easily exceed the storage
capacity of machines available; therefore, a direct Gauss reduction
could not be used. Various approaches were made by using a Gauss-Seidel
iteration. The best method was found to be a series of solutions for
all nuclide concentrations in the flowsheet; the nuclides were considered
one at a time so that some nuclides required only a few iterations. Even
with the rearrangement of nuclides, which necessitated only one solution
for each species, the time requirements were still excessive; this was
because the diagonal element of the coefficient matrices was comparable
in magnitude to the off-diagonal terms, except for nuclides with short
half-lives.

For each successive nuclide, a 70 by 70 matrix was solved to obtain
the concentration of that nuclide in each of 70 regions. The time
requirement for a direct solution for all concentrations in the processing
plant by this method was quite reasonable - less than a minute for the
IBM 360/91. In addition, the memory requirements were essentially the
same as those for the iteration technique. The storage of the solution
vector is a significant fraction of the storage requirement, and for
larger flowsheets (the IBM 360/91 at ORNL was able to handle 250 regions)
the solution vector uses most of the storage.

3.3 Iteration with Reactor Code

The existence of this reasonably rapid solution for the processing plant calculation enabled iterations with the MATADOR routine which simulated the reactor performance. MATADOR was coupled to the processing plant calculation by using removal times defined as the reactor inventory of a species divided by its net removal rate by chemical processing. Fairly rapid convergence of concentrations was achieved for most nuclides if the parameters that passed between routines were averaged with their previous values to provide damping. All concentrations converged within 20 iterations. However, the variation of more parameters between iterations, and the consideration of more difficult flowsheets required improvement to the code. The use of reactor inlet concentrations rather than removal times resulted in comparable performance, but permitted a different set of nuclides to converge rapidly. Hence, the concentrations in the stream returning to the reactor can be described as the sum of the amount remaining after processing and the amount due to production in the processing plant.

At the same time the solution is found for the processing plant concentrations, a solution is also found for the first derivative of the reactor inlet concentration with respect to the outlet concentration of the same nuclide. This is possible because the same coefficient matrix is used for both sets of equations. The constant vector in the solution for the derivative is a vector with unit concentration of each nuclide in the reactor and no terms for production by decay. Hence, with little increase in calculational effort and complexity, provision

can be made for small changes in concentration to allow the reactor
code to predict the processing plant performance. In addition, an even
greater saving of time is made by considering only those nuclides whose
concentrations are slowly converging, and by periodically reconsidering
all nuclides.

3.4 Calculation of Molar Volumes

The steady-state performance of a processing plant depends on the
rates of decay and, therefore, the molar inventories of radioactive
nuclides throughout the processing plant. It is thus necessary for the
molar inventories and molar volumes to be consistent with the volumes
specified. It is also necessary to know the ratio of the molar volumes
of the two phases in certain regions for use as a conversion factor to
convert the distribution coefficients from ratios of mole fraction to
the ratio of concentrations in moles per cubic centimeter. This problem
is not alleviated by use of molar flow rates and mole fractions, because
the parameters must be such that the mole fractions add up to 1.0.

Between iterations, each flow rate from a region is expressed as
a fraction of the total flow from that region. The molar volume of each
stream for which a molar volume correction is to be made is determined
by assuming additive molar volumes. The corrected flow rate is given
by the product of this molar volume; the molar flow rate and the ratio
of molar volumes for regions containing salt and bismuth provides a
conversion factor for the distribution coefficients.

3.5 Correction of Distribution Coefficients

For a given region n, the set of concentrations of that region is the solution of:

$$\left[F_S + \lambda_i V_{S,n} + K_{i,n}(F_B + \lambda_i V_{B,n}) \right] c_{i,n} = \sum_j \lambda_j e_{ij} (V_{S,n} + K_{j,n} V_{B,n}) c_{j,n}$$
$$+ (INPUT)_{i,n} \tag{9}$$

where

$$F_S = \sum F_{Sn,m} + D_n, \text{ and}$$

$(INPUT)_{i,n}$ = total input of nuclide i to region n from feeds or other

regions.

From a table of valences, it is possible to calculate an equivalent density in the second phase in equivalents per cubic centimeter. It is this number which must remain constant through any series of reductive extraction operations to satisfy the condition of an equivalent balance. The equivalent density, E_o, in the second phase that enters the region may be a calculated variable or an input variable. In either case, the variable $D_{Li} = K_{Li,n}$ must be determined so that

$$E - E_o = 0,$$

where

$$E = \sum_i E_{i,n} = \sum_i (VAL)_i K_{i,n} c_{i,n} = \text{total number of equivalents per}$$

milliliter in the bismuth phase, and

$(VAL)_i$ = valence of i in salt.

We have also defined:

$$K_{i,n} = A_c D_{Li}^{A_n}, \tag{10}$$

since

$$\log K_{i,n} = A_n \log D_{Li} + A_c, \tag{11}$$

where A_n and A_c are constants.

The proper value of D_{Li} can be found iteratively by Newton's method if we have a means for evaluating

$$\frac{d}{dD_{Li}} (E - E_o).$$

Although the specific derivative may be found, an approximation is given because it requires less time and memory, involves terms all of the same sign, and still arrives at the proper D_{Li}. It is convenient to redefine the constant terms:

$$A_1 = F_S + \lambda_i V_{S,n}$$

$$A_2 = F_B + \lambda_i V_{B,n}$$

$$(PROD)_i = \sum_j \lambda_j e_{ij} (V_{S,n} + K_{j,n} V_{B,n}) c_{j,n} + (INPUT)_{i,n},$$

so that

$$c_{i,n} = \frac{(PROD)_i}{A_1 + K_{i,n} A_2}$$

$$E_{i,n} = \frac{(PROD)_i K_{i,n} (VAL)_i}{A_1 + K_{i,n} A_2}. \tag{12}$$

By obtaining the derivative of $E - E_o$

$$\frac{d}{dD_{Li}} (E - E_o) = \frac{d}{dD_{Li}} E = \sum_i \frac{d}{dD_{Li}} (E_{i,n}) \tag{13}$$

from Eq. (12):

$$\frac{dE_{i,n}}{dD_{Li}} = (PROD)_i (VAL)_i \left[\frac{1}{A_1 + K_{i,n}A_2} - \frac{K_{i,n}A_2}{(A_1 + K_{i,n}A_2)^2} \right] \frac{dK_{i,n}}{dD_{Li}} \quad , \qquad (14)$$

assuming that (PROD) is independent of D_{Li}. While this assumption is

$$\frac{d\ K_{i,n}}{d\ D_{Li}} = \frac{A_n A_c D_{Li}^{A_n}}{D_{Li}} = \frac{A_n}{D_{Li}} K_{i,n} \qquad (15)$$

for

$$D_{Li} \neq 0 \ ,$$

then

$$\frac{dE_{i,n}}{dD_{Li}} = \frac{(PROD)_i (VAL)_i\ K_{i,n}}{A_1 + K_{i,n}A_2} \left(\frac{A_1}{A_1 + K_{i,n}A_2} \right) \frac{A_n}{D_{Li}} =$$

$$E_{i,n} \left(\frac{A_1}{A_1 + K_{i,n}A_2} \right) \frac{A_n}{D_{Li}} \quad . \qquad (16)$$

Thus, as calculations are made for $E = \Sigma\ E_{i,n}$, calculations are easily made for

$$\frac{dE}{dD_{Li}} = \sum_i \frac{d}{dD_{Li}} (E_{i,n}) \ . \qquad (17)$$

The proper D_{Li} for use in the next iteration can be obtained by allowing convergence of

$$(D_{Li})_{m+1} = (D_{Li})_m + (E - E_o) / \frac{d}{dD_{Li}} (E - E_o) \ . \qquad (18)$$

Since $(PROD)_i$ is dependent upon D_{Li}, two terms have been left out. Some provision must also be made for the failure of this scheme to converge. The calculation procedure examines the sign of $E - E_o$ in order to contin-uously specify upper and lower bounds for D_{Li}. Any new D_{Li} value that is outside this interval is replaced with the value at the midpoint of the interval. This value of D_{Li} might tend to overcorrect in a flowsheet,

especially if these regions are in series. As a means of providing a damping effect, the value returned for the next iteration is a weighted average of this value and the old value.

3.6 Investigation of Faster Solution Methods

When flowsheet parameters such as flow rates (i.e., molar volume) and distribution constants are not changed between iterations, the solution for all but the first iteration can be speeded up. The solution by reduction and back substitution is a series of row operations on an augmented matrix; much of the calculational time is used to perform operations on the coefficient matrix. For double precision calculations, the information required for making the necessary row operation on only the constant vector is a double precision constant and two indexes which indicate the rows involved. This information can be stored easily in the eliminated matrix positions during the first iteration so that they may be used on subsequent iterations. The calculations after the first iteration are manipulations performed on a vector rather than operations on a matrix; in addition, the matrix, which is not required, need not be determined in these subsequent iterations.

Before the introduction of changing flowsheet parameters, the number of row operations required for a 50 by 50 flowsheet matrix was determined. Between 200 and 280 row operations were performed with about 80% of the calculational time required for operations on the coefficient matrix. The number of constants needed for all nuclides requires more memory than is available as fast memory; however, because the constants need only be accessed sequentially, storage on a direct access device is sufficient, and only a small buffer space is required.

4. LIMITATIONS AND SPECIAL CONSIDERATIONS

4.1 Limitations of Steady State Calculation

A number of problems arise from the consideration of a steady state process (i.e., flowsheet steps that are designed to be intermittent). A good example is a waste tank that is slowly filled with waste salt over a period of about 1 year, after which the salt is held up for a decay period and fluorinated for uranium recovery. In addition, a number of materials are accumulated over the lifetime of a reactor. As a result, some materials must be removed by a discard stream so that the residence time is about one reactor lifetime if the nuclides (e.g., lead) are to be discarded.

It is also desirable to treat some nuclides (e.g., transuranium isotopes) in a steady state concentration, even though steady state requires several reactor lifetimes to achieve. Although the code makes a steady-state material balance calculation, the convergence is slowed down by this kind of calculation, because the code requires time to permit the concentrations of this material to build up during successive iterations.

A steady state calculation for processing plants with near total recycle of any species requires user caution. The existence of any material that cannot be removed from some region or group of regions causes the system of equations to be singular, since the concentration of this material in those regions is undefined. Iterations with the reactor code do not alter the concentrations of major salt components, and provision is made in the processing plant code for the concentration of any nuclide to be defined by the input at any point.

4.2 System of Units

In the system of units that was used, the code calculated concentrations in moles per cubic centimeter by using flow rates in cubic centimeters per second, volumes in cubic centimeters, etc.; however, this is not necessarily implied by the equations, since they are just as valid in other systems of units (the nuclear library uses seconds as the unit of time). The most logical alternate set of units is the description of volume in moles, rates in moles per second, etc., which results in concentrations in mole fraction. Some calculations might be changed by the user who prefers this system of units. For example, the calculation of mole fractions is redundant and might be replaced by the calculation of concentrations in moles per cubic centimeter. The greatest alteration required in such a change of units is the replacement of the output headings. By such label changes, it would be possible to treat any system of equations of this form.

4.3 Description of Particular Items Using Mass Transfer Coefficients

In most simulations, the use of mass transfer coefficients was limited to gas-liquid contacts requiring one region for the gas phase and one region for the liquid phase containing only one or two liquids. It was assumed that the noble gas concentrations were small enough so that no appreciable error would occur by treating the gas bubbles as if they had the same concentrations as the bulk gas. This is not an essential assumption, however, since by using more regions, the same case might be described as having more than one gas region. The bulk gas

would be a separate region from the gas bubbles, and the gas bubbles
in separate liquids could easily be separate regions. If the gas con-
centration changes as it rises through the liquid and is not small
relative to its equilibrium concentration, the gas at various levels
in the liquid would be in different regions and would represent a
column.

Careful consideration has also been given to simulation of mass
transfer-limited transfer rates as a replacement of equilibrium stages
in a liquid-liquid contactor. For turbulent flow when transfer is lim-
ited by eddy diffusivity, the only difficulty arises in specification
of the distribution coefficients at the interface. This case can be
described by three regions: the bulk liquid phase S, the bulk liquid
phase B, and the interface contact of both phases. The flows between
the interface region and the bulk liquid region are the rates of eddy
transport in each phase. The product of eddy diffusivity and concentra-
tion driving force in each phase is implicitly obtained in two flow rate
terms for each phase in the same manner described in the section on mass
transfer coefficients. In this case, the distribution coefficients can
still be determined iteratively by the same technique used for equilib-
rium stages even though the distribution coefficients are not expected
to be the same.

4.4 Uses Requiring Modifications

It is possible to consider flowsheets involving batch processes
that might not be represented by either an equilibrium process or a rate
limited process, because the amount of material transferred between the

phases was dependent on the concentrations of other elements as well (e.g., oxide precipitation). The rate coefficients that are used have always remained constant, but this is not essential to the code since iteration is required to converge other parameters in the flowsheet. By using a reasonable estimate for percent removal (i.e., rate constants) for various components, a set of concentrations would be obtained. Subroutine VOLUME would then be used to calculate the proper removal rates on the basis of these concentrations, and it would modify the rate constants accordingly for the next iteration. This system for treating more general processing steps should cause no greater convergence problem than the iterative determination of distribution parameters described later.

An alternative to this approach is to provide concentration or feed rate links to such a subroutine as is done with MATADOR, or even to replace MATADOR with a routine simulating some section of the flowsheet. However, the most likely substitution for MATADOR is either a subroutine that reads and stores entering processing plant concentrations or one that simulates a different reactor type. In this last case, the reactor and processing plant need not be linked directly and processing need not be continuous (i.e., there can be batch replacements of fuel and a decay period); therefore, the steady state performance of a system of reactors and processing plants can be obtained.

5. REFERENCES

1. J. S. Watson, L. E. McNeese, and W. L. Carter, MSR Program Semiannu. Progr. Rep. Aug. 31, 1967, ORNL-4191, pp. 245-47.

2. M. J. Bell and L. E. McNeese, Engineering Development Studies for Molten-Salt Breeder Reactor Processing No. 1, ORNL/TM-3053 (November 1970), pp. 38-48.

3. R. J. Kedl and A. Houtzeel, Development of a Model for Computing ^{135}Xe Migration in the MSRE, ORNL-4069 (June 1967).

4. H. F. Bauman et al., ROD: A Nuclear and Fuel-Cycle Analysis Code for Circulating-Fuel Reactors, ORNL/TM-3359 (September 1971).

5. R. W. Schaftlein and T. W. F. Russell, "Two-Phase Reactor Design," Ind. Eng. Chem. 60(5), 12-27 (1968).

6. W. L. Carter, ORNL, personal communication, June 1972.

7. M. J. Bell, ORIGEN - The ORNL Isotope Generation and Depletion Code, ORNL-4628 (May 1973).
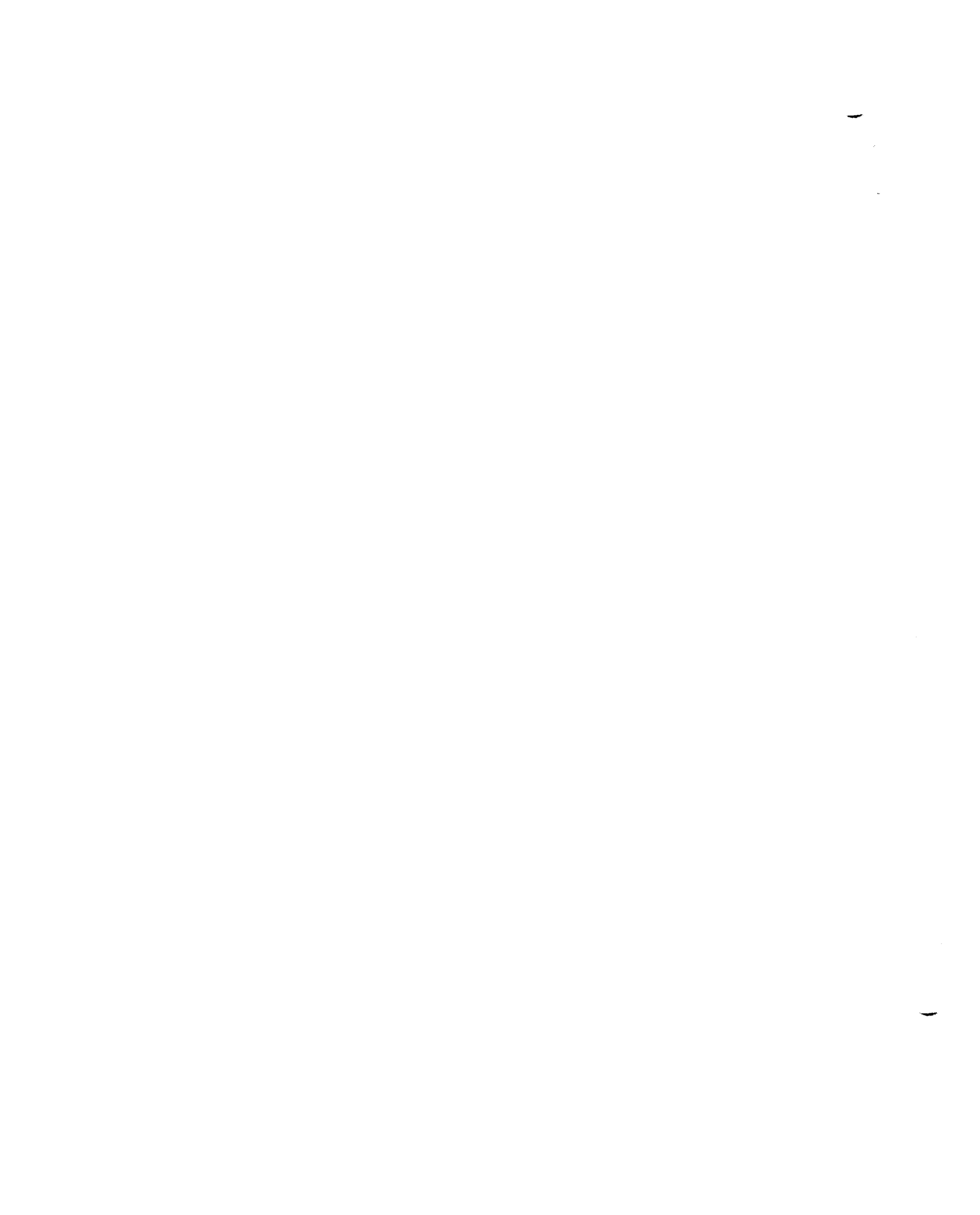
APPENDIXES

USER'S MANUAL

APPENDIX A: DESCRIPTION OF SUBROUTINES USED

A description of each routine used in the program is given in the
approximate order of use. Sufficient information is available to permit
use of the program and some modification by users. In particular, the
user should be able to utilize the reactor subroutines independently
of the processing plant code by supplying any necessary information and
a main program which calls MATADOR (with the BLOCK DATA routine described
in the processing plant code).

Subroutine MATADOR

MATADOR directs all the reactor calculations. It begins by reading
the input defining the reactor and the variables that will provide an
initial guess for process plant removal rates. Through several calls
to GRAPHT, it sets up the coefficient matrix for nuclear transitions
in the graphite. Whereas this matrix is stored separately from the
corresponding matrix for transitions in the salt, the processing plant
code can use the matrix for decay in the salt, and the matrix for graphite
may be changed between iterations when the code is used iteratively with
ROD.

MATADOR begins by setting up the constant vector and the diagonal
(the inverse of the diagonal elements is calculated to save calculational
time later on) for the calculation of the reactor salt concentrations.
While the variables XIN and DXIN are zero for the first call to MATADOR,
the chemical removal rate of each nuclide is assumed to be proportional
to the reactor concentration. Hence, chemical processing is specified
in the diagonal element.

Two additional options are available on subsequent calls. First, it may be assumed that the flow rate of a nuclide into the reactor (XIN) is independent of its concentration in the reactor. Second, it may be assumed that the flow rate of material into the reactor is a linear function of the reactor concentration. In this case, DXIN and the derivative of XIN with respect to reactor concentration are nonzero. In considering chemical processing, it was necessary to allow negative efficiencies to adequately treat those nuclides produced by decay in the processing plant that were removed primarily by neutron captures in the reactor. Accordingly, care is taken to ensure that the diagonal elements do not approach zero. A message is printed if the diagonal element reaches some predefined limiting value.

The program obtains a solution for the reactor salt with a first estimate being made for only the first call to MATADOR. Solutions are then obtained for the noble gases and noble metals that are in contact with salt both inside and outside the reactor. Noble gases leaving the holdup region inside the reactor are sent to the region outside the reactor. The diagonal elements and the constant vectors are defined so that the calculations provide the holdup in moles for noble gases and noble metals, and the flow rate into the salt in moles per second for all other materials. The calculation actually assumes that the returning nuclides are held up for 1 sec during which they may decay or capture neutrons. In the calculations of the transfer rates from these holdup volumes to the salt, the three systems of I equations are weakly coupled and are assumed to converge in three passes. This may not be true for long holdup times in these phases; therefore, the values are remembered

between calls to MATADOR so that each call improves the value of these transfer rates.

Several output parameters are calculated before calling KEE, which controls output; these are fission product poisoning per fissile absorption, nuclide production rates, molar volume, uranium mole fraction, and breeding ratio. If the input variable KARD is nonzero, punched output is obtained. If KARD is positive, the concentrations are in a format for input to the CALDRON[6] code; if KARD is negative, the concentrations are in a format for input to ORIGEN.[2] The ORIGEN input can also be prepared by the calling program so that a modified ORIGEN code may be used as a second job step.

Subroutine AMATRX

The main function of AMATRX is the construction of the matrix A, which contains decay and capture rates for all transitions between isotopes, except fission. The subroutine reads the data in the format described in the section on input and writes a table summarizing the nuclear library. Nuclides are identified by the value of 10,000 times the atomic number plus 10 times the molecular weight (plus 1 for excited state). It stores the identification of all decay daughters and capture products in a matrix NPROD with the corresponding production rates in COEFF. It also stores the parent nuclide identification, NUCL, the total decay rate, DIS, and the capture cross section, TOCAP, as well as heat generation rates, the fraction of heat which is gamma energy, fission product yields, etc. Thermal, resonant, and fast neutron cross sections are stored so that the program can be used iteratively with ROD by

allowing corrections based on the spectral factors determined by ROD to be made to the cross section data for the next MATADOR calculation.

The program then constructs the vector A (C in this subroutine), which contains the value of all nonzero production rates, and the corresponding vector LOC, which contains the ordinal number of the parent nuclide, assuming that the daughter nuclides are in the same order of the NUCLs. These constants are stored in the order of production by decay, and by thermal, resonant, or fast neutron captures. For each nuclide there are variables KD, KTH, KRI, and NONO which store the cumulative number of terms for that nuclide with the sum of all the NONOs being stored in NON. In these variables only members of a given group produce nuclides in that group. Similarly, there is a set of vectors indicating the production rate of fission products from the fission of actinides. These variables are denoted by F, LOF, KF, and NOF.

A vector NSTAR is constructed so that for any NUCL with atomic mass NMASS, atomic number NATNO, and isomeric state (0 or 1) NISOM:

$$NSTAR = NTEMP + NATNO*10$$

for positron emitters, and

$$NSTAR = NTEMP - NATNO*10$$

for all others, where

$$NTEMP = NMASS*10000 + 1000 + NISOM.$$

A vector IO is constructed which contains the ordinal numbers of all the nuclides in decreasing order of NSTAR. This is in decreasing order of atomic mass with the ordinal number of excited state nuclides appearing before that of the corresponding ground state nuclide. The sequence

includes positron emitters with decreasing atomic numbers for a given

atomic mass, followed by other nuclides with increasing atomic numbers.

This sequence includes all nuclides before any of their decay daughters

as long as all decays represent a loss of energy, mass, or reduction in

$E + MC^2$, rather than a capture of some kind. This sequence is used

later in the processing plant code as the order of calculation.

The variables are dimensioned so that as many as 800 nuclides can

be included, of which as many as 100 may be actinides; in addition,

there may be as many as 1500 production rates of nuclides from other

nuclides by decay or neutron absorption. The nuclear library contains

739 nuclides, 99 actinides, and 1466 production rate terms.

Subroutine GRAPHT (DIS1, CAP1, DIS2,CAP2,COEFF,N,I,P2,VR,DEP,D1,P1,FLUX2)

The subroutine GRAPHT calculates rate coefficients for fission

product deposition in graphite based on the diffusion model described

in ref. 2, and the variable names are analogous: EN=n, EL=1, $ENSQ=n^2$,

etc. The calculations are made to determine FLUX2, which is returned

to be inserted in a matrix G, and is analogous to the matrix A (described

with AMATRX). This value represents the contribution to the concentra-

tion of species i from diffusion into graphite of species j $(A_g \, g_{ij}$, as

described earlier). The adjusted diagonal matrix element for the parent

isotope (G_j), the contribution to poisoning by both parent and daughter

isotopes in the graphite, and the deposition rates of nonvolatile daugh-

ters are also calculated.

GRAPHT also uses the variables in common block GRAFIT that were

read by MATADOR: AREA,VOL,PORTY,FILM,DIFFY,RADIUS, and SOLBTY. These

variables have been described in the section on input. The subroutine

requires a subroutine BESI for computing Bessel functions.

Subroutine BESI(X,N,BI,IER)

This is a library routine which computes the Bessel function, BI,

of order N with argument X, where N and X are greater than or equal to

zero. IER is used as an error indication.

IER   =   meaning

0   =   no error

1   =   N is negative

2   =   X is negative

3   =   underflow, BI.LT.1.E-69,BI set to 0.0

4   =   overflow, X.GT.170 where X.GT.N

Only Bessel functions of orders 0 and 1 are required by GRAPHT.

Subroutine CHEMPL

The value of the reciprocal of the removal time for the group of

which that nuclide is a member is stored in the variable PR(I) (processing

rate) for each nuclide. On the first call, the routine uses the group

removal times (NTIME) that are used for output purposes and the efficiencies

assumed for each element to calculate a removal time for each element in

the units used for that group of elements. This is written in a table

for output. In addition, on the first call the program adds to the

matrix elements for production of uranium from protactinium an amount

which assumes that all protactinium removed by chemical processing decays

to uranium and is returned to the reactor. A call to another entry point

(ENTRY REPAIR) causes these amounts to be subtracted from the proper matrix elements, thus assuming that the link to the processing plant routines takes this into account.

Subroutine GAUSS(XEQL,C,D,*)

GAUSS solves the matrix equation:

$$\underset{\sim}{A}(XEQL) = \underset{\sim}{C} \,,$$

where A is the coefficient matrix, and reciprocals of the diagonal elements are given by D. A nonstandard return is made if the system does not converge within the maximum allowable number of iterations. The groups of terms included in the coefficient matrix are determined by the value of the logical variables REGION, and REG2.

Subroutine KEE

Subroutine KEE ensures that subroutine RESULT (described next) is called if this is the last call to MATADOR as indicated by the logical variable ILLOG. Subroutine KEE prints the number of calls to MATADOR and a convergence message, and calls RESULT at specified intervals.

Subroutine RESULT (XEQL)

RESULT prints output tables for a given set of reactor concentrations (XEQL). It first calculates the number of moles of material in the stream returning to the reactor for use in the calculation of mole fraction for that stream. It then initializes or calculates other important variables such as:

FISSL = the mass of materials lost due to fission,

FISSN = absorption rate by the five fissionable isotopes,

FISSA = absorption rate by fissile isotopes,

DMOLAR = molar density,

TOCAP(IGAS) = the adjusted cross section for noble gases that
reflects the number of absorptions of noble gases
in graphite.

The program calculates the gamma heat rate in the salt, the absorption rate in the salt normalized to absorptions per fission, the removal rate due to chemical processing, and the contribution to radioactivity in the salt for each of the three groups of nuclides (light elements, actinides, and fission products). A series of calls to subroutine SORT is used to identify the 25 most important materials in each category. Calculations for the fission products are somewhat complicated by the contribution of nuclides in the circulating bubbles and by the contribution of nuclides plated onto the reactor surfaces. Once these are printed, the program prints the totals for removal of fission products and actinides, the burnup rate of thorium, and a correction factor based on the nuclear transitions to nuclides not listed in the library. A correction factor not considered is the amount by which the average mass yield from fission, as given in the library, fails to match the mass of the fissionable nuclide less the average number of neutrons emitted. Additional important variables in this subroutine are:

COMPNG = composition of noble gases,

COMPNM = compositions of noble metals,

HEATNM = heat generation rates of noble metals,

COMPBI = composition of materials extracted into bismuth,

COMPRE = composition of rare earths,

COMPF2 = compositions of materials removed from fluorinators,

   primarily the halogens,

COMPLB = compositions of materials removed in group 6 by chemical

   processing,

SRATE = sum of the removal rate for each nuclide group,

SHEAT = sum of the heat generation rates in each group, and

SCAPT = sum of the absorption rates in each group.

MAIN Program

The main program controls the processing plant calculations, and
begins by calling MATADOR for setting up the matrix for decay chains.
It then reads the input describing the flowsheet and prints the tables
of input data. Region names are compared with stream origin and des-
tinations to construct vectors which describe these streams by the
ordinal numbers of the region names used. In addition, total flows and
fractional flows are determined. All streams with a destination not in
the list of region names are assumed to be discard streams and are added
to the loss rates. The program calls subroutine EQKN to define the
equilibrium constants to be used.

The program then alternates calls to MATADOR with calculations of
processing plant concentrations. Processing plant concentrations are
calculated for one nuclide at a time in the order defined by IO to
ensure that decay products follow their precursors. The program defines
a coefficient matrix and two constant vectors for each nuclide before

calling a subroutine MATQD to solve the two sets of equations represented. The first constant vector defines the production rates of the nuclide and results in the list of concentrations. The second vector has its only nonzero value corresponding to the reactor, and it results in the solution of the concentrations for the case of no production by decay in the processing plant; if instructed to do so, it determines the influence of reactor concentration on the effluent from the processing plant.

If the input instructions specify averaging between iterations, the program averages concentrations to and from the reactor, and averages removal times with their previous values. It also checks for convergence (perhaps an ordered list of the reactor inlet concentrations in decreasing order of their relative change in the last iteration) and prints the first 50 in this list. VOLUME is called for possible modification of the flowsheet parameters, and a logical vector is defined that identifies the nuclides that have converged; therefore, they need only be considered on every tenth iteration.

After the calculations have converged, or the program reaches the allowed time limit or number of iterations, a deck of punched cards is prepared. If I2 is greater than zero, the deck includes reactor effluent concentrations and removal efficiencies for all the nuclides. If I6 is greater than zero, much of the description of the processing plant is included. The program then calls OUTO and stops.

BLOCK data. A block data subroutine is used to initialize the 100-element variable ELE to the element symbols, and the variable STA(1) and STA(2) to blank and "M".

Subroutine EQKN

EQKN reads data for distribution constants and assigns values to the distribution coefficient matrix, EQK. The data are selected for characterizing a region by matching an input variable (NS) in the data with the input variable (NSTR) corresponding to the region. For region REGION(N), the distribution coefficient for the element with atomic number NZ is EQK((N-1)*100+NZ). For NS = 1 or NS = 2 the value of NSTR is used to choose between two temperatures. If NS = NSTR(N), the calculation assumes a temperature of 640°C; if NS+20 = NSTR(N), the calculation assumes a temperature of 550°C. Sufficient data is stored in variables IAE, AIE, AKE, and BKE to later allow a different calculation for these distribution coefficients in VOLUME.

Subroutine VOLUME (NLEFT)

On the first call to VOLUME, the input variables are read, each phase is assigned a starting molar density, and each region is assigned a ratio of molar densities for use in converting distribution constants from ratios of mole fraction to ratios of concentrations. Estimates of molar density are found in the values of XVEST and YVEST for first and second phases, respectively. The 16 values in each vector correspond to the values of the two hexadecimal digits of NC which identify the phases present.

On subsequent calls, VOLUME begins by calculating new values of XLIB for those regions with NV.LT.0. The entering equivalent density for a given region is determined by using the tables of valences, and

for specifying the equivalent density in that region if the value supplied on input is less than 0. The equations for concentrations in the region are then solved for each new intermediate lithium distribution XLI, with the equivalent density and the derivative of equivalent density with respect to XLI being determined for each solution. Each new XLI value is then determined by Newton's method, but with XLI confined to continually readjusting narrowing limits based on the sign of the difference of equivalent density and the reference value. A logical variable DECIDE is defined to identify all nuclides that have a significant influence on the derivative of equivalent balance with respect to XLI. NLEFT is the number of regions requiring more than one iteration.

The converged values of XLI are stored in XLIB so that new distribution coefficients can be determined later, just before returning to the main program. Before this calculation is done, however, the molar volume is calculated for all phases indicated by the first digit of NC. This is used to redefine the values for molar density, total flow rate, and ratios of molar densities for use as a conversion factor for the distribution coefficients.

This subroutine is used to change flowsheet parameters between iterations. It could easily be modified to adjust other parameters as well. The most probable adjustable parameter is the rate coefficient table that would be adjusted to match some arbitrary function of processing plant concentrations. This is the method considered for simulating the oxide precipitation flowsheet that involved nonlinearities, nonequilibrium contacts, and batch operations.

Subroutine MATQD(A,X,NR,NV,DET,NA,NX)

MATQD solves a system of linear algebraic equations in double precision, with coefficient matrix A and constant vector X, and returns with the solution vector in X. Any number, NV, of systems of equations with the same coefficient matrix may be solved by including NV constant vectors, each of which occupies the first NR position in consecutive segments in X of length NX; thus,

A = the coefficient matrix, where the element in row I and column J is element number $(J-1)*NA+I$.

X = solution vector and constant vector. The position represented by variable I in the solution set K is $(K-1)*NX+I$.

NR = number of unknowns.

NV = number of solutions required with the matrix A.

DET = returns with value 0.0 for a singular matrix and a value 1.0 for a nonsingular matrix. Originally, DET was the value of the determinant of A.

NA = number of elements in A allowed for each column.

NX = number of elements in X allowed for each set of equations.

Variables A, X, and DET are double precision.

MATQD was originally obtained from the ORNL computer library; however, the solution here involves a sparse matrix. For this reason, modifications were made to have the subroutine check for zeros before performing multiplications and divisions in the coefficient matrix. A vector ISTAR was defined to remember the row numbers of up to 50 rows, with nonzero elements below the pivot element to be eliminated in a

given reduction step. After modification, the routine obtained the solution in one-fifth the time required previously. This modified version has been used to replace the library version in one other instance, and it achieved a solution in one-third the time required previously.

Subroutine OUTO

OUTO supplies input for the modified ORIGEN[7] code used as a second job step and prints all output tables that supply concentrations or heat generation rates. It requires the function NOAH, described previously, for producing alphameric names from nuclide identifications. It also calls the subroutine OUTL.

Subroutine OUTL (REG, RA, RB, XL, YL, NR)

OUTL searches through the loss rates and prints a summary of the loss rates from the processing plant, where

 REG = list of region names,

 RA = list of first-phase names,

 RB = list of second-phase names,

 XL = first-phase loss rates,

 YL = second-phase loss rates, and

 NR = number of regions.

Miscellaneous Short Routines

Subroutine SORT (X,LABEL,Y,NAME,NX,NY)

SORT makes one pass through the vector, Y, and inserts the values of Y and NAME in the vectors X and LABEL so that the X values are in

decreasing order. X and LABEL have dimension NX in subroutine SORT, while Y and NAME have dimension NY.

Subroutine ZERO(A,B,N)

ZERO zeroes the space between address A and B inclusive in units of N bytes.

Subroutine HALF(A.I)

HALF computes the decay constant, A, in units of sec^{-1} from the half-life A in units denoted by I, where I corresponds to IU in the nuclear library.

Function NOAH (NUCLI)

This routine constructs a three-word alphameric symbol for an isotope from its six-digit identifying number, NUCLI. The three words consist of the symbol for the chemical element, the atomic weight, and either a blank or an "M" to designate a ground state or metastable state. These symbols are used only when printing output tables.

APPENDIX B: INPUT

The input description is arranged by subroutine names in the order in which they are called. The input description is further divided so that each type of input card can be identified. The argument list and the format are given for each read statement.

Nuclear Library (AMATRX)

A. NDT: (-I10) used to determine format for nuclear library.

B. card 1: 80 character title; alphameric format

    card 2: ERR, NMO, NDAY, NYR, NGO

    FORMAT(F10.5,4I2)

    ERR = number below which constants (decay constants, etc.) will be assumed to be zero

    NMO, NDAY, NYR = date, month, day, year used as heading

    NGO -- no longer used.

    card 3: NSORS(I),I=1,5

    FORMAT(5I10)

    The six-digit identifying numbers for the five fissionable materials ^{233}U, ^{235}U, ^{232}Th, ^{238}U, and ^{239}Pu. These must be in order, because they refer to columns of data on fission product yields.

The six-digit identifying numbers are made up so that the highest order digits give the atomic number, the next three digits give the atomic mass, and the lowest order digit is 0 for ground state and 1 for excited state. This is the same system used by the ORIGEN[7] code. The next cards

describe the properties of each nuclide in the nuclear library. If the previously read variable, NDT is nonzero, the library is read from unit 7 in the formats given in ref. 7. Unit 7 has usually been a series of disc files. For NDT=0 the library is read on unit 50 in the format and order given here.

Light Elements[*]

NUCL(I), DLAM,IU,FBI,FP,FPI,FT,FA,SIGTH,FNG1,FNA,FNP,RITH,FINA,FINP

SIGMEV,FN2N1,FFNA,FFNP,Q,FG

FORMAT(I6,F5.3,I1,5F5.3,E5.2,3F3.3,E5.2,2F3.3,E5.2,3F3.3,F4.3,F3.3,F6.3)

Ended by NUCL(I)=0

Actinides

NUCL(I),DLAM,IU,FBI,FP,FPI,FT,FA,SIGNA,RING,FNG1,SIGF,RIF,SIGFF,SIGN2N,

FN2N1,Q,FG,SIGN3N,FSF

FORMAT(I6,F5.3,I1,5F3.3,2E5.2,F3.3,4E5.2,F3.3,F4.4,F3.3,2E5.2)

Ended by NUCL(I)=0

SIGN3N, and FSF are not used.

Fission Products[*]

NUCL(I),DLAM,IU,FBI,FP,FPI,FT,SIGNG,RING,FING1,Y23,Y25,Y02,Y28,Y49,Q,FG

FORMAT(I6,F5.3,I1,3X,4F3.3,2E5.2,F3.3,5E5.2,F4.3,F3.3)

Ended by NUCL(I)=0

---

[*]These variable names are not the names of program variables but were chosen to conform to the names used in and defined by ref. 7.

MATADOR Input

A. Title card; 80 characters, alphameric format.

B. FLUX, POWER, FLOW1, THERM, RES, FAST, FPABS

FORMAT(8F10.5)

FLUX = nominal--flux, cm^{-2} sec^{-1}

POWER = power, MW(t)

FLOW1 = feed rate, equivalent per second

THERM = thermal spectrum factor

RES = resonance spectrum factor

FAST = fast spectrum factor

FPABS = reference fission product absorption.

C. EPS, ERR, MAX, KARD

FORMAT(2F10.3,2I5)

EPS = error criterion in Gauss-Seidel solution

ERR = number less than which concentrations will be considered

zero

MAX = maximum number of iterations allowed for solution

KARD = signal for punched output, if no punched cards from

MATADOR.

D. AREA, VOL, RADIUS, PORTY, FILM, DIFFY, SOLBTY, VRATIO

FORMAT(8E10.3)

AREA = graphite area, cm^2

VOL = salt volume, cc

RADIUS = graphite rod radius, cm

PORTY = graphite porosity, cc-void/cc

FILM = mass transfer coefficient for krypton, cm/sec

DIFFY = diffusivity of krypton in pores, cm^2/sec

SOLBTY = solubility of krypton in fuel salt (mole/cc of liquid)/

(mole/cc of gas)

VRATIO = ratio of graphite volume to salt volume in core;

if AREA=0 next card is type G.

    E.   IGAS, IDAU, IENT, I3, I3DEP, NEXT

        FORMAT(4X,I6,4X,I6,I5,4X,I6,2I5)

        IGAS = gas identifier

        IDAU = identification of daughter

        IENT = 0 if daughter deposits

             = 1 if daughter is volatile

        I3   = identification of granddaughter formed from volatile

             daughter

        I3DEP = same as IENT except that it refers to I3

        NEXT = number used to control program

             > 0 return to next type E card

             = 0 read type G card

             < 0 read type F card followed by one of type E.

The first set refers to daughters of krypton, the next set after the

type F card refers to daughters of xenon.

    F.   FILM,DIFFY,SOLBTY

        FORMAT(3E12.5)

The variables are defined in the same way as those of card type D. After

this card, the next card read is of type E.

G.   INPUT(I),X(I),I=1,4

FORMAT(4(I5,F10.3))

INPUT = six-digit identifier of nuclide

X   = composition in atom fraction or mole fraction.

The materials fed to the reactor are ^{7}Li, ^{9}Be, ^{232}Th, and ^{19}F.

The molar feed rate is FLOW1 * X(I) in g-atom/sec.

H.   TIME(N),N=1,10

FORMAT(10E8.1)

TIME(N) = the removal time for group N of elements.

I.   NP(I),I=1,100

FORMAT(40(I2))

NP(I) = the group number for element with atomic number I.

J.   NZ(I),EFF(I),I=1,8

FORMAT(8(I3,F7.4))

EFF = the removal efficiency of the element with atomic number NZ.

Only those elements with a removal efficiency different from 1.0 need be

specified. As many cards as necessary may be used with input of EFFs,

stopping when NZ=0 is encountered.

K.   THETA   = six values.

FORMAT(8F10.3)

THETA(1) = holdup time of group 1 elements (noble gases) in the

circulating gas bubbles.

THETA(2) = holdup of group 2 elements (noble metals) deposited

in the reactor core.

THETA(3) = holdup of group 3 elements (seminoble metals) deposited

in the reactor core.

THETA(4) = holdup of group 1 elements in contact with salt

outside the flux of the reactor. The gas is assumed

to be the effluent of gases from the reactor.

THETA(5),THETA(6) = holdup time of group 2 and 3 elements,

respectively, in contact with salt outside the

reactor.

L. NTIME(I),PUNIT(I),I=1,10

FORMAT(10(I4,A4))

NTIME(I) = the removal time of group I elements expressed in units of

PUNIT(I); these are used only for printout purposes when going through

MATADOR for the first time.

M. NZ(I),EFF(I),I=1,8

FORMAT(8(I3,F7.4))

The variables are dummy variables with no relation to the variables

read as card type J. Here NZ is an atomic number as before, but EFF is

a fraction of the element of group 2 or 3 which is plated out in the

reactor. The remaining material is assumed to be deposited outside the

reactor flux. As many cards are read as necessary until NZ=0 is encountered.

MAIN input

The main program primarily reads input describing the flowsheet and

variables which control execution and output. The cards that follow are

read from unit 2:

A. I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,NWASTE,I12,I13,I14,I15

FORMAT (16I5)

Explanation of variables:

I1   If I1.GT.0 flowsheet input concentrations and removal

efficiencies are input from previous cases with I2.GT.0.

I2   If I2.GT.0 punch output to be read if I1.GT.0

I3   Predicted number of iterations; maximum number of

iterations when no time limit is used.

I4   Not needed

I5   Not needed

I6   If I6.GT.0 flowsheet information is punched for later

input when I8.GT.0.

I7   Not needed

I8   If I8.GT.0 flowsheet information is read from cards

punched earlier.

I9   No averaging is done on concentrations until I9 itera-

tions are complete.

I10   If I10.GT.0 averaging is not done on concentrations

between iterations.

NWASTE Number of waste streams

I12   If the number of iterations is less than this, nuclides

are not excluded because they do not affect D_{Li} values.

I13   I13 iterations are made at end in which nuclides do not

affect D_{Li} values, and are not excluded.

I14   If I14.GT.0, the program supplies MATADOR with a

derivative of reactor inlet concentration with respect

to reactor outlet concentration.

I15   If I15.GT.0, the program starts output section after

I15 minutes of calculation.

B. EPS, ERR, MAX

FORMAT (2F10.3, I5)

Explanation of variables:

    EPS   Convergence criterion divided by 10 (typical value).

    ERR   Number below which concentrations are set to 0.0.

    MAX   Not used.

C. WTDX

FORMAT (F10.3)

WTDX = the weighting factor given to DXIN.

D. NREG, NINX, NINY, NINP, NRATE, KRATE, JRATE

FORMAT (16I5)

Explanation of variables:

    NREG   Number of regions.

    NINX   Number of first-phase flowing streams.

    NINY   Number of second-phase flowing streams.

    NINP   Number of feed streams in processing plant.

    NRATE   Number of rate streams using a scale factor and a table number.

    KRATE   Number of rate streams described by parameters of gas-liquid contact for gas bubbled into tank.

    JRATE   Number of rate streams proportional to the first-phase flow rate (these are fluorinators described by a percent removal mechanism).

Many of these are used as limits for implied do-loops for input described below.

E. REGION(N), VOLX(N), VOLY(N), XLIB(N), NV(N), NSTR(N), IP(N), NC(N),

REGA(N), REGB(N), EIN(N), N=1, NREG

FORMAT (A8, 2X, 3E10.0, I2, I3, I2, Z3, A8, 2X, A8, E12.0)

Each of these cards describes a region. Three are NREG cards.

Explanation of variables:

REGION   Name of region, 8 characters.

VOLX   Volume of the first phase, cc.

VOLY   Volume of the second phase, cc.

XLIB   Parameter for equilibrium (e.g., lithium distribution,

temperature). Number of column applicable in table

of valences NV.LT.0 implies a correction is to be

made to XLIB.

NSTR   Number indicating the type of equilibrium.

IP   Controls printout on the basis of binary representation.

IP=0 Printing includes both phases.

IP=1 First phase information stored in left half

of line.

IP=2 Second phase information stored in left half

of line.

IP=3 First phase information stored in right half

of line.

IP=4 Second phase information stored in right half

of line.

Printout of a line occurs when a region has IP=0 or when two successive

regions have IP≠0. One of these should have a value of IP=1 or 2, and

the other should have IP=3 or 4.

NC   This variable has three hexadecimal digits. The first digit has a value from 0 to 3. Its binary digits indicate the phases whose molar volumes must be iteratively recalculated. Its values are:

0-00 Neither molar volume is to be recalculated.

1-01 Second phase molar volume calculations only.

2-10 First phase molar volume calculations only.

3-11 Both phases require molar volumes calculation.

The next two hexadecimal digits are used to identify the first and second phases, respectively. Even when molar volume calculations are not made, these can be used for assigning the assumed molar volumes.

REGA   Name of first phase to be used in output, 8 characters.

REGB   Name of second phase to be used in output, 8 characters.

EIN   The number of equivalents per cubic centimeter of the materials considered assumed in the second phase. If EIN is negative, the number of equivalents per cubic centimeter entering is used.

F. RINTO(I), INPUT(I), COMP(I), ICP(I), I=1, NINP

FORMAT (A8, 2X, I10, E12.0, I3)

Each card describes an input of one nuclide to a region. Number of cards is NINP.

Explanation of variables:

RINTO   Name of region fed, 8 characters.

INPUT   Numeric name (6-digit identification) of the nuclide.

COMP   Feed rate of INPUT to RINTO, moles/sec.

ICP   ICP.GT.0 identifies streams included in calculation
of second phase inlet equivalent density.

G. NSET

FORMAT (16I5) (only one variable read).

NSET is the number of cards of type H is NSET.GT.0.

H. RDEF(I), INDEF(I), DCOMP(I), I=1, NSET

FORMAT (A8, 2X, I10, E12.0)

These cards list the places where certain concentrations are
defined. They might be bismuth in a bismuth supply tank or $\sim$0.5
Li-Bi solution in a supply tank of that material.

     RDEF   Region name, 8 characters.

     INDEF   Numerical name of nuclide.

     DCOMP   The concentration in moles/cc of nuclide INDEF in
region RDEF.

I. RXFROM(J), RXTO(J), XRATE(J), J=1, NINX

FORMAT (A8, 2X, A8, E12.0)

     RXFROM   Region name, 8 characters.

     RXTO   Region name, 8 characters.

     XRATE   Flow rate of material of the first phase, cc/sec,
from region RXFROM to region RXTO.

J. RXFROM(J), RYTO(J), YRATE(J), J=1, NINY

FORMAT (A8, 2X, A8, E12.0)

Variable definitions are analogous to variables read as type I,
except that flows are second-phase flow rates. The cards of types F, H,
I, and J, and some of the cards that follow require the identification
of regions by name. These are the regions named by cards of type E.

The names are compared for exact matches (spaces count). When a name

for a stream destination (type I, and J) is not in the list of region

names, the name is compared with "DISCARD" (left justified). If this

comparison indicates a match, the stream is assumed to be discarded.

If the name does not match, it is assumed to be discarded, but a message

to this effect is printed. Note that the comparison with this key word

is made only when no successful comparison is made to listed region names.

Thus, the key word can be a legitimate region if the user so desires.

Cards of type K, L, and M are read only if NWASTE $\neq$ 0.

K. RWASTE(IW), IW=1, NWASTE

FORMAT (10A8)

      RWASTE   Names of destination of streams to be sent to

      ORIGEN.[7]

L. FWASTE (IW), IW=1, NWASTE

FORMAT (10F8.0)

      FWASTE   Multiplying factor to be used for these streams.

M. KWASTE (IW), IW=1, NWASTE

FORMAT (10I8)

      KWASTE   Used by OUTO to make a distinction between elements

      in these streams.

N. DET

FORMAT (20A4)

The value read is not used. This simply allows a comment card

in the input stream.

O. Read only if I1.GT.0

XIO   (800 values)

FORMAT (20A4)

XIO is the list of concentrations punched from previous programs.
The format allows a very compact representation of single pre-
cision numbers. One must ensure that the first number on the
card does not identify the card as an end-of-file or end-of-
information indication.

Next is read (if I1.GT.0):

DXIN   (800 values)

FORMAT (20A4)

DXIN may be removal efficiencies or molar flow rates; their
values are examined for indications of which one they represent,
and the other is then calculated. This is not a fool-proof
method of identification, however.

If I8.GT.0, the program now reads a description of the flowsheet in the
format in which it can later be punched. In addition to replacing the
values in the previously defined variables, it replaces the values in
several variables determined from these. This is done to allow the use
of the parameter values arrived at in previous runs and thus save recal-
culation by iterations.

   P. Read only if NRATE.GT.0

       RXNA(I), RXNB(I), SCFACT(I), NFORM(I), I=1, NRATE

       FORMAT [2(A8, 2X, A8, E12.0, I5, 5X)]

       Elements are assumed to go from the region RXNA to the region

       RXNB at a rate equal to the product of their concentration,

       the value of SCFACT, and the value listed in the rate table

       opposite the element symbol in column NFORM. If NFORM is

greater than 100, the table used is NFORM-100, but the table

values are changed to 1 except where the value is 0.0.

Q.   There are JRATE cards of this type.

REGL, REGG, ITH, COMM

FORMAT (A8, 2X, A8, 2X, I5, 5X, 25A1)

REGL   Name of region at origin of the stream.

REGG   Name of region at destination of the stream.

ITH   Column number of rate constant table.

COMM   25-character comment to appear in output listing.

Streams of this type use a scale factor equal to the flow of

material through REGL.

R.   There are KRATE cards of this type.

REGL, REGG, AKL, AB, AS, XG, XL, ITD, ITH, ITC

FORMAT (A8 ,2X, A8, 2X, 5F10.7, 2I3, I2)

REGL, REGG, ITH are defined as above.

AKL   Overall liquid phase mass transfer coefficient,

cm/sec.

AB   Bubble area, cm^2.

AS   Surface area, cm^2.

XG   Thickness of stagnant gas through which diffusion

occurs.

XL   Thickness of stagnant liquid through which diffusion

occurs.

ITD   Column number giving gas phase diffusivities.

ITC   Variable controlling input. If ITC is not 0, the

next card will be an extension of this card and

will replace the value of the comment with the

first 25 characters on the next card.

This type of card produces three rate streams which represent mass

transfer to and from the gas at the bubbles and at the surface, and

the transfer of nonvolatile nuclides from the gas to the liquid at the

surface.

S. FMT   (100 values, 5 cards required)

FORMAT   (20A4)

FMT is in the form of a format instruction and is the heading

printed at the top of the table of rate constant data read next.

T. RDATA(I), I=1,800

FORMAT   (8X, 8E8.0)

RDATA   Contains the data referred to by column number

by variables ITH, ITD, and NFORM. The data should

be arranged on cards so that cards are in order of

atomic number (100 cards) with the columns con-

taining the numbers described previously. These

can be diffusivities, Henry's law constants,

PR/(1-PR) (where PR is percent removal), or any

other constants which might vary with atomic

number.

After these cards are read, input is read by subroutine EQKN and

subroutine VOLUME, respectively.

INPUT TO EQKN

EQKN reads a series of card pairs from unit 50 to assign values to the distribution coefficients throughout the processing plant. The cards read contain:

A. LZ, NS, NT, AN, A, B, ENAME

FORMAT (I3, 2I2, 8X, F2.0, 10X, E8.0, 2X, E8.0, 33X, A2)

B. LZ2, NS2, REF

FORMAT (I3, I2, 75A1)

LZ = atomic number

NS = number identifying the type of equilibrium corresponds to NSTR in the description of regions.

NT = identifies the type of equation involved.

0   End of input, return

1       $\log_{10}(D) = (AN) \log_{10} XLIB + A+B/T$ (in $^\circ$ K)

2       $D = A$

3,4 $\log(P) = A/(273.2 + XLIB) + B$

D   Distribution coefficient

XLIB   Constant read with region descriptions

P   Partial pressure

AN, A, B = data

ENAME = elemental symbol name for output of equilibrium

LZ2, NS2 = not used

REF = 75 character reference for data in the first card.

Two special flags are recognized. If the element atomic number is 102, the value of NT is assumed to be 2. In addition, all elements are

assigned the same distribution constant for the distribution type NS. This overrides previous input, but it is assumed that exceptions follow. This device greatly reduces the amount of input. If the atomic number LZ is 101, no data are taken from the cards, but the variable REF is assumed to be a page heading for this type of equilibrium. For proper output tabulation, it is assumed that all cards specifying a particular type of equilibrium are together and that these are separated by title changes.

The program prints the form of the equation in the output table. The space allowed between constants on input cards may be used to type the equation that will be used.

INPUT TO VOLUME

All input to VOLUME is read from unit 2 and is used to make estimates of molar volumes and a table of valences.

A. NTA, NTB

   FORMAT (32I2)

   Each variable has 16 elements corresponding to the value of the second and third hexadecimal digit of NC, which corresponds to regions and identifies the phases present. The values identify the column numbers in VOLC.

B. VOLC

   FORMAT (8X, 8E8.0)

   Eight hundred values are read, corresponding to 8 values each (1 card) for the elements in increasing order of atomic number. A listing gives 8 columns, which correspond to phases, and 100

rows, which correspond to elements. VOLC contains the contri-
bution of the elements to the molar volume of each of the phases.

C. VAL

FORMAT (8X, 8E8,0)

VAL has 800 values arranged similarly to VOLC. The values of
VAL are valences for use in contactors, with columns being
referred to by the variable NV.

APPENDIX C: OUTPUT

Output Produced by MATADOR and its Subroutines

A.   Output from AMATRX:

(1)   Nuclear Library

(2)   Summary of fission yields

The fission yields are summarized by mass chains in two groups, and cumulative totals are given after each group for particle and mass yields as indicated by the nuclear library.

B.   Output from MATADOR on the first call:

MATADOR produces a list of the input variables.

C.   Output from CHEMPL on the first call:

CHEMPL produces an element-by-element table of removal times using the starting efficiencies, the removal time given by NTIME, and the units given by PUNIT. The user is responsible for ensuring agreement of NTIME and PUNIT with the variable TIME in units of seconds.

D.   Output from RESULT produced each time RESULT is called:

(1)   A table is produced that lists the concentration for each nuclide in moles per cubic centimeter and atom fraction, its contribution to poisoning, its inventory in curies, its contribution to the power density, the fraction of that power produced as gamma heat, its chemical processing rate, and its mole fraction in the inlet stream as determined by the processing plant code.

(2)   This table is followed by a list of calculated values which indicate reactor performance.

(3) This information is summarized for the 25 greatest values under most of the headings in the first table; another table is given providing totals for each stream.

(4) A summary is presented that compares thorium burnup rate with the weight of fission products and actinides removed.

(5) A summary table is given that lists the total rate of chemical processing for each element.

(6) A note is included on the number of calls to MATADOR. (This is only accurate if KEE is called every time a pass is made through MATADOR.)

Output from the Processing Plant Program

Printed Output

The output begins with output obtained by the first call to MATADOR. The program then prints the input variables. Tables appear that describe all regions, list-feed streams and specified concentrations, describe flowing streams in each phase and streams defined by rate coefficients, indicate expressions for distribution coefficients, and compute additive molar volumes and valences.

A number of intermediate calculated results are printed to monitor the progress of the calculations. Between iterations, there is a list of the 50 nuclides with the greatest relative change in concentration returning to the reactor, and the concentrations and the relative changes. Upon convergence, this list is printed for all nuclides. The lithium distribution XLIB calculated by VOLUME is also printed between iterations, with the change in equivalent density entering

each stage for which the distribution coefficients are recalculated.
As a means of monitoring the progress of calculations, the time in
hundredths of a second is printed at various points in the calculation.

When the time limit or convergence criterion is met, another call
is made to RESULT to obtain output summarizing the reactor calculations.
A partial listing of the flowsheet description is made which reflects
the new flow rates used. A table, whose heading gives the half-life
and removal time of the nuclide, is included for each nuclide in the
library. The table lists the concentration (moles/cc), the inventory
(in moles and in curies), and the total flow rate (in moles/day) of the
nuclide in each phase of each region in the processing plant. The tables
are shortened by the combining of one-phase regions as indicated by the
input variable, IP. Each of these tables is followed by a list of all
discard rates in moles per day, and a list of flow rates in moles per
day in all streams which do not represent the total flow of one phase
out of a region. In all of these, the regions and phases are identified
by their names. When two regions each have one phase to be listed on
the same line, as specified by IP, the region name of the first of the
two regions in the input is used to apply to both. This output is repeated
in a summary table for each element in which the totals are given for the
values in the previous tables. These tables are followed by a table giving
the totals for all materials.

This series is followed by a series of tables which give heat genera-
tion rates in each phase of each region. For each table, the 50 nuclides
with the highest heat generation rate at that point in the flowsheet are
listed along with their contribution to total heat generation rate, gamma

heat generation rate, and beta heat generation rate. Each of these is given as total power in megawatts, and as power density of kilowatts per liter. These are totaled for all nuclides, and an adiabatic temperature rise in °C per minute is printed.

In addition to printed output, the program prepares punched output which includes reactor concentrations and processing plant removal times for all nuclides, and a description of the flowsheet. This output, if requested by positive values for the input variables I2 and I6, can be used later to essentially continue the calculations from the point at which output was obtained. The reactor concentrations and processing plant removal times can also be used to eliminate many of the iterations required when a flowsheet with similar performance is used.

Output to be Used by ORIGEN

A copy of ORIGEN[7] was obtained, and modifications were made to allow the printing of only those tables specified by input variables, to allow input formats to be used which require flow rates in a form convenient for output by the processing plant code, and to allow input to be read from a source other than the card reader. Output in this form describing a series of streams is produced on unit 28 for later input to ORIGEN. In all programs used to date, this output was written on a disc file read by ORIGEN in a second job step.

The first streams included are specified by the input variables NWASTE, RWASTE, FWASTE, and KWASTE. For output stream IW ($\leq$NWASTE), the variable RWASTE(IW) gives the name of the destination of all streams that are included. Streams that are discarded may be sent to a nonexistent

region name in the list of waste regions. The included streams are
further decreased by the use of KWASTE(IW) to specify the phase as follows:

KWASTE·LT·0   only first-phase streams are considered,

KWASTE·EQ·0   streams from either phase are considered,

KWASTE·GT·0   only second-phase streams are considered.

The molar flows of streams meeting these requirements are added and
multiplied by ABS[FWASTE(IW)] before being sent to ORIGEN and included
in the total. For the special case of KWASTE(IW)·NE·0, and FWASTE(IW)
between -0.001 and 0.0, the value of FWASTE(IW) is 0.001 when considering
isotopes of neptunium. Streams are then output which represent 5 liters
of salt sticking to the graphite divided by the number of days in 4 years.
these represent daily discard rates of noble metals from the reactor core,
noble gases from the salt drain tank, and noble metals from outside the
reactor core. All of these, except the salt sticking to the graphite,
are added to the flow output streams with FWASTE(IW)·GT·0; the total is
output to represent total daily waste other than graphite removal.
Finally, the daily rate of deposition of materials on the graphite is
output as a stream that ORIGEN will assume is irradiated for 4 years.

ORNL/TM-4210
Dist. Category UC-76

INTERNAL DISTRIBUTION

1-2.	MSRP Director's Office	48.	H. G. MacPherson, UT
3.	C. E. Bamberger	49.	R. E. MacPherson
4.	M. Bender	50.	A. P. Malinauskas
5.	R. E. Brooksbank	51.	C. L. Matthews, ERDA-ORO
6.	C. H. Brown, Jr.	52.	G. T. Mays
7.	W. D. Burch	53.	H. E. McCoy
8.	W. L. Carter	54.	T. W. Pickel
9.	H. D. Cochran, Jr.	55.	H. Postma
10.	R. M. Counce	56.	M. W. Rosenthal
11.	F. L. Culler	57.	H. C. Savage
12.	J. M. Dale	58.	C. D. Scott
13.	F. L. Daley	59.	J. T. Shannon
14.	J. R. DiStefano	60.	M. J. Skinner
15.	R. L. Egli, ERDA-ORO	61.	F. J. Smith
16.	J. R. Engel	62.	J. W. Snider
17.	G. G. Fee	63.	I. Spiewak
18.	D. E. Ferguson	64.	M. G. Stewart
19.	L. M. Ferris	65.	D. B. Trauger
20.	A. T. Gresky	66.	D. Y. Valentine
21.	W. R. Grimes	67.	J. W. Wachter
22.	W. S. Groenier	68.	J. S. Watson
23.	R. H. Guymon	69.	A. M. Weinberg, ORAU
24.	B. A. Hannaford	70.	J. R. Weir
25.	R. F. Hibbs	71.	M. E. Whatley
26-27.	J. R. Hightower, Jr.	72.	J. C. White
28.	V. A. Jacobs	73.	M. K. Wilkinson
29-43.	C. W. Kee	74.	R. G. Wymer
44.	O. L. Keller	75.	E. L. Youngblood
45.	A. D. Kelmers	76-77.	Central Research Library (2)
46.	H. T. Kerr	78.	Document Reference Section
47.	J. A. Lenhard, ERDA-ORO	79-81.	Laboratory Records (3)
		82.	Laboratory Records (LRD-RC)

CONSULTANTS AND SUBCONTRACTORS

83. J. C. Frye
84. C. H. Ice
85. E. A. Mason
86. W. K. Davis
87. R. B. Richards

EXTERNAL DISTRIBUTION

88. Research and Technical Support Division, ERDA, Oak Ridge Operations Office, P. O. Box E, Oak Ridge, Tenn. 37830
89. Director, Reactor Division, ERDA, Oak Ridge Operations Office, P. O. Box E, Oak Ridge, Tenn. 37830
90-91. Director, ERDA Division of Reactor Research and Development, Washington, D.C. 20545
92-195. For distribution as shown in TID-4500 under UC-76, Molten Salt Reactor Technology