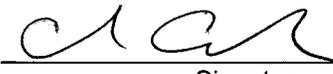
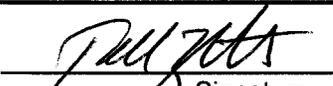


Appendix E

<p style="text-align: center;"><b>NUCLEAR WASTE MANAGEMENT PROCEDURE</b></p> <p><small>Sandia National Laboratories</small></p>	<h2 style="margin: 0;">Design Document Criteria</h2>	<p><b>Form Number: NP 19-1-4</b></p> <p><b>Page 1 of 1</b></p>
<p>1. <b>Software Name:</b> <u>BRAGFLO</u></p> <p>2. <b>Software Version:</b> <u>6.02</u></p> <p>3. <b>Document Version:</b> <u>6.02</u></p> <p>4. <b>ERMS #:</b> <u>558660</u></p> <p style="font-size: small;">Prior to sign-off of the DD, all items shall be appropriately addressed by the code sponsor so that "Yes" may be checked. Include this form as part of the DD.</p>		
<p><b>Are the following appropriately defined and documented in the DD?</b></p> <p>5. Major Software Components <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>6. Technical description of the software with respect to: theoretical basis, embodied mathematical model, major control flow, control logic, and data structures <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>7. Functionalities and interfaces of object components, functions, and subroutines. <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>8. Allowable or prescribed ranges for inputs and outputs <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>9. Verifiability: Is the design verifiable through testing or other means? <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>10. Consistency and Traceability: Is the design consistent with and traceable to the software's requirements? <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>11. Technical Feasibility: Is the design technically feasible? <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p> <p>12. Implementation: Is the design presented in sufficient detail to allow for implementation as computer software? <span style="float: right;"><input checked="" type="checkbox"/> Yes</span></p>		
13.	<p><u>Chris Camphouse</u> <b>Code Team/Sponsor</b> <i>(print)</i></p>	<p style="text-align: center;"> Signature</p> <p style="text-align: right;"><u>12/18/12</u> Date</p>
14.	<p><u>Todd R. Zeitler</u> <b>Technical Reviewer</b> <i>(print)</i></p>	<p style="text-align: center;"> Signature</p> <p style="text-align: right;"><u>12/18/2012</u> Date</p>
15.	<p><u>Shelly Nielsen</u> <b>QA Reviewer</b> <i>(print)</i></p>	<p style="text-align: center;"> Signature</p> <p style="text-align: right;"><u>12-18-12</u> Date</p>
16.	<p><u>Sean Dunagan</u> <b>Responsible Manager</b> <i>(print)</i></p>	<p style="text-align: center;"> Signature</p> <p style="text-align: right;"><u>12/18/12</u> Date</p>
17.	<p><u>Jennifer Long</u> <b>SCM Coordinator</b> <i>(print)</i></p>	<p style="text-align: center;"> Signature</p> <p style="text-align: right;"><u>1/8/13</u> Date</p>

# Information Only

**Design Document**

**for**

**BRAGFLO, Version 6.02**

**Document Version 6.02**

**ERMS# 558660**

**Information Only**

## Table of Contents

1.0	INTRODUCTION .....	5
1.1	Software Identifier .....	5
1.2	Points of Contact .....	5
2.0	FUNCTIONAL REQUIREMENTS .....	6
3.0	MAJOR SOFTWARE COMPONENTS .....	9
4.0	DESCRIPTION OF THE MODELS AND METHODS .....	11
5.0	MAJOR CONTROL FLOW AND CONTROL LOGIC .....	12
5.1	Flow Chart with Functional Requirements Map .....	12
5.2	Call Tree with Subroutine Descriptions .....	18
5.3	BRAGFLO Routines and Descriptions .....	26
5.4	Data Flow .....	37
5.4.1	Common Blocks .....	38
5.4.2	Subroutine Arguments .....	45
6.0	DATA STRUCTURES.....	46
6.1	Problem Sizing Parameters.....	46
6.2	Input Data File .....	46
6.3	Creep Closure Look-Up Table Data .....	46
6.4	Output Data Files.....	46
6.4.1	Binary Output (.BIN) File.....	47
6.4.2	ASCII Output (.OUT) File.....	47
6.4.3	Summary (.SUM) File .....	47
6.4.4	Restart Output (.ROT) File.....	48
7.0	CAPABILITIES AND LIMITATIONS OF THE SOFTWARE FOR THE WIPP APPLICATION .....	49
8.0	REFERENCES .....	51

**Information Only**

**List of Figures**

Figure 1. BRAGFLO flow chart – overall flow..... 13  
Figure 2. Flow chart for solver phase. .... 14  
Figure 3. Flow chart for materials phase. .... 15  
Figure 4. Flow chart for initialization phase..... 16  
Figure 5. Flow chat for properties phase. .... 17  
Figure 6. Flow chart for Jacobian phase..... 18

**List of Tables**

Table 1. BRAGFLO Common Block Descriptions..... 38  
Table 2. BRAGFLO Common Block Usage ..... 40  
Table 3. BRAGFLO Common Block Usage (continued)..... 42

## Acronyms

<b>BRAGFLO</b>	BRine And Gas FLOW code
<b>CCA</b>	Compliance Certification Application
<b>CDB</b>	CAMDAT Database
<b>CRA</b>	Compliance Recertification Application
<b>DEC</b>	Digital Electronic Corporation
<b>DOE</b>	U.S. Department of Energy
<b>DRZ</b>	Disturbed Rock Zone
<b>EOS</b>	Equation of State
<b>PA</b>	Performance Assessment
<b>PAVT</b>	Performance Assessment Verification Tests
<b>PDE</b>	Partial Differential Equation
<b>PREBRAG</b>	Preprocessor for BRAGFLO
<b>POSTBRAG</b>	Postprocessor for BRAGFLO
<b>QA</b>	Quality Assurance
<b>QAP</b>	Quality Assurance Procedure
<b>RKS</b>	Redlich–Kwong–Soave (gas equation of state)
<b>VMS</b>	Virtual Memory System (a DEC operating system)
<b>WIPP</b>	Waste Isolation Pilot Plant

**Information Only**

## 1.0 INTRODUCTION

This document serves as a Design Document for BRAGFLO Version 6.02. As such, this document describes the following major features of software design: software components, theoretical basis, mathematical model, control flow and logic, data structures, and functionality and the interfaces of subroutines. The *BRAGFLO User's Manual* (Camphouse 2012c) describes in detail the allowed or prescribed ranges for input data. Because BRAGFLO was acquired as developed software and has been extensively modified for performing WIPP PA calculations, this document provides an 'as built' design rather than prescribing how the code is to be developed.

### 1.1 Software Identifier

Code Name: BRAGFLO

WIPP Prefix: BF2

Version: 6.02

Platform: Compaq FORTRAN 7.5 for OpenVMS AXP, version 8.2, on DEC Alpha.

The previous version of BRAGFLO was Version 6.0.

The previous version of the RD/VVP was Version 6.0, ERMS# 545014.

### 1.2 Points of Contact

Sponsor: R. Chris Camphouse  
Sandia National Laboratories Carlsbad Office  
4100 National Parks Highway  
Carlsbad, NM. 88220  
(575) 234-0130

Consultant: Daniel Clayton  
Sandia National Laboratories  
P.O. Box 5800, MS-0747  
Albuquerque, NM 87185  
(505) 284-5360

**Information Only**

## 2.0 FUNCTIONAL REQUIREMENTS

The functional requirements for BRAGFLO version 6.02 include all of those specified for BRAGFLO version 6.0 (Nemer 2006), with one change to requirement R.20, and are listed below.

- R.1 Input defining the problem to be run is read in, including mesh size, simulation time specifications, and output files to be used.
- R.2 The time and/or frequency of output to each output file and the variables whose values will be output are specified in the input.
- R.3 The code reads input defining the finite difference grid for the problem to be run, including values of  $\Delta x$ ,  $\Delta y$ , and  $\Delta z$ , and elevations of grid block centers, or input parameters that allow BRAGFLO to calculate the elevations. These input parameters may also specify how the mesh is rotated or dipped relative to a reference right hand coordinate system.
- R.4 The default boundary conditions are no-flow conditions. BRAGFLO also allows constant pressure (Dirichlet) boundary conditions to be applied at specified grid blocks. The Dirichlet conditions fix the brine pressure and gas saturation at their initial values.
- R.5 The following initial conditions are specified on input at each grid block: brine pressure, brine saturation, iron concentration, the concentration of biodegradables and the MgO concentration.
- R.6 Parameters are input that controls the numerical behavior of BRAGFLO. These include specifications of convergence tolerances, iteration limits, upstream weighting control parameters, dependent variable increments used to calculate Jacobian element derivatives numerically, and the solver to be used. Two solvers are currently available, the original LU decomposition solver and the point successive over-relaxation (SOR) solver, and only the original LU decomposition solver is qualified for use in WIPP compliance calculations.
- R.7 Material maps at specified times are input. For each material, material properties are input, including two-phase flow parameters, intrinsic permeability's, reference condition porosities, and compressibility's. Also specified are the relative permeability and capillary pressure model to be used for each material. Furthermore, BRAGFLO allows for the change of pressure and saturation values, as well as the capability to turn off the chemistry reactions when materials change as specified by the .INP input file.
- R.8 Relative permeabilities and capillary pressure are calculated for each material using one of several available models. The compliance calculations will use both a modified Brooks-Corey model and a van Genuchten/Parker model. Some of the QA verification

- and validation tests use relative permeability models that are included in BRAGFLO specifically for those tests.
- R.9 A simplified rock fracture model allows the porosity and permeability to increase as pore pressure increases above a threshold value, simulating a fracture network. In the absence of fracturing, porosity may vary slightly with pressure due to rock compressibility effects.
- R.10 The Klinkenberg effect is included, allowing the gas-phase permeability to become elevated over the intrinsic brine-phase permeability at low pressures in low-permeability materials.
- R.11 Brine salinity, density at reference conditions, and compressibility are specified by input values, and the density of brine varies as a function of pressure.
- R.12 Gas properties are given by the Redlich-Kwong-Soave (RKS) equation of state (EOS). WIPP compliance calculations will use a single pure gas, H<sub>2</sub>.
- R.13 Two chemical reactions are modeled in BRAGFLO: anoxic corrosion of iron and microbial degradation of celluloses. Both of these reactions produce gas (H<sub>2</sub>) and consume brine according to specified stoichiometry. Reaction rates and stoichiometry are specified for brine-inundated conditions and for humid conditions. BRAGFLO reads the necessary parameters from the .INP input file.
- R.14 The effects of salt creep, whereby the surrounding halite closes in on the waste or other excavated regions, compressing them, may be simulated in BRAGFLO. Using this model, the porosity of the waste changes with time and brine pressure. The input file, containing the closure surface data (.CSD), is an ASCII input file that is read by BRAGFLO if creep closure is to be simulated as specified by a parameter in the ASCII input file (.INP). If creep closure is not to be simulated a dummy file must be included even though this dummy file will not be read. The information in the .CSD file, that is used in the validation of BRAGFLO, will be the same information that was incorporated into the BRAGFLO 4.10 executable file. BRAGFLO reads the necessary information from the .INP and .CSD input files.
- R.15 BRAGFLO numerically calculates the flow of two phases, brine and gas, in porous media as a function of time and space, using an implicit finite difference method with variable time step control.
- R.16 Well models in BRAGFLO allow simulation of wells with a single completion within the formations or porous media being modeled. The types of wells that can be modeled are constant injection or production rate wells and constant down-hole pressure wells. Well data are assigned values in the input control file.
- R.17 BRAGFLO simulates flow through heterogeneous as well as homogeneous porous media.

- R.18 BRAGFLO writes binary and ASCII output files. These files regurgitate much of the input file information as well as primary and secondary dependent variables at user specified frequencies or at specific times. Many output variables are user selected and defined.
- R.19 BRAGFLO provides the user with an error check on the consistency with how waste regions are specified.
- R.20 Six additional chemical reactions are modeled in BRAGFLO: sulfidation of iron hydroxide; sulfidation of iron; hydration of MgO; carbonation of MgO; carbonation of Mg(OH)<sub>2</sub>; and conversion of hydromagnesite. Stoichiometry is specified for each reaction. BRAGFLO reads the necessary parameters from the .INP input file.
- R.21 Chemical rates of all seven reactions are zero below a specified cutoff value in brine saturation, which is set in the .INP input file.
- R.22 Chemical rates of all seven reactions are multiplied by their respective initial concentrations if specified by a flag in the .INP input file.
- R.23 BRAGFLO calculates the change in solid volume from the chemical reactions.
- R.24 BRAGFLO allows for the smooth change in permeability of materials with time, as specified by the .INP input file.
- R.25 Software: For input, this code reads the output file generated by the code PREBRAG.
- R.26 Software: For input, BRAGFLO reads an external file that contains the creep closure look-up table data.
- R.27 Software: The output of this code is read as input by the code POSTBRAG.

### 3.0 MAJOR SOFTWARE COMPONENTS

There are basically three processing phases within the BRAGFLO software: problem specification, solution and output of results. The modules that make up each of these phases can be found in the flow charts and subroutine descriptions in Section 5.0. The problem specification phase consists of reading user specified input data, defining the computation grid, mapping material parameters to the grid, defining boundary and initial conditions and setting up the matrices and arrays that form the system of equations to be solved. Excluding the restart capability, which is not qualified for WIPP compliance calculations, there are two BRAGFLO input files and three output files. The general BRAGFLO input control file (BF2\_UIF\$INPUT) and the salt creep closure data input file (BF2\_UIF\$INPUTCS) are described in detail in the *BRAGFLO User's Manual* (Camphouse 2012c) and summarized in Sections 6.2 and 6.3. The solution phase consists of a time loop. Within the time loop is an iteration loop for the solution of the nonlinear mass balance equations. Within the iteration loop is an update of system matrices and arrays to reflect current conditions, a choice of solvers (detailed below), and application of solution convergence criteria. The details of the mathematical model and solution algorithm are provided in Section 4.0 The output phase consists of writing user specified variables at specified times and spatial locations to specified binary, (BF2\_BIN\$OUTPUT), and ASCII, (BF2\_DBG\$OUTPUT, BF2\_DBG\$SUMMARY) files. The format of these files is described in Section 6.4.

The BRAGFLO code has been maintained as a standalone code – all input comes from two-formatted text files and output is written to both binary and formatted text files for subsequent post processing. The code is interfaced to the WIPP PA ‘system’ through the pre- and post-processing tools PREBRAG (Gilkey and Rudeen, 2007) and POSTBRAG (Nemer 2007). PREBRAG reads the problem specification data and material parameters from an input CAMDAT Database file (.CDB) (Gilkey, 2006b) and creates one of the input text file. The other input text file must be built by the user and contains the salt creep closure data. Both are described in the *BRAGFLO User's Manual* (Camphouse 2012c). POSTBRAG reads the files output by BRAGFLO and creates a CDB containing the input grid and some material parameters as well as time dependent results.

The solution phase of BRAGFLO over a time step requires the solution of a nonlinear system of mass balance equations. The nonlinear system uses a Newton-Raphson iteration, which solves successive linear approximations to the nonlinear system. At each Newton-Raphson iteration there is an update of the matrices and arrays that form the linear approximation. A linear equation solver then provides a solution of the approximating linear system. Linear equation solvers fall into two general categories - direct or elimination type solvers and iterative solvers. Four solver packages are implemented in BRAGFLO - two direct solvers and two iterative solvers, however, only the LU decomposition direct solver is qualified for WIPP compliance calculations.

Finally, BRAGFLO makes use of the WIPP PA libraries CAMCON\_LIB and CAMSUPES\_LIB to standardize common user interface and I/O functions (Gilkey, 2006a; Gilkey, 2006c). These libraries are linked to the BRAGFLO executable during the build process as documented in the *BRAGFLO Implementation Document* (Camphouse 2012a).

The CAMCON\_LIB library provides a library of standardized QA and user interface routines, which provide a code banner with credits (authors), software ID, revision date, and current date and time. File processing routines read and parse the command line, and specify and open files; free-field input routines; and string processing routines.

The CAMSUPES\_LIB library is a collection of machine dependent routines to access the system clock and allocate memory. However, dynamic memory allocation has not been implemented in BRAGFLO.

**Information Only**

## 4.0 DESCRIPTION OF THE MODELS AND METHODS

Quantification of the effects of gas and brine flow on radionuclide transport for undisturbed and disturbed conditions requires use of a two-phase flow code. For performance assessment at the WIPP repository, the U.S. Department of Energy (DOE) uses the two-phase flow code BRAGFLO to simulate brine and gas flow. The BRAGFLO code incorporates the effects of disposal room consolidation and closure, gas generation from corrosion and biodegradation reactions, brine consumption from corrosion, and inter-bed fracturing in response to gas pressure. The *BRAGFLO User's Manual* (Camphouse 2012c) in the section titled *Description of Models and Methods* develops the governing equations for BRAGFLO, its initial and boundary conditions, and the submodels incorporated in BRAGFLO that were developed specifically for performance assessment of the WIPP.

## 5.0 MAJOR CONTROL FLOW AND CONTROL LOGIC

The BRAGFLO control flow and logic discussed in subsequent sections provides the following: an overview of BRAGFLO using flow charts at the module and subroutine level; a hierarchical call tree that includes a brief description of all subroutines contained in BRAGFLO; and finally, data flow is presented as a common block ‘include’ table, common block descriptions and a description of subroutine arguments. The hierarchical structure of the call tree provides the details of which and where subroutines are called.

### 5.1 Flow Chart with Functional Requirements Map

The major control flow and control logic for BRAGFLO is presented in Figure 1 in the form of a flow chart. In the flow chart subroutine names are shown in UPPER CASE, and descriptive information in ‘Sentence case’. Also shown in the figure is a mapping of Functional Requirements. The requirement numbers (i.e., R.1) corresponds to those presented in the *Requirements Document* (Camphouse 2012b) and summarized in Section 2.0. The qualifier, “See #”, is a reference number to a description of that routine that appeared on an earlier subroutine in the set of flow chart figures. Figure 1 shows the overall flow logic of BRAGFLO and roughly corresponds to the BRAGFLO main routine. This main routine contains the time loop. Subsequent figures show details of modules that are first introduced in pentagonal elements in earlier figures. The modules (groups of subroutines) detailed in Figure 1-Figure 6 are:

INITIALIZE – Reads input data and sets up the problem to be solved.

MATERIALS – Initializes and loads the material properties arrays, such as, relative permeability, threshold capillary pressure, creep closure, gas and brine density, consolidation and dissolved gas parameters.

PROPS (Properties) – calculates the above material properties and adds rock compressibility, H<sub>2</sub> production due to radiolysis, corrosion rates, and modifies gas permeability for the Klinkenberg effect.

SOLVER – controls the Newton-Raphson iterations for the current time step by updating material properties and the Jacobian, applying boundary conditions, choosing the solver, incorporating wells, and checking for convergence.

GETJAC (Jacobian) – controls the calculation of the Jacobian by incorporating materials properties, wells and setting boundary conditions.

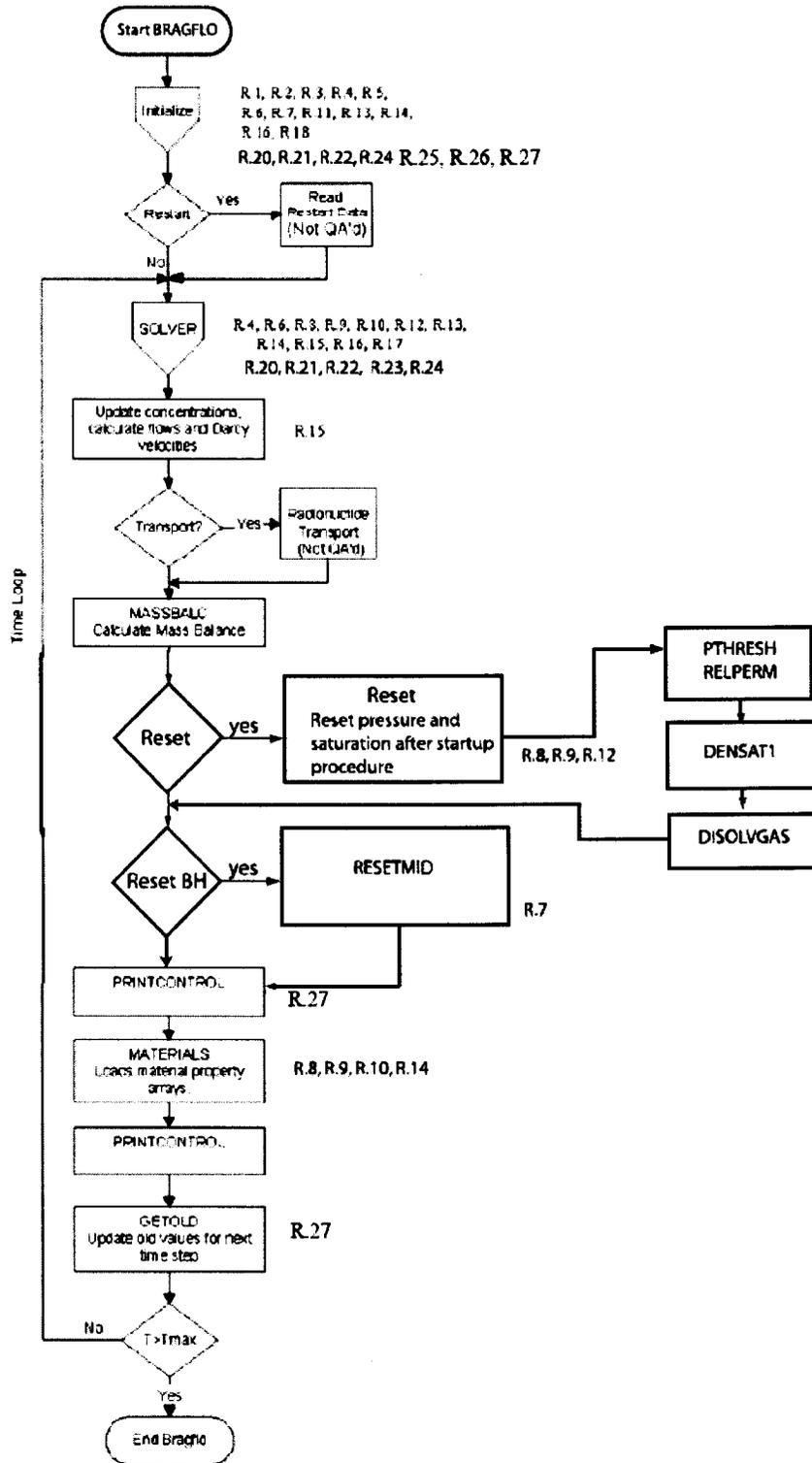
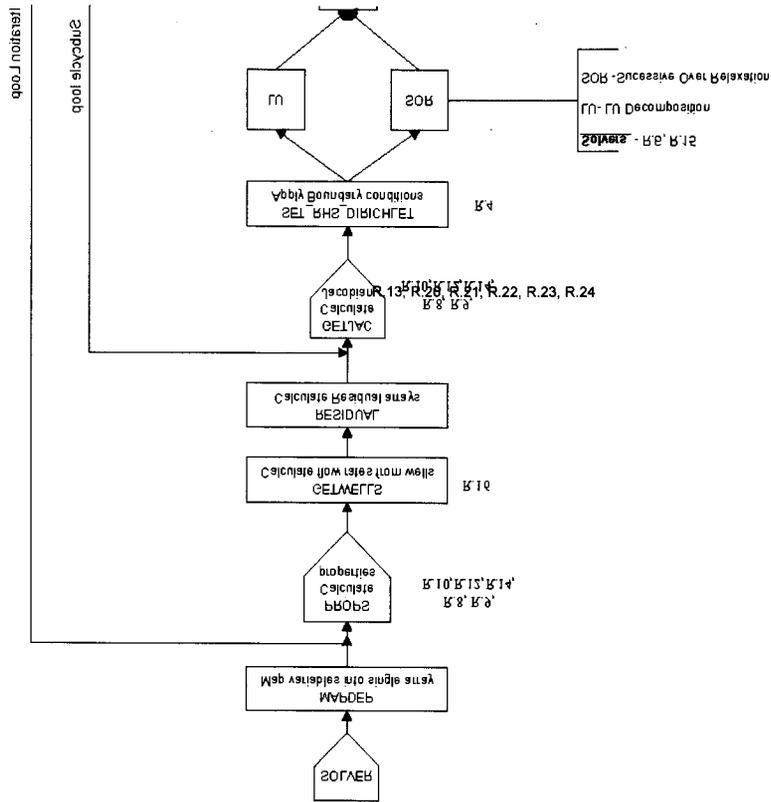


Figure 1. BRAGFLO flow chart – overall flow.



R.13, R.20, R.21, R.22, R.23, R.24

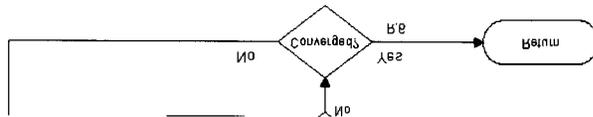


Figure 2. Flow chart for solver phase.

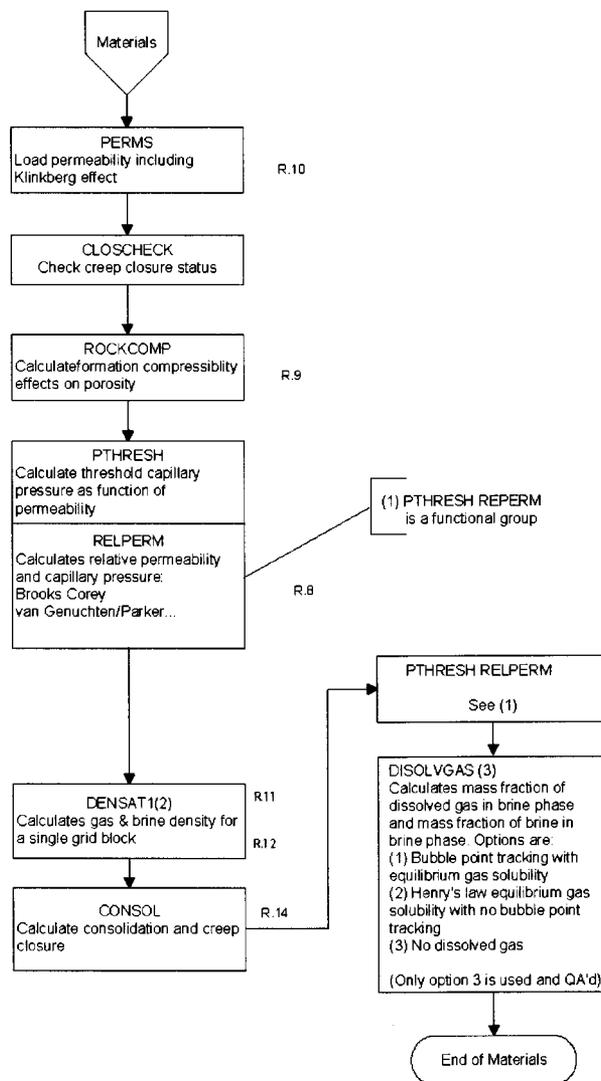


Figure 3. Flow chart for materials phase.

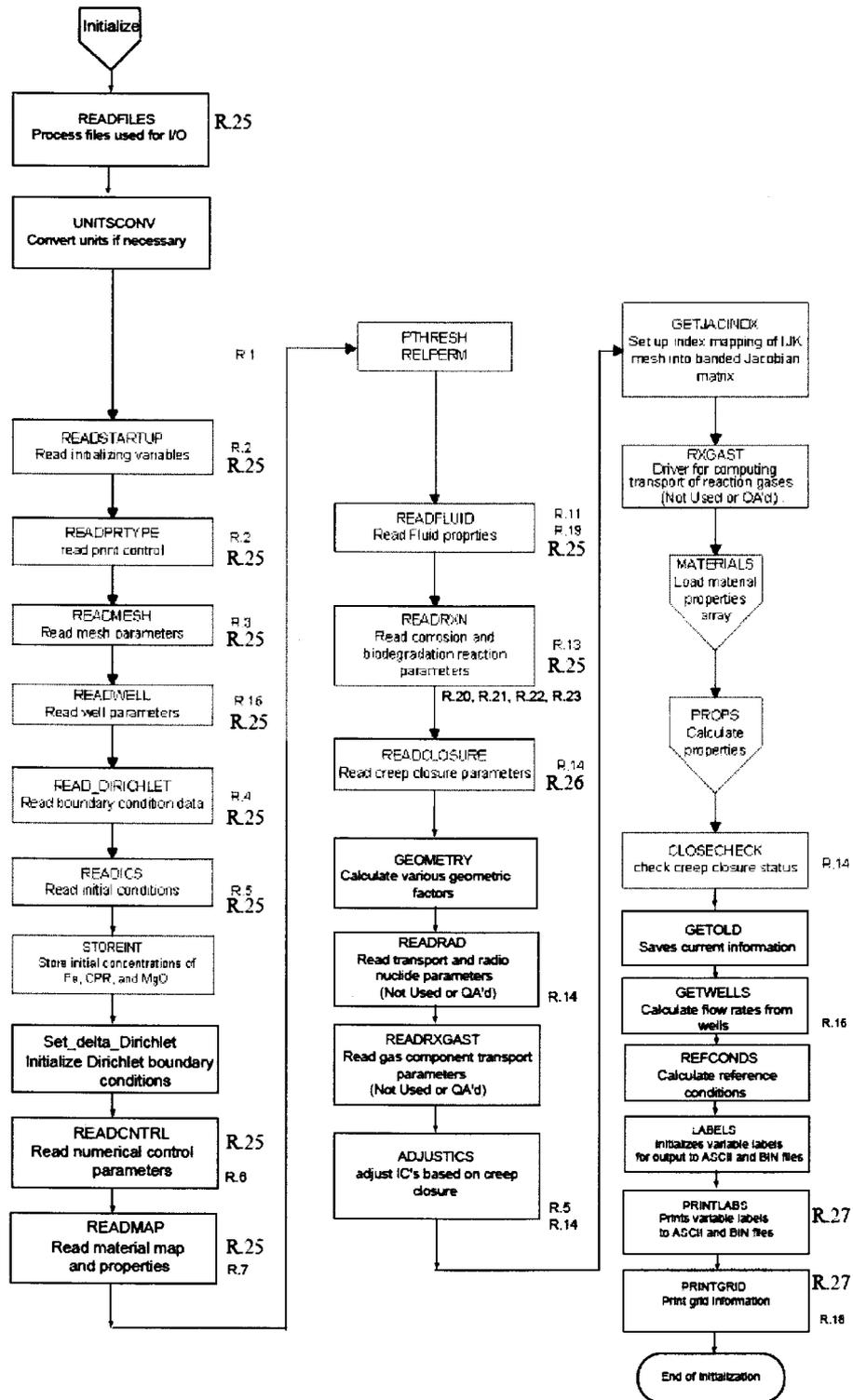


Figure 4. Flow chart for initialization phase.



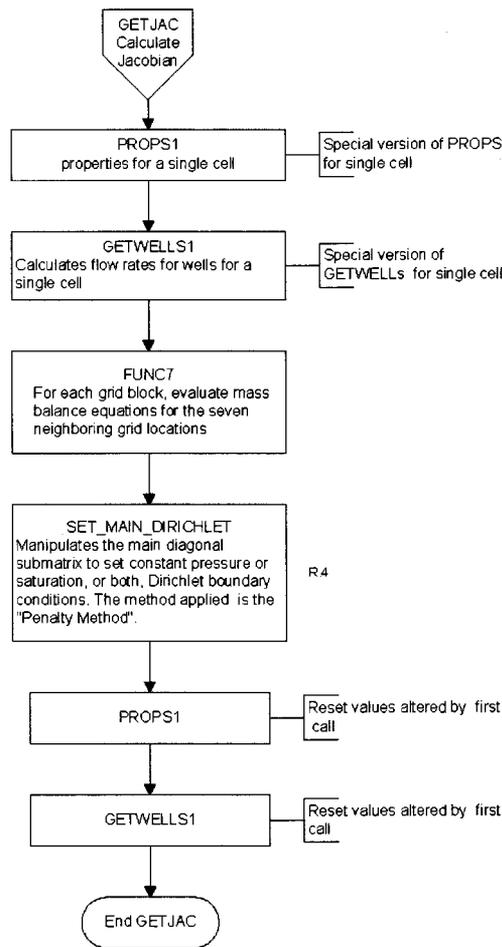


Figure 6. Flow chart for Jacobian phase.

## 5.2 Call Tree with Subroutine Descriptions

The following is the BRAGFLO call tree. Descriptions of the routines can be found in Subsection 5.3. An indentation indicates a call to the named routine. For example, BRAGFLO calls DATNTIM and INITIAL. INITIAL calls READFILES and UNITSCON etc. If a routine is used more than once, it will be referenced by the subroutine's number in subsequent calls later in the tree. For example, consider DATNTIM, line 2, in the tree. DATNTIM appears again (in line 7) with a descriptor of "See 2". The sub-tree underneath DATNTIM appears only in the first occurrence of DATNTIM in the table (line 2) but is also valid elsewhere (i.e. line 7). The combination of the flow charts in Section 5.1 and the following call tree provides a complete description of the BRAGFLO flow logic. Routines that are either not used or qualified for WIPP compliance calculations are identified but are included in the table for completeness.

**Routine**

BRAGFLO

**\*\*\*Initialization\*\*\***

1. . . BLOCK\_DATA
2. . . DATNTIM
3. . . TTIME
4. . . INITAL
5. . . READFILES
6. . . . QABGNL
7. . . . . DATNTIM (SEE 2)
8. . . UNITSCONV
9. . . READSTARTUP
10. . . READPRTYPE
11. . . . PRORDER
12. . . . LABELS
13. . . READMESH
14. . . READWELL
15. . . READ\_DIRICHLET
16. . . READICS
17. . . STOREINT
18. . . SET\_DELTA\_DIRICHLET
19. . . READCNTRL
20. . . READMAT
21. . . READFLUID
22. . . . DENGZ
23. . . . REFCONDS
24. . . . . DENO
25. . . . . DENGZINT
26. . . . . DENGZ
27. . . READRXN
28. . . READCLOSURE
29. . . GEOMETRY
30. . . READRAD
31. . . READRXGAST

- 32. . . . ADJUSTICS
- 33. . . . GETJACINDX
- 34. . . . RXGAST
- 35. . . . BANDSTORE
- 36. . . . GEN\_COEF
- 37. . . . BANDIT
- 38. . . . MATERIALS
- 39. . . . PERMS
- 40. . . . . KLINKBERG
- 41. . . . . CLOSCHECK
- 42. . . . . ROCKCOMP
- 43. . . . . PERMPHI
- 44. . . . . PTHRESH
- 45. . . . RELPERM
- 46. . . . DENSAT1
- 47. . . . . DENGZINT
- 48. . . . . DENGZ
- 49. . . . . DENO
- 50. . . . CONSOL
- 51. . . . . POR\_SURF
- 52. . . . . BL\_INTERP
- 53. . . . . LOCATE
- 54. . . . . TRI\_INTERP
- 55. . . . CONSOL1
- 56. . . . . POR\_SURF (SEE 51)
- 57. . . . PTHRESH
- 58. . . . RELPERM
- 59. . . . DENSAT1 (SEE 46)
- 60. . . . DISOLVGAS (NOT USED)
- 61. . . . . DENO
- 62. . . . . BUBBLE
- 63. . . . MAPDEP
- 64. . . . PROPS
- 65. . . . DEPINVERT
- 66. . . . SMOOTHPERM
- 67. . . . PTHRESH

- 68. . . . RELPERM
- 69. . . . DENGZINT
- 70. . . . DENGZ
- 71. . . . DENO
- 72. . . . CLOSCHECK
- 73. . . . CONSOL (SEE 50)
- 74. . . . CONSOL1 (SEE 55)
- 75. . . . ROCKCOMP (SEE 42)
- 76. . . . RADIOLYSIS
- 77. . . . . DECAY
- 78. . . . . WHICH
- 79. . . . . WHERE
- 80. . . . KLINKBERG
- 81. . . . DISOLVGAS (SEE 60) (Not Used)
- 82. . . . CORSAT
- 83. . . . CORHUM
- 84. . . . CORMGO
- 85. . . . HYDROCONV
- 86. . . . BIOSAT
- 87. . . . BIOHUM
- 88. . . . BIOFES
- 89. . . . BIOMGO
- 90. . . . RPGM\_INTRFC
- 91. . . . . GASC
- 92. . . CLOSCHECK
- 93. . . GETOLD
- 94. . . GETWELLS
- 95. . . REFCONDS (SEE 23)
- 96. . . LABELS1
- 97. . . PRINTLABS
- 98. . . PRINTGRID
- 99. . . WRITEARRAY
- 100. . . WRITBIN
- 101. . . . ALI2VAX
- 102. . . . ALR2VAX

\*\*\*End Main Initialization\*\*\*

**\*\*\*Update Initialization If A Restart File Is To Be Used\*\*\***

107 . DISKR  
108 . . MATERIALS (SEE 38)  
109 . . PERMS (SEE 39)  
110 . . GETWELLS  
111 . . GETOLD  
112 . . TIMESTEP  
113 . . . FMIN  
114 . RESETMID  
115 . PRINTCONTROL  
116 . . PRINTHIV  
117 . . . WRITHIV  
118 . . TSAVG  
119 . . PRINTASC  
120 . . . WRITEARRAY  
121 . . . PORSOLID  
122 . . PRINTBIN  
123 . . . . ALI2VAX  
124 . . . . ALR2VAX  
125 . . . . PORSOLID  
126 . . . WRITBIN (SEE 100)  
127 . . . TSAVG0  
128 . . DISKW  
129 . . PRINTHIV  
130 . . TSAVG  
131 . . PRINTASC  
132 . . PRINTBIN (SEE 122)  
133 . . DISKW

**\*\*\*End Restart Initialization Update\*\*\***

**\*\*\*Begin Solution Integration Main Loop\*\*\***

128. . SOLVER  
129. . . MAPDEP

**\*\*\*Begin Integration Across A Timestep Loop\*\*\***

130. . . RAPHSON

- 131. . . . PROPS (SEE 64)
- 132. . . . GETWELLS
- 133. . . . RESIDUAL
- 134. . . . FUNCT
- 135. . . . CLOSCHECK

**\*\*\*Begin The Newton-Raphson Intra-Timestep Iterations For The Current Timestep\*\*\***

- 136. . . . PROPS (SEE 64)
- 137. . . . GETWELLS
- 138. . . . RESIDUAL
- 139. . . . DELTAMAP
- 140. . . . GETJAC
- 141. . . . PROPS1
- 142. . . . . DEPINVERT1
- 143. . . . . PTHRESH
- 144. . . . . RELPERM
- 145. . . . . DENGZINT
- 146. . . . . DENGZ
- 147. . . . . DENO
- 148. . . . . CONSOL1 (SEE 55)
- 149. . . . . ROCKCOMP (SEE 42)
- 150. . . . . RADIOLYSIS (SEE 76)
- 151. . . . . KLINKBERG
- 152. . . . . DISOLVGAS (SEE 60) (NOT USED)
- 153. . . . . CORSAT
- 154. . . . . CORHUM
- 155. . . . . CORMGO
- 156. . . . . HYDROCONV
- 157. . . . . BIOSAT
- 158. . . . . BIOHUM
- 159. . . . . BIOFES
- 160. . . . . BIOMGO
- 161. . . . . RPGM\_INTRFC (SEE 90)
- 162. . . . . GETWELLS1
- 163. . . . . FUNCT7
- 164. . . . . SET\_MAIN\_DIRICHLET

- 165. . . . . PROPS1 (SEE 141)
- 166. . . . . GETWELLS1
- 167. . . . . SET\_RHS\_DIRICHLET

**\*\*\*Begin Jacobian Matrix Solution, Only One Method Is Done Choices Are 'A' Through 'D'\*\*\***

- 168. . . . . SOLUTION

**\*\*\*A: IF LINPAK Is Selected\*\*\***

- 169. . . . . DGBCO
- 170. . . . . DAXPY
- 171. . . . . DDOT
- 172. . . . . DSCAL
- 173. . . . . DASUM
- 174. . . . . DGBFA
- 175. . . . . DSCAL
- 176. . . . . IDAMAX
- 177. . . . . DAXPY
- 178. . . . . DGBSL
- 179. . . . . DDOT
- 180. . . . . DAXPY

**\*\*\*End LINPAK Section\*\*\***

**\*\*\*B: IF L/U Is Selected\*\*\***

- 181. . . . . DECBR
- 182. . . . . SOLBR

**\*\*\*End L/U Section\*\*\***

**\*\*\*C: If MULTIGRID Is Selected\*\*\***

- 183. . . . . MULTIGRID

**\*\*\*End MULTIGRID Section\*\*\***

**\*\*\*D: IF SOR Is Selected\*\*\***

- 184. . . . . SOR

**\*\*\*End SOR Section\*\*\***

\*\*\*End Jacobian Matrix Solution Section\*\*\*

\*\*\*Continue In The Newton-Raphson Intra-Timestep Iterations For The Current Timestep Section\*\*\*

185. . . . PROPS1 (SEE 141)  
186. . . . PROPS1 (SEE 141)  
187. . . . PROPS (SEE 64)  
188. . . . GETWELLS  
189. . . . RESIDUAL  
190. . . . RXGAST (SEE 34)  
191. . . . MEAS\_CONV  
192. . . . CLOSCHECK

\*\*\*End of Newton-Raphson Intra-Timestep Iteration Loop\*\*\*

\*\*\*Continue In Integration Across A Timestep Loop\*\*\*

193. . . . MEAS\_CONV

\*\*\*End of Integration Across a Timestep Loop\*\*\*

\*\*\*Continue In Solution Integration Main Loop\*\*\*

194. . CUMULGEN  
195. . RADTRANS  
196. . . FMAX  
197. . . DECAY (SEE 77)  
198. . MASSBALNC  
199. . DATNTIM (SEE 2)  
200. . SUMMARY  
201. . RESET  
202. . . PTHRESH  
203. . . RELPERM  
204. . . DENSAT1 (SEE 46)  
205. . . DISOLVGAS (SEE 60) (NOT USED)  
206. . . ROCKCOMP (SEE 42)  
207. . PRINTCONTROL  
208. . TIMESTEP  
209. . PRINTCONTROL

\*\*\*Change Materials If Needed\*\*\*

210. . MATERIALS (SEE 38)

\*\*\*Continue In Solution Integration Main Loop\*\*\*

Information Only

211. . PRINTCONTROL

212. . GETOLD

\*\*\*End Solution Integration Main Loop \*\*\*

\*\*\*If End of Simulation Time Is Reached Write QA Stamp and Exit\*\*\*

213. . DATNTIM (SEE 2)

214. . QAENDL

### 5.3 BRAGFLO Routines and Descriptions

The number of routines in BRAGFLO Version 6.02 is greater than the number in the previously qualified version 6.0. BRAGFLO Version 6.02 consists of 122 various subroutines, functions, and a BLOCK DATA routine. The following is a list of all of the BRAGFLO Version 6.02 Subroutine, Functions, and other routines with a brief description. Routines that were either not used or not qualified for WIPP compliance calculations are indicated.

#### SUBROUTINES

- 1) SUBROUTINE ADJUSTICS (ICWASTE,POWASTEIC,SOWASTEIC)  
Adjusts initial conditions, depending on whether or not closure is to be used.
- 2) SUBROUTINE ALI2VAX (N,INBUF,OUTBUF)  
Swaps bytes to convert Alliant INTEGER\*4 word into VAX INTEGER\*4
- 3) SUBROUTINE ALR2VAX (N,INBUF,OUTBUF)  
Swaps bytes to convert Alliant (IEEE) REAL\*4 word to VAX REAL\*4.
- 4) SUBROUTINE BANDIT (IBW,N,C,B) (Not Used)  
Band solver with row interchange.
- 5) SUBROUTINE BANDSTORE (Not Used)  
Assign locations to pointer array, IBS. This directs the storage of the coefficients in the banded matrix structure.
- 6) SUBROUTINE BIOFES (RXBIOS,RXBIOH,FEI,FEOH2,CONCFE,DELT,SFEWMOL,  
& SFEOH2WMOL,RXH2S,RXFEOHH2S,RXFEH2S,LARXN2,ALPHARXN)  
Calculates the iron and iron hydroxide sulfidation rates
- 7) SUBROUTINE BIOMGO (RXBIOS,RXBIOH,RXCO2,CONCMGO,CMGOH2,CONCMGOI,  
& SMGOWMOL,SMGOH2WMOL,SOACT,DELT,RXMGOHCO2,RXMGOCO2,LARXN2,  
& ALPHARXN)  
Calculates the rate of carbonation of MgO and magnesium hydroxide
- 8) SUBROUTINE BL\_INTERP (X,Y,F,XP,YP,Z)  
Bilinear interpolation of F at XP, YP over four points (X(I),Y(I)),  
I = 1,4. Returns interpolated value Z.
- 9) SUBROUTINE BUBBLE (SG,SO,ROWG,ROWO,GOR,PBUBBLE) (Not Used)  
Calculates bubble pressure point.
- 10) SUBROUTINE CLOSCHECK (CLOSTEMP,CLOSTEMP1,SUBNAME,KLOSTEST,LPT)

Information Only

Checks whether closure gets turned off, either temporarily or permanently.

11) SUBROUTINE CONSOL (MKL, TTIME)

Consolidation / creep closure function.

Note: If there is more than one waste region (NWST>1), all waste regions are lumped together to determine closure behavior, so all waste regions will behave identically.

Waste porosity is obtained from "porosity surface" generated by SANCHO simulations of creep closure when gas is generated at various fixed rates. Porosity is a function of time and moles of gas present in the waste, with gas generation rate as a parameter. To obtain porosity, 2-D interpolation is done at the current time. Interpolation can be based on the gas generation rate or the average pressure in the waste. An input flag, KLOSINT, determines which interpolation is used.

Original SANCHO porosity, based on current room volume, has been converted to BRAGFLO porosity, based on initial room volume, before inserting into DATA statements in BLOCK DATA. Note that raw data for Surface #2 (North End) were room volumes (SANCHO porosities of 1.0); permeability function for that surface uses BRAGFLO porosities instead of SANCHO porosities.

12) SUBROUTINE CONSOL1 (MKL, TTIME, I, J, K)

Consolidation/creep closure function.

This version of CONSOL calculates porosity when the "porosity surface" is applied to individual grid blocks, instead of to entire regions having average properties over the region.

Waste porosity is obtained from "porosity surface" generated by SANCHO simulations of creep closure when gas is generated at various fixed rates. Porosity is a function of time and moles of gas present in the waste, with gas generation rate as a parameter. To obtain porosity, 2-D interpolation is done at the current time.

Interpolation can be based on the gas generation rate or the average pressure in the waste. An input flag, KLOSINT, determines which interpolation is used.

Original SANCHO porosity, based on current room volume, has been converted to BRAGFLO porosity, based on initial room volume, before inserting into DATA statements in BLOCK DATA. Note that raw data for Surface #2 (North End) were room volumes (SANCHO porosities of 1.0); permeability function for that surface uses BRAGFLO porosities instead of SANCHO porosities.

13) SUBROUTINE CORMGO (BRUCITEI, BRUCITEH, CONCMGO, CONCMGOI, SMGOWMOL, & SOACT, DELT, RXMGOH2O, LINTRIN, LARXN, LARXN2, ALPHARXN)

Calculates the humid and inundated rates of MgO hydration

14) SUBROUTINE CUMULGEN

Updates concentrations, calculates cumulative quantities, and calculates flows and Darcy velocities. Quantities obtained are general, i.e., for any mesh.

15) SUBROUTINE DATNTIM (IN, IDATE, ITIME, DAYS, HOUR, MINUTE, SECOND, SECTOT)

Gets Date & Time from machine-dependent system subroutines.

Library subroutines:

EXDATE: Returns date in format "mm-dd-yy"  
EXTIME: Returns time in format "hh:mm:ss"  
EXCPUS: Returns cpu time elapsed since program began,  
in units of 0.01 second  
EXPARM: Returns system information: name of processor,  
name of the machine, and operating system.

Alliant subroutines: [in /usr/lib/libU77.a]  
ETIME: Returns total elapsed time since program began [s]  
CTIME: Returns date & time in format "www mmm dd hh:mm:ss  
YYYY"  
TIME: Returns total seconds since 31-DEC-69 19:00:00

Paragon subroutines:  
TIMER: Returns elapsed CPU time; also initializes timer.

- 16) SUBROUTINE DAXPY(N,DA,DX,INCX,DY,INCY) (Not Used)  
Constant times a vector plus a vector.  
Uses unrolled loops for increments equal to one
- 17) SUBROUTINE DECAY (TIME,XLOAD,XNOW,IDN) (Not Used)  
Calculates radioactive decay; obtained from program PANEL.
- 18) SUBROUTINE DECBR (IER)  
This subroutine constructs the LU decomposition of a band matrix  
A in the form  $L*U = P*A$ , where P is a permutation matrix, L is a  
unit lower triangular matrix, and U is an upper triangular  
matrix.  
The matrix A is assumed here to be stored by rows in the array B.
- 19) SUBROUTINE DELTAMAP(DELTA,DUM)  
Convert variable changes to correspond to mapping used in  
MAPDEP. The values of the DELTA array are used for evaluating  
the partial derivatives that form the Jacobian.  
These DELTAs may be different for each dependent variable but  
are the same for a given variable irrespective of grid block  
on a relative basis.
- 20) SUBROUTINE DENSAT1 (I,J,K)  
Calculates gas & brine density for a single grid block.
- 21) SUBROUTINE DEPINVERT (DEP)  
Convert DEP array into individual dependent variable arrays  
according to the same mapping as in MAPDEP.
- 22) SUBROUTINE DEPINVERT1 (DEP,I,J,K)  
Convert DEP array into individual dependent variable arrays  
according to the same mapping as in MAPDEP.  
Single-grid-block version of DEPINVERT.
- 23) SUBROUTINE DGBCO(ABD,LDA,N,ML,MU,IPVT,RCOND,Z) (Not Used)  
Linpack version of the L/U direct solver.  
Only the following routines from general library are contained  
here. (Just the routines needed in BRAGFLO)  
Routines: DGBCO,DGBSL,DASUM,DGBFA,DSCAL,DDOT,DAXPY,IDAMAX

Reference: Linpack User's Guide  
J.J. Dongarra, C.B. Moler, J.R. Bunch and G.W. Stewart  
SIAM, Philadelphia, PA 1979.

- 24) SUBROUTINE DGBFA (ABD,LDA,N,ML,MU,IPVT,INFO) (Not Used)

Information Only

DGBFA factors a double precision band matrix by elimination.

25) SUBROUTINE DGBSL (ABD, LDA, N, ML, MU, IPV, B, JOB (Not Used))

DGBSL solves the double precision band system  
 $A * X = B$  or  $TRANS(A) * X = B$   
using the factors computed by DGBCO or DGBFA.

26) SUBROUTINE DISKR (Not Used)

This subroutine reads the restart file that was written by Routine DISKW in a previous run. This routine is called after Routine INITIAL reads the entire input file, thus overwriting the Variables and arrays which will define the state of the calculation when the restart file was written by DISKW. This includes the dependent variables, their previous values (OLD) and any accumulation terms. In this way, TIME, DELT, IPRINT are overwritten with the restart values. If for some reason different values are needed, the user must define them, either at the end of this routine, or in the calling program.

The first read is information about the database saved, to see if it is the correct one. If, say, the dimensions do not match, the user may have assigned the wrong restart file, so the program is terminated. Any information to test on can be added here as well as in DISKW.

27) SUBROUTINE DISKW (Not Used)

This subroutine writes information to disk so that a restart would be possible in the future. Enough information has to be saved such that identical answers are generated whether using this restart capability or not. The approach taken is to write just the variables and arrays needed to perform the restart.

The first write contains information that defines the database written. It contains the dimension size, title, time and time step. When read, these quantities can be checked to make sure that the restart file corresponds to the current job.

28) SUBROUTINE DISOLVGAS (I, J, K, IJAC) (Not Used)

Calculates CGO & COO terms (mass fraction of dissolved gas in brine phase and mass fraction of brine in brine phase, resp.), using different approaches to account for dissolved gas:  
(1) Bubble point tracking, with equilibrium gas solubility defined by RSO parameters and correlation;  
(2) Henry's law equilibrium gas solubility, with no bubble point tracking;  
(3) No dissolved gas.

29) SUBROUTINE DSCAL (N, DA, DX, INCX) (Not Used)

Scales a vector by a constant.  
Uses unrolled loops for increment equal to one.

30) SUBROUTINE FUNCT (I, J, K, F)

Evaluates PDE's for system to be solved.  
Current set up -- ordering and dependent variable association:

Function	Description	Dependent Variable
F(1)	Gas phase	SG (gas saturation)
F(2)	Brine phase	PO (brine pressure)

31) SUBROUTINE FUNCT7 (II, JJ, KK, F7)

Evaluates PDE's for system to be solved.  
Does 7 evaluations for the 7-spot in a single call, saving  
results in F7 array.  
Current set up -- ordering and dependent variable association:

Function	Description	Dependent Variable
F(1)	Gas phase	SG (gas saturation)
F(2)	Brine phase	PO (brine pressure)

- 32) SUBROUTINE GASC (DT, NN, NL) (Not Used)  
This is a temporary placeholder.
- 33) SUBROUTINE GEN\_COEF(L) (Not Used)  
Generates coefficients for component gas transport.
- 34) SUBROUTINE GEOMETRY  
Calculates various geometric factors that are used repeatedly  
throughout rest of code, and so should be done just once, here.
- 35) SUBROUTINE GETJAC (DX, DEPOLD, CLOSTEMP, CLOSTEMP1)  
Forms banded Jacobian.
- 36) SUBROUTINE GETJACINDX  
Sets up indices for mapping I-J-K mesh into banded Jacobian  
matrix.
- 37) SUBROUTINE GETOLD  
Update old values for next time step with current converged  
values explicit dating could be done here.
- 38) SUBROUTINE GETWELLS  
Calculate flow rates from production and injection wells.
- 39) SUBROUTINE GETWELLS1 (I, J, K)  
Calculate flow rates from production and injection wells.  
Used only in evaluating the Jacobian; applied to a single  
grid block.
- 40) SUBROUTINE HYDROCONV (HYMAGCON, CONCMGOI, CONCHYDRO, SHYDROMOL,  
& DELT, RXHYMAG, LARXN2, ALPHARXN)  
Returns the hydromagnesite conversion reaction rates.
- 41) SUBROUTINE INITAL  
Makes calls to read input file and do initial processing of input  
data.
- 42) SUBROUTINE KLINKBERG (BKLINK, EXPKLINK, PG, PERMBX, PERMBY, PERMBZ,  
PERMGX, PERMGY, PERMGZ, KLINK)  
Modify gas permeabilities for Klinkenberg effect;  
use gas pressure (PG) to determine gas phase permeabilities.
- 43) SUBROUTINE LABELS  
Initializes ASCII printout titles & units labels, and values of  
units conversions, depending on output units system specified.  
Prepares variable names that POSTBRAG can use and computes units  
conversion factors from UNITSO to SI units (will be 1.0 if UNITSO  
is 'SI') for use by POSTBRAG.
- 44) SUBROUTINE LOCATE (X, Y, XP, YP, INTERIOR, NODE)  
Determine if (XP, YP) is to the left of each line connecting nodes



- Checks to see if printing is required. If required, calls printing routine for ASCII and/or Binary and/or Restart output.
- 56) SUBROUTINE PRINTGRID (IDEPTHFLAG)  
Prints out grid information: dimensions, grid block volumes, and x-, y-, and z-coordinates of grid blocks.
- 57) SUBROUTINE PRINTHIV (LPRBIN)  
Prints out history variables to binary output file.
- 58) SUBROUTINE PRINTLABS  
Prints out variable labels to output files; prints only those variables that are actually used, as specified by MPRNTA and MPRNTB flags. Also prints out labels for history variables and global variables.
- 59) SUBROUTINE PROPS (DEP,IUPFIRST,IUPMASSFR,CLOSTEMP,CLOSTEMP1)  
Calculates reservoir and material properties over the entire mesh.
- 60) SUBROUTINE PROPS1 (I,J,K,DEP,CLOSTEMP,CLOSTEMP1)  
Single-grid-block version of PROPS, called only when evaluating the Jacobian in GETJAC.
- 61) SUBROUTINE PRORDER (TIMES,NT,NMAX,TI)  
Arranges times in ascending order and eliminates duplicate times. Also, eliminates print times that are earlier than starting time, TI, and adjusts the number of output times, NT, accordingly.
- 62) SUBROUTINE PTHRESH (MAT,PERMBX,PTC)  
Calculates threshold capillary pressure as a function of permeability. Permeability used is brine phase permeability in the x-direction.
- 63) SUBROUTINE QABGNL (IUNIT,FILTYP)  
Produces QA info at the beginning of a program run.
- 64) SUBROUTINE QAENDL (IUNIT,FILTYP)  
Produces a QA line at the end of a program run.
- 65) SUBROUTINE RADIOLYSIS (DDT,DBVOL,XLOAD,XNOW,H2PROD,NRAD) (Not Used)  
Calculates amount of H2 produced by radiolysis, with decay of radionuclide inventory.
- 66) SUBROUTINE RADTRANS (Not Used)  
Performs rudimentary transport/particle tracking to allow movement of contaminated brine from the waste regions to be tracked. Uses inter-block flows with FINITE source in the waste to estimate the relative concentrations of contaminants everywhere in the mesh. Assumes instantaneous dissolution of radionuclides up to a solubility limit.
- 67) SUBROUTINE RAPHSON (DEPOLD,DEPOUT,SGVEL,POVEL,GASVEL,IVEL,JVEL,KVEL,MVEL,EPSMAX,FTOLMAX,IFTOL,JFTOL,KFTOL,IEPS,JEPS,KEPS,IER)  
This routine formulates and controls the Newton-Raphson solution method. The Jacobians are computed numerically in GETJAC and solved in SOLUTION by either LU decomposition (CHSOLVER='LU'), multi-grid (CHSOLVER='MG'), by LINPACK LU decomposition (CHSOLVER='LP'), or SOR (CHSOLVER='SR').

- 68) SUBROUTINE READCLOSURE  
Read input data for creep-closure-related parameters.
- 69) SUBROUTINE READCNTRL  
Read input data for numerical control.
- 70) SUBROUTINE READFILES  
Processes files for I/O: Names and opens files, does inquiries, and writes names to output files. Also writes PREBRAG and BRAGFLO QA info to output files (versions, revision dates, run dates and times).
- 71) SUBROUTINE READFLUID  
Read input data for fluid properties.
- 72) SUBROUTINE READICS  
Read input data for initial conditions: pressure, saturation, Fe concentration, and C6-H10-O5 concentration.
- 73) SUBROUTINE READMAT (ICWASTE, POWASTEIC, SOWASTEIC)  
Read input data for material map and properties, including parameters for fracture model and Klinkenberg effect.
- 74) SUBROUTINE READMESH (IDDEPTHFLAG)  
Read input data for mesh setup.
- 75) SUBROUTINE READPRTYPE  
Read input data for printout control.
- 76) SUBROUTINE READRAD (Not Used)  
Read input data for transport and radionuclide decay: various control parameters, nuclide ID's, daughter ID's, half-lives, atomic weights, initial inventory.
- 77) SUBROUTINE READRXGAST (Not Used)  
Read input data for gas component transport.
- 78) SUBROUTINE READRXN  
Read input data for corrosion and biodegradation reaction parameters.
- 79) SUBROUTINE READSTARTUP  
Read input data for starting up a run -- all the stuff from the top of the input file: Model type (1-, 2-, or 3-D), grid size, starting time, initial time step, etc.
- 80) SUBROUTINE READWELL  
Read input data for wells.
- 81) SUBROUTINE READ\_DIRICHLET  
Initializes and reads blocks that have constant Dirichlet boundary conditions.
- Note that brine saturation is read in, whereas gas saturation is the dependent variable being solved for, so it is DELTA\_SATG\_DIR (gas saturation changed) that is held constant in the Dirichlet BC treatment.
- 82) SUBROUTINE REFCONDS (KFLAG)  
Calculates reference conditions: original brine and gas in place, initial grid volumes, etc. Then prints out this

information.

- 83) SUBROUTINE RELPERM (SW, SG, SWR, SGR, PT, SETERM, SETERM2, XLAM1, XLAM2, XLAM3, XLAM4, XLAM5, XLAM6, XLAM7, PCFIX, POMIN, SEMIN, IPCFLG, PO, IPRINT, KRP, KPC, KRW, KRNW, PC)  
Calculates relative permeability and capillary pressure. Several models and correlations are available, including Brooks-Corey and van Genuchten/Parker correlations and various linear models. Capillary pressures can be computed without constraints on the results (potentially risky numerically), or subject to maximum values or minimum effective saturations. Choices are explained in READMAT, where input parameters are read in.
- 84) SUBROUTINE RESET  
Resets pressures and saturations at the end of the startup procedure, in NMATRESET material regions. Material region numbers are in array MATRESET. Also adjusts porosity of those regions and fixes T3 to account for the step change in porosity and saturation. Finally, adjusts saturations in DRZ.
- 85) SUBROUTINE RESETMID (LRESET)  
Resets pressure, saturation, and concentrations in a borehole material after an intrusion
- 86) SUBROUTINE RESIDUAL (NEEDRES)  
Calculates residual arrays, RES and FF2.  
The residual, RES, is equal to the function, FF2, normalized by dividing by the amount of phase present in a grid block, T3, as long as T3 is greater than T3ZERO, which is set in BLOCK DATA. T3ZERO is an arbitrarily small amount (say, 1.E-15) that prevents Over-flow or excessive weighting by a quantity of phase that is probably round-off or noise.
- 87) SUBROUTINE ROCKCOMP (PHIZERO, CROCKL, PZERO, P, PHI, MAT, PERMBX, PERMBY, PERMBZ, PERMGX, PERMGY, PERMGZ, II, JJ, KK)  
Formation compressibility effect on porosity.
- 88) SUBROUTINE RPGM\_INTRFC (DELT, I, J, K, POREVOL) (Not Used)  
Interface between the Reaction Path Gas generation Model (GASC) and BRAGFLO.
- 89) SUBROUTINE RXGAST (IPASS) (Not Used)  
Driver for computing the transport of the reaction gases.
- 90) SUBROUTINE SET\_DELTA\_DIRICHLET  
Sets the delta pressure and saturation required to keep constant boundaries. This is necessary because the solution is for the change of the primary variables (delta) and not for the variables themselves.
- 91) SUBROUTINE SET\_MAIN\_DIRICHLET (LL, I, J, K, IEQ)  
Manipulates the main diagonal submatrix to set constant pressure or saturation, or both, Dirichlet boundary conditions. The method applied here is the "Penalty Method". The idea is to make the main diagonal "very large" compared with the nondiagonal elements, then multiply the right-hand-side by the main diagonal element value, so that the solution for that variable is the original right hand side. To make the main diagonal very large, it is first multiplied by a large number (PENFAC=1.E8); in case the diagonal is already very small, the same large number is

Information Only

- added to the diagonal element to ensure that it is large.
- 92) SUBROUTINE SET\_RHS\_DIRICHLET  
Manipulates the Right Hand Side (RHS) to set constant Dirichlet boundary conditions. The method applied here is the "Penalty Method".
- 93) SUBROUTINE SMOOTHPERM  
Smoothly changes the permeability of a material during a material change over a finite time interval
- 94) SUBROUTINE SOLBR  
This subroutine computes the solution of the banded linear system  $A * X = C$ , given the LU decomposition of the matrix A from DECBR.
- 95) SUBROUTINE SOLUTION (IJACOBIAN, IER)  
Driver for calling the various solvers currently available.
- 96) SUBROUTINE SOLVER (IER)  
Driver for Newton-Raphson iteration solution technique.
- 97) SUBROUTINE SOR (NDIM, AE, AW, AN, AS, AP, BE, BW, BN, BS, BP, CE, CW, CN, CS, CP, DE, DW, DN, DS, DP, YY, LDEPMAP, NX, NY, LSETZ) (Not Used)  
This routine solves a block matrix equation by a point successive-over-relaxation technique. Boundary conditions are assumed to be built into the coefficients.
- 98) SUBROUTINE STOREINT  
Stores the initial concentrations of iron, MgO, and CPR
- 99) SUBROUTINE SUMMARY  
Prints out 1-line summary at each time step along with certain results at the monitor blocks.
- 100) SUBROUTINE TIMESTEP (IPCOUNTASC, IPCOUNTBIN, IPCOUNTRST)  
Determine time step size to be used for next time step using:  $\text{MIN}((2.0 * \text{DELTADEPNORM}) / (\text{DELTADEPNORM} + \text{DELTADEPMAX}), \text{DELTADEP})$   
Each dependent variable may have different DELTADEPNORM and DELTADEPMAX,  
where: DELTADEPNORM is specified in input for each dependent variable and represents the largest change in the dep. variable across a time step such that the new time step is unchanged.  
DELTADEPMAX is the calculated maximum change in a dependent variable across the previous time step over all grid blocks, either DELTASATMAX, DELTAPAMAX, or DELTAPRESMAX.  
Last time-step selected such that TIME=FINISH.
- 101) SUBROUTINE TRI\_INTERP (X, Y, F, XP, YP, Z)  
Interpolation of F at XP, YP over three points (X(I), Y(I)), I = 1, 3. Returns interpolated value Z.
- 102) SUBROUTINE TSAVG  
Computes time-step averages for velocities and inter-block flows.
- 103) SUBROUTINE UNITS CONV  
Initializes & calculates units conversion factors for converting between SI & English units.

- 104) SUBROUTINE WHERE (ION, IDN ,IDX) (Not Used)  
Determines identifier of the stable isotope that terminates the decay chain.
- 105) SUBROUTINE WHICH (TIME, ID, IDN ,ID1, ID2, ID3, ID4) (Not Used)  
Determines which isotopes to include in the chain.  
Each isotope can be the parent of up to four generations.  
The BATEMAN equation is used to determine the decay.  
If the time-step is too large, four generations may not be sufficient.
- 106) SUBROUTINE WRITBIN (RVARI,UNTCNV)  
Writes output to binary file, in a manner depending on the machine used. In order to transfer files from Alliant to VAX, binary files written on Alliant must have fixed-length records of length 512 bytes, with each byte written in VAX format. When running on Alliant, WRITBIN calls subroutines ALI2VAX and ALR2VAX to convert Alliant words to VAX words before they are written to the binary output file.
- 107) SUBROUTINE WRITEARRAY (ARRAY,UNTCNV,PRL,UNTS)  
Prints array values to ASCII output file.
- 108) SUBROUTINE WRITHIV (NV,VAR,LHI,LHF,R4VARI)  
Puts history variables into single vector for simpler printout. Also does units conversion and converts to REAL\*4.

#### FUNCTIONS

- 109) FUNCTION BIOHUM (RKHUM,CONCBIO,SCELWMOL,SOACT,DELT,LARXN,ALPHARXN)  
Biodegradation reaction rate (unsaturated).  
Unsaturated rate is assumed to be zero order.
- 110) FUNCTION BIOSAT (RKSAT,CONCBIO,SCELWMOL,SOACT,DELT,CONCBIOTEMP)  
Biodegradation reaction rate (saturated).  
Saturated rate is assumed to be zero order.
- 111) FUNCTION CORHUM (RKHUM,CONCFE,SFEWMOL,SOACT,DELT,LARXN,ALPHARXN)  
Corrosion reaction rate (unsaturated).  
Unsaturated rate is assumed to be zero order.
- 112) FUNCTION CORSAT (RKSAT,CONCFE,SFEWMOL,SOACT,DELT,CONCFETEMP)  
Corrosion reaction rate (saturated) [mol Fe/(s\*m<sup>3</sup>)]  
Saturated rate is assumed to be zero order.
- 113) DOUBLE PRECISION FUNCTION DASUM (N,DX,INCX) (Not Used)  
Takes the sum of the absolute values.
- 114) DOUBLE PRECISION FUNCTION DDOT (N,DX,INCX,DY,INCY)  
Forms the dot product of two vectors.  
Uses unrolled loops for increments equal to one.
- 115) FUNCTION DENGZ (NG,N1,T,PG,Y,Z,INTERPDENG)  
Computes density and compressibility factor for gases using:  
1) Redlich-Kwong-Soave Equation of State, or  
2) Ideal gas law.
- 116) FUNCTION DENGZINT (PG)  
Calculates gas density (DENGZ) [kg/m<sup>3</sup>] by interpolating linearly

Information Only

in a table of gas density computed using Redlich-Kwong-Soave EOS at fixed temperature as specified in input file.

Interpolator assumes table values of pressure are in 1.E4 Pa intervals:

```
PINT(1)      = 1.0E4
PINT(2)      = 2.0E4
.
.
PINT(NPINT) = DBLE(NPINT)*1.0E4
```

This enables the interpolator to locate the proper pressure table entries without having to search through the table.

If  $PG > NPINT$ , extrapolates from the 2 maximum table entries.  
If  $0 < PG < 1$ , interpolates between pressures of 0. and 1.0E4 Pa.  
If  $PG \leq 0$ , set density to 0.

- 117)       FUNCTION DENO (P,GBR,FVF)  
          Calculates brine density at pressure P at fixed temperature, using compressible liquid equation of state. Also calculates gas/brine ratio (fully-gas-saturated value) at P, and brine formation volume factor (neglecting the effect of dissolved gas). Properties evaluated at temperature 27 C.
- 118)       FUNCTION FMAX (X,N) (Not Used)  
          Find the maximum value of the one-dimensional array X.
- 119)       FUNCTION FMIN (X,N)  
          Find the minimum value of the one-dimensional array X.
- 120)       INTEGER FUNCTION IDAMAX (N,SX,INCX) (Not Used)  
          Finds the index of element having max. absolute value. Uses unrolled loops for increments equal to one.
- 121)       double precision function PORSOLID(i,j,k)  
          Calculates the volume fraction increase or decrease due to solids production.
- 122)       FUNCTION TTIME (I)  
          Calls Alliant subroutine TIME [in /usr/lib/libU77.a] to get total seconds since 31-DEC-69 19:00:00, for argument to subroutine TIME.

#### OTHERS

- 123)       BLOCK DATA STARTUP  
          Initializes variables in labeled COMMONs.

## 5.4 Data Flow

The flow of data through BRAGFLO occurs in two ways, either through FORTRAN common blocks that are included in the routines where the data is needed or the data is passed as arguments in the routines call list. Routines use one or both of the data flow methods. It was left to the programmer to determine which method was most suitable and was dictated by the quantity and type of data and the structure of the process being evaluated by the routine. Both methods are detailed below.

Information Only

### 5.4.1 Common Blocks

Data within BRAGFLO common blocks are closely related to specific physical process models or component areas and are grouped for convenience in handling. Table 1 provides an alphabetical list of all 70 named common blocks along with a brief description of the type of data stored.

Table 1. BRAGFLO Common Block Descriptions.

<u>Common Block</u>	<u>Description</u>
BAND	Band storage pointer array and equation index pointer array.
BASE	For Gas Generation Reaction Path Model.(Not QA'd or Used)
BF	Brine flow, production/consumption rates.
BRATE	Reaction path gas generation model parameters. (Not QA'd or Used)
BRINETERMS	Brine properties and parameters.
CCONVR	Control and monitoring parameters for used by subroutine SOLUTION.
CLOSURE	Creep closure data.
CMULTG	Conjugate gradient solver flags. (Not QA'd or Used)
CONC	Dissolved gas terms. (Not QA'd or Used)
CONPRE	Dirichlet boundary condition parameters.
CPRINT	CPU time parameters.
CPUNTC	Iteration counters for Multigrid solver. (Not QA'd or Used)
CRESTART	Restart control parameters. (Not QA'd or Used)
CUM	Cumulative component quantities.
DIR_MULTIPLIERS	Dirichlet boundary condition multipliers.
FLOWS	Phase component flows.
FRACTURE	Fracture model parameters.
FUN	Coefficients for PDEs to be solved
GASINTERMS	RKS Gas EOS interpolation table parameters.
GASTERMS	Gas parameters.
GEOTRANS	Geometric parameters at cell block interfaces used for harmonic averaging.
ICRESET	Parameters used for resetting Initial Conditions.
INITL	Initial saturation and pressure.
IO	I/O control parameters.
JACOBIAN	Parameters used to define Jacobian.
LU	LU decomposition solver parameters.
MACHINE	Flags for writing output on an Alliant computer.
MASSBAL	Brine and gas mass balance arrays and parameters..
MATTERMS	Material Properties arrays.
MODELTYPE	Geometry model flag.
MONITOR	Solution monitoring parameters.
MXTDIM	Bandwidth parameters.
NEWTON	Newton-Raphson control parameters.

<b>Common Block</b>	<b>Description</b>
NUCLIDE	Radionuclide tracking parameters and arrays. (Not QA'd or Used)
OLD	Parameter values at previous step.
PHYSCON	Physical constants.
PERMPARM	Parameters used for the new permeability characteristic curves described in Subsection 4.9
PRNTC	Character arrays used as printed labels.
PRNTI	Integer variables used to control printing..
PRNTR	Real arrays and parameter storage for printed output.
PROGRAM	Program QA parameters – name, version, revision data, etc.
PROP	Arrays for reservoir material properties.
PT	Potential and density arrays.
RADACT	For Gas Generation Reaction Path Model.(Not QA'd or Used)
RADCHAIN	Radionuclide properties. (Not QA'd or Used)
RATES	For Gas Generation Reaction Path Model.(Not QA'd or Used)
REF	Reference temperatures and densities.
REU	Initial concentrations of CPR, iron, and MgO
RELPERMS	Relative permeability parameters.
RKSEOS	RKS EOS parameters.
RPGM	Reaction path gas generation model parameters. (Not QA'd or Used)
RPGMCONC	Reaction path gas generation model - gas concentrations (Not QA'd or Used)
RPGMGR	Reaction path gas generation model- gas generation rates (Not QA'd or Used)
RPGMR	Reaction path gas generation model parameters. (Not QA'd or Used)
RSIZE	Grid description variables.
RSTBR	Parameters for the resetting of borehole pressure, saturation, and concentration as described in Subsection 4.17.4
RXN	Average Stoichiometry Gas Generation model parameters.
SATN	Saturation arrays and corrosion product concentrations.
SIZES	Array sizing parameters.(Version 4.30)
SMPRM	Parameters for the smooth change in permeability as described in Subsection 4.18
SUBS	Reaction path model parameters (Not QA'd or Used)
SWITCH	Variable switching maps.
T	Various time related variables.
TERMS	Harmonically averaged transmissibility arrays.
TSOLVER	Coefficient matrix, C for gas component transport. (Not QA'd or Used)
UNITCNV	Unit conversion factors.
UPSTREAM	Arrays used for upstream weighting direction.
VEL	Inter-block fluxes and velocities.
VELAVE	Time averaged inter-block fluxes and velocities.
WELLS	Well parameters.

The flow of the common block data through BRAGFLO is represented in Table 2. All routines that include common block are listed in the first column. All common blocks are listed in the

# Information Only

first row. A check mark indicates that the common block is used in the corresponding routine. Any routine or common block not qualified or used in WIPP compliance calculations is indicated by the phrase, "N-QA". Looking down a column one can determine where a common block is used. Looking along a row one can determine which common blocks are used by a given routine. Because of its size the table has been split into two tables for the combination of two sets of subroutines and two sets of common blocks.

Table 2. BRAGFLO Common Block Usage

Routine Name	BAND	BASE (N-QA)	BF	BRATE (N-QA)	BRINETERMS	CCONVR	CLOSURE	CMULTG (N-QA)	CONC	CONPRE	CPRINT	CPUNTC (N-QA'd)	CRESTART(N-QA)	CUM	DIR_MULTIPLIERS	FLAWS	FRACTURE	FUN	GASINTERMS	GASTERMS	GEOTRANS	ICRESET	INITL	IO	JACOBIAN	LU	MACHINE	MASSBAL	MATTERMS	MODELTYPE	MONITOR	MXTDIM	NEWTON	NUCLIDE (N-QA)	
BLOCK DATA					✓	✓	✓				✓	✓	✓	✓	✓	✓				✓		✓	✓	✓	✓							✓	✓		
BRAGFLO											✓	✓	✓	✓	✓	✓	✓			✓		✓	✓	✓	✓	✓							✓		
ADJUSTICS				✓																	✓	✓	✓	✓	✓				✓						
ALI2VAX																																			
ALR2VAX																																			
BANDIT (N-QA)																																			
BANDSTORE (N-QA)	✓																																✓		
BI_INTERP																																			
BIOFES																																			
BIOHUM																																			
BIOMGO																																			
BIOSAT																																			
BUBBLE (N-QA)				✓																															
CLOSCHECK							✓																✓	✓					✓						
CONSOL							✓						✓								✓		✓	✓					✓						
CONSOL1							✓														✓		✓	✓					✓						
CORHUM																																			
CORMGO																																			
CORSAT																																			
CUMULGEN							✓		✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
DASUM (N-QA)																																			
DATNTIM																																			
DAXPY (N-QA)																																			
DDOT																																			
DECAY (N-QA)																																			✓
DECBR																										✓	✓								
DELTAMAP																										✓	✓								
DENGZ																						✓													
DENGZINT																						✓													
DENO					✓																														
DENSAT1																																			
DEPINVERT																																			
DEPINVERT1																																			
DGBCO																																			
DGBFA																																			
DGBSL																																			
DISKR (N-QA)						✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	

Routine Name	BAND	BASE (N-QA)	BF	BRATE (N-QA)	BRINETERMS	CCONVR	CLOSURE	CMULTG (N-QA)	CONC	CONPRE	CPRINT	CPUNTC (N-QA'd)	CRESTART(N-QA)	CUM	DIR_MULTIPLIERS	FLWS	FRACTURE	FUN	GASINTERMS	GASTERMS	GEOTRANS	ICRESET	INITL	IO	JACOBIAN	LU	MACHINE	MASSBAL	MATTERMS	MODELTYPE	MONITOR	MXTDIM	NEWTON	NUCLIDE (N-QA)			
DISKW (N-QA)						✓	✓	✓	✓		✓	✓	✓	✓								✓	✓														
DISOLVGAS (N-QA)					✓				✓												✓																
DSCAL																																					
FMAX (N-QA)																																					
FMIN																																					
FUNCT									✓							✓		✓														✓					
FUNCT7									✓							✓		✓																			
GASC																																					
GEN_COEF (N-QA)		✓														✓																					
GEOMETRY																						✓	✓														
GETJAC							✓	✓								✓						✓	✓											✓			
GETJACINDX								✓																		✓									✓		
GETOLD							✓		✓																												
GETWELLS									✓							✓																					
GETWELLS1									✓							✓																					
HYDROCONV																																					
IDAMAX (N-QA)																																					
INITAL																									✓												
KLINKBERG																																					
LABELS																									✓												
LOCATE																																					
MAPDEP																											✓										
MASSBALNC																												✓									
MATERIALS					✓		✓		✓														✓		✓	✓									✓	✓	
MEAS_CONV																								✓	✓	✓											
MULTIGRID																																					
PERMPHI																		✓																			
PERMS																		✓						✓													
PORSOLID																																					
POR_SURF							✓																		✓											✓	
PRINTASC				✓					✓		✓			✓		✓									✓												
PRINTBIN				✓					✓				✓	✓		✓								✓			✓	✓									
PRINTCONTROL													✓										✓													✓	
PRINTGRID																								✓													
PRINTHIV			✓						✓		✓		✓	✓		✓								✓			✓	✓	✓						✓	✓	
PRINTLABS																								✓			✓									✓	✓
PROPS					✓		✓		✓								✓	✓	✓	✓	✓	✓	✓	✓													
PROPS1					✓		✓		✓								✓	✓	✓	✓	✓	✓	✓	✓													
PRORDER																									✓												
PTHRESH																									✓												
QABGNL																											✓										
QAENDL																																					✓
RADIOLYSIS (N-QA)																																					✓
RADTRANS (N-QA)																							✓	✓	✓												
RAPHSON							✓	✓		✓	✓						✓							✓	✓	✓											
READ_DIRICHLET										✓														✓	✓	✓											✓
READCLOSURE							✓																	✓													
READCNTRL								✓									✓							✓	✓												
READFILES												✓												✓	✓		✓										
READFLUID					✓																			✓			✓										

Routine Name	BAND	BASE (N-QA)	BF	BRATE (N-QA)	BRINETERMS	CCONVR	CLOSURE	CMULTG (N-QA)	CONC	CONPRE	CPRINT	CPUNTC (N-QA'd)	CRESTART(N-QA)	CUM	DIR_MULTIPLIERS	FLAWS	FRACTURE	FUN	GASINTERMS	GASTERMS	GEOTRANS	ICRESET	INITL	IO	JACOBIAN	LU	MACHINE	MASSBAL	MATTERMS	MODELTYPE	MONITOR	MXTDIM	NEWTON	NUCLIDE (N-QA)					
READICS																																							
READMAT							✓										✓																						
READMESH																																							
READPRTYPE													✓																										
READRAD (N-QA)																																							
READRXGAST (N-QA)																																							
READRXN					✓																																		
READSTARTUP													✓																										
READWELL																																							
REFCONDS					✓				✓											✓	✓																		
RELPERM												✓																											
RESET					✓		✓		✓														✓	✓															
RESETMID					✓		☒		✓																														
RESIDUAL																																							
ROCKCOMP																																							
RPGM_INTRFC (N-QA)		✓	✓	✓																																			
RXGAST (N-QA)	✓																																						
SET DELTA DIRICHLET											✓																												
SET MAIN DIRICHLET								✓		✓						✓										✓													
SET RHS DIRICHLET										✓						✓										✓	✓												
SMOOTHPERM																																							
SOLBR																																							
SOLUTION							✓	✓			✓	✓													✓	✓	✓												
SOLVER					✓		✓	✓	✓			✓						✓							✓	✓	✓												
SOR																																							
STOREINT																																							
SUMMARY								✓			✓																												
TIMESTEP							✓																																
TRI INTERP																																							
TSAVG																																							
TTIME																																							
UNITSCONV																																							
WHERE (Not QA'd)																																							
WHICH (Not QA'd)																																							
WRITBIN																																							
WRITEARRAY																																							
WRITHIV																																							

Table 3. BRAGFLO Common Block Usage (continued)

Routine Name	OLD	PERMPARM	PHYSCON	PRNTC	PRNTI	PRNTR	PROGRAM	PROP	PT	RADACT (N-QA)	RADCHAIN (N-QA)	RATES (N-QA)	REF	REU	RELPERMS	RKSEOS	RPGM (N-QA)	RPGMCONC (N-QA)	RPGMGR (N-QA)	RPGMR (N-QA)	RSIZE	RSTBR	RXN	SATN	SIZES	SMPRM	SUBS (N-QA)	SWITCH	T	TERMS	TSOLVER(N-QA)	UNITCNV	UPSTREAM	VEL	VELAVE	WELLS				
BLOCK DATA			✓	✓	✓	✓																																		
BRAGFLO			✓	✓	✓	✓																																		
ADJUSTICS								✓																																

Information Only

Routine Name	OLD	PERMPARM	PHYSCON	PRNTC	PRNTI	PRNTR	PROGRAM	PROP	PT	RADACT (N-QA)	RADCHAIN (N-QA)	RATES (N-QA)	REF	REU	RELPERMS	RKSEOS	RPGM (N-QA)	RPGMCONC (N-QA)	RPGMGR (N-QA)	RPGMR (N-QA)	RSIZE	RSTBR	RXN	SATN	SIZES	SMPRM	SUBS (N-QA)	SWITCH	T	TERMS	TSOLVER(N-QA)	UNITCNV	UPSTREAM	VEL	VELAVE	WELLS		
ALI2VAX																																						
ALR2VAX																																						
BANDIT (N-QA)																																						
BANDSTORE (N-QA)																																						
BIOFES																																						
BIOHUM																																						
BIOMGO																																						
BIOSAT																																						
BL_INTERP																																						
CLOSCHECK																																						
CONSOL																																						
CONSOL1																																						
CORHUM																																						
CORMGO																																						
CORSAT																																						
CUMULGEN																																						
DASUM (N-QA)																																						
DATNTIM																																						
DAXPY (N-QA)																																						
DDOT																																						
DECAY (Not QA'd)																																						
DECBR																																						
DELTAMAP																																						
DENGZ																																						
DENGZINT																																						
DENO																																						
DENSAT1																																						
DEPINVERT																																						
DEPINVERT1																																						
DGBCO																																						
DGBFA																																						
DGBSL																																						
DISKR (N-QA)																																						
DISKW (N-QA)																																						
DISOLVGAS (Not QA'd)																																						
DSCAL (N-QA)																																						
FMAX (N-QA)																																						
FMIN																																						
FUNCT																																						
FUNCT7																																						
GASC																																						
GEN_COEF (N-QA)																																						
GEOMETRY																																						
GETJAC																																						
GETJACINDX																																						
GETOLD																																						
GETWELLS																																						
GETWELLS1																																						
HYDROCONV																																						
IDAMAX (N-QA)																																						
INITAL																																						
KLINKBERG																																						
LABELS																																						
LOCATE																																						
MAPDEP																																						
MASSBALNC																																						

Routine Name	OLD	PERMPARM	PHYSCON	PRINTC	PRINTI	PRINTR	PROGRAM	PROP	PT	RADACT (N-QA)	RADCHAIN (N-QA)	RATES (N-QA)	REF	REU	RELPERMS	RKSEOS	RPGM (N-QA)	RPGMCONC (N-QA)	RPGMGR (N-QA)	RPGMR (N-QA)	RFSIZE	RSTBR	RXN	SATN	SIZES	SMPRM	SUBS (N-QA)	SWITCH	T	TERMS	TSOLVER(N-QA)	UNITCNV	UPSTREAM	VEL	VELAVE	WELLS			
MATERIALS								✓	✓						✓														✓	✓									
MEAS CONV			✓																						✓	✓				✓									
MULTIGRID																																							
PERMPHI																																							
PERMS								✓	✓																														
PORSOLID														✓										☒	☒	✓													
POR SURF																																							
PRINTASC				✓	✓	✓		✓	✓									✓	✓	✓	✓	✓		✓	✓	✓				✓				✓	✓	✓			
PRINTBIN				✓	✓	✓		✓	✓									✓	✓	✓	✓	✓		✓	✓	✓				✓				✓	✓	✓			
PRINTCONTROL				✓	✓	✓																								✓									
PRINTGRID																																							
PRINTHIV				✓	✓	✓		✓	✓									✓	✓	✓	✓	✓		✓	✓	✓				✓					✓	✓			
PRINTLABS				✓	✓	✓																				✓													
PROPS		✓	✓					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓				✓	✓	✓			
PROPS1		✓	✓					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓				✓	✓	✓			
PROORDER																																							
PTHRESH															✓											✓													
QABGNL								✓																															
QAENDL								✓																															
RADIOLYSIS (Not QA'd)											✓																												
RADTRANS (Not QA'd)			✓					✓		✓														✓	✓					✓							✓		
RAPHSON			✓					✓							✓	✓			✓	✓	✓	✓		✓	✓				✓	✓	✓								
READ_DIRICHLET															✓										✓														
READCLOSURE																									✓													✓	
READCNTRL																	✓												✓	✓								✓	
READFILES																																							
READFLUID			✓					✓				✓												✓		✓													
READICS																									✓		✓												
READMAT		✓	✓					✓							✓									✓		✓	✓	✓											
READMESH			✓																							✓													
READPRTYPE				✓	✓	✓																																	
READRAD (Not QA'd)			✓								✓																												
READRXGAST (Not QA'd)																																							
READRXN		✓																						✓															
READSTARTUP																																							
READWELL																																							
READ_DIRICHLET																																							
REFCONDS								✓	✓				✓											✓	✓														
RELPERM		✓																																					
RESET								✓	✓															✓	✓														
RESETMID								✓	✓					✓	✓																								
RESIDUAL																									✓	✓													
ROCKCOMP			✓																																				
RPGM_INTRFC (Not QA'd)										✓	✓							✓	✓	✓	✓	✓						✓											
RXGAST (Not QA'd)								✓	✓									✓	✓	✓	✓	✓																	
SET_DELTA_DIRICHLET																																							
SET_MAIN_DIRICHLET																																							
SET_RHS_DIRICHLET																																							
SMOOTHPERM								✓																		✓	✓												
SOLBR																																							
SOLUTION																																							
SOLVER		✓		✓	✓	✓		✓	✓						✓								✓	✓	✓	✓				✓	✓								
SOR																																							
STOREINT																									☒	✓													
SUMMARY		✓						✓	✓																														
TIMESTEP		✓		✓	✓	✓		✓	✓																														

Routine Name	OLD	PERMPARM	PHYSCON	PRNTC	PRNTI	PRNTR	PROGRAM	PROP	PT	RADACT (N-QA)	RADCHAIN (N-QA)	RATES (N-QA)	REF	REU	RELPERMS	RKSEOS	RPGM (N-QA)	RPGMCONC (N-QA)	RPGMGR (N-QA)	RPGMR (N-QA)	RSIZE	RSTBR	RXN	SATN	SIZES	SMPRM	SUBS (N-QA)	SWITCH	T	TERMS	TSOLVER(N-QA)	UNTCNV	UPSTREAM	VEL	VELAVE	WELLS			
TRI_INTERP																																							
TSAVG																																							
TTIME																																							
UNITSCONV																																							
WHERE (Not QA'd)																																							
WHICH (Not QA'd)																																							
WRITBIN																																							
WRITEARRAY																																							
WRITHIV																																							

### 5.4.2 Subroutine Arguments

The second method of data flow through BRAGFLO is by passing the required data as arguments in the routines call list. For each routine that uses arguments, the BRAGFLO source code contains a comment section that describes all variables passed as arguments. Since this design document is an 'as built' the reader is referred to the source code for a complete description of the argument lists. Shown below is an example pseudo code fragment for the subroutine ADJUSTICS that uses arguments. The code fragment contains the subroutine name, the argument list and descriptions are provided in a comment section for the routine. The following subroutine declaration and argument list comments were extracted from the BRAGFLO source code and is representative of BRAGFLO routines using an argument list.

```

SUBROUTINE ADJUSTICS (ICWASTE, POWASTEIC, SOWASTEIC)
C
C   Input Arguments:
C   -----
C       ICWASTE   = Flag to indicate whether to reset input initial
C                   conditions in waste to uniform IC's POWASTEIC and
C                   SOWASTEIC, as when initializing waste as an empty
C                   cavity to simulate operational phase.
C       POWASTEIC = Array of initial waste brine pressure, used to
C                   simulate empty excavation during operational phase.
C       SOWASTEIC = Array of initial waste brine saturation, used to
C                   simulate empty excavation during operational phase.
C
C-----
    
```

## 6.0 DATA STRUCTURES

BRAGFLO data structures include: common blocks; a set of problem sizing parameters contained in common block(s); an input file for specifying the problem to be solved; an input file containing time dependent creep closure (porosity) data; and several output files containing diagnostics, results and restart information. Common blocks were previously described in Section 5.4.1 as part of the data flow.

### 6.1 Problem Sizing Parameters

BRAGFLO uses one INCLUDE file, BF2\_PARAMS.INC, containing a single PARAMETER statement that sets dimensions on many of the arrays used in BRAGFLO. (Where the term “PARAMETER statement” is used, it refers to the statement in BF2\_PARAMS.INC.) The *BRAGFLO User’s Manual* (Camphouse 2012c) in the section titled *Description of input files* describes in detail the problem sizing parameters.

### 6.2 Input Data File

BRAGFLO requires two input files: 1) an input control file, BF2\_UIF\$INPUT, example of the name format is BF2\_TEST\_R000.INP and 2) a closure look-up data table input file, BF2\_UIF\$INPUTCS, example of name format is BF2\_TEST.CSD.

The input control file is an ASCII text file that controls the problem setup, solution and output. It is generally created by PREBRAG, but since it is a text file, it can be built by hand. The *BRAGFLO User’s Manual* (Camphouse 2012c) describes in detail all input parameters.

### 6.3 Creep Closure Look-Up Table Data

Versions previous to Version 5.0 had the information contained on this creep closure file specified in a series of data statements in BRAGFLO routine “BLOCK\_DATA”. This input file is an ASCII text file that contains information, which describes the closure behavior of the host rock surrounding the repository. In the case of WIPP PA applications, this data is produced by the rock mechanics code SANTOS and consists repository porosity and pressure time histories for a number of imposed gas generation rates. The *BRAGFLO User’s Manual* (Camphouse 2012c) in the section titled *Description of input files* describes in detail all input parameters.

### 6.4 Output Data Files

There can be up to 4 output files created by BRAGFLO. They are as follows:

- The BRAGFLO binary output (.BIN). This file is the primary results file that is converted by POSTBRAG into CAMDAT Database (.CDB) format.

- The BRAGFLO ASCII output (.OUT). This file echoes input and provides user-readable output and QA information.
- The BRAGFLO summary (.SUM). This file contains a limited amount of summary information on results at each time step for monitoring progress of the solution over time.
- The BRAGFLO restart output (.ROT). Enough information is stored in this binary file to permit restarting BRAGFLO if execution is halted.

#### 6.4.1 Binary Output (.BIN) File

This file is the primary results file that is converted by POSTBRAG into CAMDAT Data Base (.CDB) format. This file contains output in a more compact form than is possible in an ASCII output file. Most results of interest printed typically every 20 or so time steps and a more limited amount of information printed every time step. The beginning of the file contains QA information, as in the ASCII file (Subsection 6.4.2). It also contains an output variable glossary. Enough information is stored to enable a postprocessor, POSTBRAG to read the file and place the results into a CAMDAT data base for additional post-processing and analysis by the downstream codes. The structure of this file is described in the *BRAGFLO User's Manual* (Camphouse 2012c).

#### 6.4.2 ASCII Output (.OUT) File

This file echoes input and provides user-readable output and QA information. It is a formatted ASCII text file. Use of this file is optional. The output, by topic, include in this file is as follows:

- QA information, including a time-and-date-of-run stamp, BRAGFLO version number, and revision date.
- Input file echo, but with more information and annotation than found (or allowed) in the input file. Some processed input is then printed out, including information on grid sizes and volumes of each grid block, since this is not necessarily read directly from the input file.
- Initial conditions are printed, with the amount of information specified by the user.
- Results specified by the user are printed periodically, at times or frequencies specified by the user.
- Results at the end of the run are also printed, the amount of information printed being specified by the user.
- Finally an ending time-and-date stamp is printed.

#### 6.4.3 Summary (.SUM) File

A limited amount of summary information is written to the screen at each time step when BRAGFLO is run interactively or to a log file when run in background. The same information is

written to this summary file so that a detailed time step summary is available later if BRAGFLO is run interactively. This information is often useful for diagnostic purposes when problems occur. The *BRAGFLO User's Manual* (Camphouse 2012c) in the section titled *Description of output files* describes in detail the summary file.

#### **6.4.4 Restart Output (.ROT) File**

NOTE: Restart capability is not qualified and will not be used in WIPP compliance calculations.

This binary file is written, if requested in the input, at specified times during a run. It is always written at the end of a run if the restart write flag has been set in the input. Enough information is saved to enable BRAGFLO to pick up from where it left off. Each batch of information written is referred to as a restart record. It is up to the user to keep track of when each restart record is written so that if a restart run is needed, the point where BRAGFLO is to resume can be indicated by the number of the restart record. Periodic writing of restart records is strongly recommended in calculations that are anticipated to be lengthy as a precaution against computer or power failures or unforeseen input errors that do not go into effect until late in the calculation. A message is written to the screen and to the summary file (if one is being written) when a restart record is written to assist the user in keeping track of when and how many restart records were written.

## 7.0 Capabilities and Limitations of the Software for the WIPP Application

BRAGFLO is the two-phase (brine and gas) finite difference program used to study the fluid flow within the WIPP repository site. The BRAGFLO code is an essential part of the probabilistic modeling necessary for performance assessment of the WIPP site. BRAGFLO Version 6.02 has more capabilities than are necessary for the WIPP PA application. The implementation of BRAGFLO Version 6.02 for the WIPP PA has the following capabilities and limitations:

1. The WIPP PA implementation of BRAGFLO assumes two fluid phases: a liquid phase (brine) and a gas phase ( $H_2$ ).
2. In the WIPP PA implementation of BRAGFLO fluid phases are assumed to be immiscible.
3. In the WIPP PA implementation of BRAGFLO the liquid phase consists of one component, brine. Dissolution of gas in the brine phase is not considered.
4. In the WIPP PA implementation of BRAGFLO the gas phase consists of one component,  $H_2$ . Water vapor in the gas phase is not considered. The primary component of the gas generation process is  $H_2$ .
5. The WIPP PA implementation of BRAGFLO has the capability of computing relative permeability and capillary pressure using several different empirical relations. Choice of the characteristic curves is controlled by the user.
6. The WIPP PA implementation of BRAGFLO allows for coupling to the mechanical system (rock matrix) through the introduction of a variable rock compressibility.
7. The fracture treatment in the WIPP PA implementation of BRAGFLO allows for pressure induced alterations to the porosity by introducing a pressure-dependent porosity.
8. In the WIPP PA implementation of BRAGFLO fracture treatment further allows for change in the fracture material permeability.
9. Representation of the creep closure within the WIPP PA implementation of BRAGFLO is accomplished by a change in the formation porosity. This porosity function or porosity surface will be time dependent and is also driven by pressure. The resulting closure information comes from a geomechanical simulation using the structural mechanics code SANTOS.
10. The WIPP PA implementation of BRAGFLO corrects gas permeability for the Klinkenberg effect.
11. Well models in the WIPP PA implementation of BRAGFLO allow simulation of gas or brine injection or production.
12. Within the WIPP PA implementation of BRAGFLO there are chemical reactions which consume brine and generate gas. The WIPP PA implementation of BRAGFLO accounts for these reactions by using the Average Stoichiometry Model, which,

assumes that all reaction rates are zero order (independent of reactant or product concentration). However, the WIPP PA implementation of BRAGFLO recognizes a different reaction rate if the solid reactant is in contact with liquid brine (inundated condition) or if the solid reacts with brine in the gas phase (humid condition) .

13. The WIPP PA implementation of BRAGFLO assumes that the directions of the permeability tensor are aligned with the reservoir coordinate axis; that is,  $\theta = 0$ .
14. The WIPP PA implementation of BRAGFLO evaluates brine density assuming its fluid compressibility is constant.
15. The WIPP PA implementation of BRAGFLO uses the Redlich-Kwong-Soave equation of state to relate gas density to pressure.
16. The WIPP PA implementation of BRAGFLO treats viscosity constant for both water and gas.
17. The WIPP PA implementation of BRAGFLO assumes a no-flow Neumann type boundary condition at all exterior grid boundaries.
18. The WIPP PA implementation of BRAGFLO uses a fully implicit formulation to solve the material balance equations.
19. The WIPP PA implementation of BRAGFLO uses a first order difference quotient approximation for the Jacobian evaluation.
20. In the WIPP PA implementation of BRAGFLO the user can specify from input directives the frequency of the Jacobian evaluation in the Newton-Raphson iteration.
21. For both storage and computational considerations the linear equation solver should take advantage of the banded structure. The WIPP PA implementation of BRAGFLO uses the banded LU solver.
22. To minimize the bandwidth, the WIPP PA implementation of BRAGFLO orders the grid blocks so that sequential indexing proceeds first in the shortest grid direction, then in the longest grid direction.
23. The WIPP PA implementation of BRAGFLO is written with double precision word length which produces floating point number representation of 64 bits or approximately 15 significant digits on a single word length machine such as the DEC Alpha. The only qualified solver is the LU-decomposition.
24. The WIPP PA implementation of BRAGFLO implements a partial pivoting (row interchanges) which reduces the roundoff errors in the arithmetic and is essential for the numerical stability of Crout's method.

## 8.0 REFERENCES

- Camphouse, C. 2012a. "Implementation Document for BRAGFLO, Version 6.02." Carlsbad, NM: Sandia National Laboratories. ERMS 558662.
- Camphouse, C. 2012b. "Requirements Document & Verification and Validation Plan for BRAGFLO, Version 6.02." Carlsbad, NM: Sandia National Laboratories. ERMS 558659.
- Camphouse, C. 2012c. "User's Manual for BRAGFLO, Version 6.02." Carlsbad, NM: Sandia National Laboratories. ERMS 558663.
- Gilkey, A. P. 2006a. "CAMCON\_LIB, Version 2.21." Carlsbad, NM: Sandia National Laboratories. ERMS 543028.
- Gilkey, A. P. 2006b. "CAMDAT\_LIB, Version 1.26." Carlsbad, NM: Sandia National Laboratories. ERMS 543020.
- Gilkey, A. P. 2006c. "CAMSUPES\_LIB, Version 2.23." Carlsbad, NM: Sandia National Laboratories. ERMS 543022.
- Gilkey, A. P. and D. K. Rudeen. 2007. "User's Manual for PREBRAG version 8.00." Carlsbad, NM: Sandia National Laboratories. ERMS 545274.
- Nemer, M. B. 2006. "Requirements Document & Verification and Validation Plan for BRAGFLO Version 6.0." Carlsbad, NM: Sandia National Laboratories. ERMS 545014.
- Nemer, M. B. 2007. "POSTBRAG, Version 4.00A." Carlsbad, NM: Sandia National Laboratories. ERMS 545667