DOE/ID/12842-2
Distribution Category UC-122

BOAST II for the IBM 3090 and RISC 6000

Final Report

By
Paul Hebert
Adam T. Bourgoyne, Jr.
John Tyler

May 1993

Work Performed Under Grant No. DE-FG07-89ID12842

Prepared for
U. S. Department of Energy
Assistant Secretary for Fossil Energy

Gene Pauling, Project Manager
Metairie Site Office
900 Commerce Road East
New Orleans, LA 70123

Chandra Nautiyal, Project Manager
Bartlesville Project Office
P. O. Box 1398
Bartlesville, OK 74005

Prepared by
Louisiana State University
Petroleum Engineering Department
Computer Science Department
Baton Rouge, LA 70803-6417

Table of Contents

Section VI Appendices

Introduction I

LSU was supported by DOE under Grant DE–FG07–89ID12842 to assist the search for by–passed oil in old domestic reservoirs by small independent oil companies that do not have technical support staff. As part of this effort, LSU worked on improving reservoir simulation software that is available in the public domain. BOAST II (A Three–Dimensional, Three–Phase Black Oil Applied Simulation Tool) was obtained from DOE as the starting point for this effort. It was modified for efficient use on IBM mainframe computers. A vectorized code was prepared that will run on the IBM Risc 6000 workstation as well as other large scale platforms. A vectorized–parallel code was constructed to run efficiently on large scale IBM platforms. Preprocessing and post–processing programs were written to assist in data preparation and output analysis. This manual is a modification of the previously released manual by Fanchi, Kennedy, and Dauben[1] (1987). It was written to provide documentation for the revised software available as a result of the work done al LSU.

The Introduction contains two parts: an overview of the simulator as an engineering tool and an introduction to running BOAST and using the manual.

1 Fanchi, J.R., Kennedy, J.E., and Dauben, D.L., "Boast II: A Three–Dimensional, Three Phase Black Oil Applied Simulation Tool," DOE/BC –88/2/SP

Overview

BOAST II simulates isothermal, darcy flow in three dimensions. It assumes that reservoir liquids can be described in three fluid phases (oil, gas, and water) of constant composition, with physical properties that depend on pressure only.

These reservoir fluid approximations are acceptable for a large percentage of the world's oil and gas reservoirs. Consequently, BOAST II has a wide range of applicability. BOAST II can simulate oil and/or gas recovery by fluid expansion, displacement, gravity drainage, and capillary imbibition mechanisms. Typical field production problems that BOAST II can handle include primary depletion studies, pressure maintenance by water and/or gas injection, and evaluation of secondary recovery waterflooding and displacement operations.

Technically, BOAST II is a finite–difference, implicit pressure, explicit saturation (IMPES) numerical simulator. It applies both direct and iterative solution techniques for solving systems of algebraic equations. The well model allows specification of rate or pressure constraints on well performance, and the user is free to add or to recomplete wells during the simulation. In addition, the user can define multiple rock and PVT regions and can choose from three aquifer models.

BOAST II also provides flexible initialization, a bubble–point tracking scheme, automatic time–step control, and a material balance check on solution stability. The user controls output, which includes a run summary and line–printer plots of fieldwide performance.

In 1982 the Department of Energy released the original version of the simulator, BOAST: A Three–Dimensional, Three–Phase Black Oil Applied Simulation Tool (Version 1.1). Various research, academic, and industrial organizations used program extensively. It was popular because it was easy to use and because it was not burdened by seldom–used features that complicate execution.

However, BOAST was not originally designed for field–scale problems. The new version, BOAST II, was developed to overcome some of the limitations of the original program. The developers aimed to keep the program uncomplicated and easy to use while adding features that would improve the simulator's versatility. The example runs in Chapter 19 illustrate the flexibility of BOAST II. The following are the major enhancements that BOAST II provides.

Reservoir Engineering Options:
Three–phase relative permeability algorithm
Multiple rock regions
Multiple PVT regions
Bubble point pressure varying with depth and PVT region
Three separate analytic aquifer models
Direct input of noncontiguous layers
Net and gross thicknesses

Well Model Features:
Individual well gas/oil ratio (GOR) and water/oil ratio (WOR) constraints
Minimum oil production and maximum liquid withdrawal well constraints

Multiple wells per grid block
Maximum water/gas injection rates

Numerical Options:
Five iterative matrix solution methods
Zero pore volume (inactive) grid blocks
Two–point upstream weighting for reducing numerical dispersion

BOAST II has been tested under a wide range of conditions. The tests have generally verified the validity of BOAST II and confirmed that BOAST II is superior to BOAST. The evaluations have also compared results from BOAST II with those from other accepted simulators. The comparisons were made for four types of problems: oil and gas depletion, waterflooding, gas injection with constant bubble point pressure, and gas injection with variable bubble point pressure.

The results compared favorably with respect to oil rates, GORs, gas saturations, and pressures. The one exception is the reservoir pressure comparison for the variable bubble point pressure problem. Despite agreement on the production rates, GORs, and gas saturation profiles, the reservoir pressures calculated by BOAST II were consistently lower than those calculated by other simulators.

A mass–conserving expansion of accumulation terms would improve the accuracy for variable bubble point problems, as discussed by Fanchi[2].

[2] Fanchi, J. R. 1987. BOAST–DRC: black oil and condensate reservoir simulation on an IBM–PC. SPE Paper 15297.

Getting Started Chapter 1

Using the Manual

Running BOAST essentially means providing the simulator with a data file and executing the program using an appropriate Job Control Language (JCL). Special preprocessor programs that can automate preparation of the data are presented in Appendix M. Sample JCL files are given in Appendix L. The manual devotes a chapter for each topic in the data. For each feature, the manual provides an introduction, then takes the user line–by–line through the input requirements. Below is an excerpt from a data input chapter.

> **Line 2** **Size of the model grid**
>
> The variables II, JJ, and KK define the number of grid blocks in the x–, y–, and z–directions, respectively.

The indication **Line 2** in the margin prompts the user to start a new line for this entry. Line numbers start from 1 in each chapter. The boldface identifies the feature; the text explains how the variables operate.

This instruction describes three variables. Variables are always indicated in capital letters (II, JJ, and KK). The prompt implies that all three variables must be entered on the same line.

Only when a Line entry calls for an array can it correspond to more than one physical line of data. Entering Arrays, below, continues this discussion.

To refer to a specific Line within a chapter, the manual names the chapter then the Line, separated by a hyphen. The indication 5–7 refers to Chapter 5, Line 7. The form 4.1–2 refers to Chapter 4, Section 1, Line 2.

BOAST Input

The input for a BOAST simulation is a text file specifying physical parameters for the study area and methods for executing the simulation. The input is divided broadly into an **Initialization Data** section and a **Recurrent Data** section. Initialization Data defines the mathematical model, the run–control parameters, and the physical attributes of study area. Recurrent Data provides time–step and output control, along with well information.

A line of input contains either raw data (numbers) or a title. A title is a line of text that serves to make the data file more readable, delineating the various sections for the user.

NOTE: The program will fail if even one variable is missing from any line of the input or if the order is not exactly correct. Read the following section carefully.

Notes

Read all comments that begin with NOTE:, as above. The Notes are not extra information; they help the prevent the user from making mistakes. They also answer common questions and clarify the more difficult sections.

Entering Data

Although titles lines serve only to make the data more readable, they are necessary. No title can be skipped because BOAST "expects" this line before it reads the actual data on the following lines. (The text of a title is arbitrary, but the length must be no more than 80 characters.)

Note that BOAST provides no default values; a value must be entered for every parameter. However, the manual does attempt to provide "typical values" for most parameters.

Data Format

Data is entered in a free–field format. In other words, the numbers must be entered in a specific order, but in no particular position on the line. Also, multiple entries on one line must be separated by at least one space or a comma. If only one entry is required, it may appear anywhere on the line. The manual will specify whenever a particular position is required.

For convenience, BOAST will interpret the pattern *15*1* as 15 entries, each equal to 1; BOAST would read *3*0* as 0 0 0. This notation may be especially useful for entering arrays, discussed below.

Entering Arrays

The order for entering a three–dimensional array (II × JJ × KK) follows:

The K = 1 layer is entered first. Within this layer, the data is entered by rows, starting with row J = 1. Values for this row are read in from columns 1 to II. Next, another II values are read for row J = 2, and so on until JJ rows have been entered. One layer is complete, and the process is repeated for layers 2 to KK.

Remember to start a new line for each grid row of data (JJ rows of II values).

Assume, for example, that a grid measures $10_i \times 5_j \times 3_k$. To enter values corresponding to each block, begin with the first layer (of three). Enter the first row, which consists of 10 column values. Therefore, enter ten values on the line. Go to a new line. For the next four rows, continue entering 10 values per line. Layer K = 1 is entered. Repeat the process for the second and third layers. If the number of columns exceeds the number that can be conveniently placed on a line, multiple lines can be used for each row of data. Many users limit the number of entries per line to ten to make checking the number of entries easier.

Modifications to . . .

Sections titled "Modifications to . . ." help the user to define a parameter when the parameter is uniformly distributed throughout the grid except for a specific region

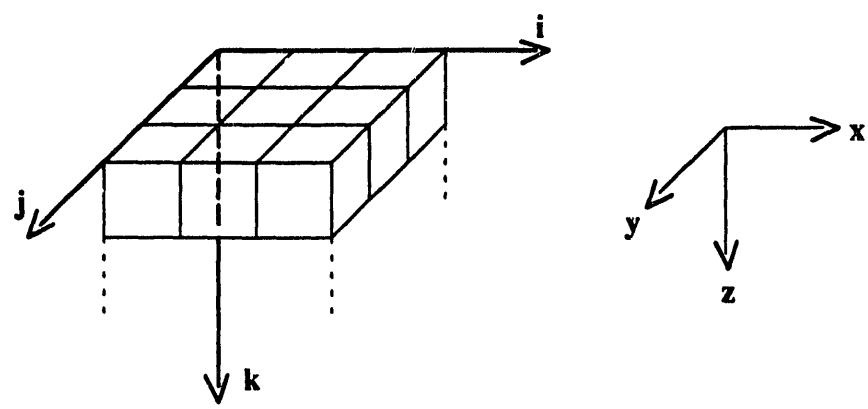
or a small number of regions. It is often convenient for the user to define the parameter as though it were uniformly distributed, then to **modify** some small portion(s) of the grid.

For example, if the user wanted 25% porosity throughout the grid, but 30% in a specific region, it would be simplest first to read one value (0.25) common to all grid blocks, then to redefine the exceptional regions as 0.30.

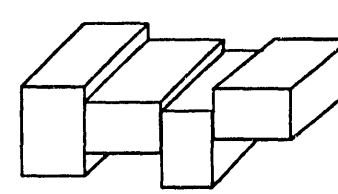
Modifications also allow the user to manipulate and experiment with the data. If the changes are incorrect, the user has not lost the original data.

Grid Geometry

BOAST uses a right–handed coordinate reference. The top layer (K = 1) is shown.



The above grid shows the basic structure of the model. However, because the depth and thickness of each block is individually sized, the following depiction of part of a row is more realistic.



Modifying Grid Regions

The user can modify parameters in three–dimensional regions within the model grid. Six variables describe a region. I_1 and I_2 are the first and last blocks in the x–direction, J_1 and J_2 in the y–direction, and K_1 and K_2 in the z–direction.

Throughout the region, BOAST replaces the old value (for length, porosity, permeability, or transmissibility) with the new one. The new parameter value will follow I_1, I_2, J_1, J_2, K_1, and K_2 on the data line.

Running BOAST

To run a BOAST simulation, the user must create an input file and initiate the program to run on that data. However, this manual addresses three versions of BOAST: a vectorized version for the MVS operating system, a vectorized–parallel version for VM, and a vectorized version modified for AIX (for the IBM RISC 6000). The input requirements are identical for the three versions, and they share the free–field input format.

The MVS version, which uses JCL, is by far most commonly used. To show how to initiate a simulation for a given input file, the manual provides an example of a JCL file (for MVS) and a VM EXEC file, along with a sample command line for the IBM RISC SYSTEM 6000. These samples are in Appendix L.

Initialization Data II

Initialization data is read all at once at the beginning of the simulation. The Initialization Data section defines the mathematical model, the run–control parameters, and the physical attributes of study area.

The Initialization Data Section contains Chapters 2–12.

2. Restart and Run Summary
3. Grid Definition
4. Porosity and Permeability
5. Rock and PVT Regions
6. Relative Permeability and Capillary Pressure Tables
7. Fluid Data Tables
8. Pressure and Saturization
9. Debug and Diagnostics
10. Run Control
11. Solution Method
12. Aquifer Model

During a restart run, most of the Initialization Data section will be skipped because the same data is automatically stored in a file (RESIN) that can be loaded for a restart run. Recurrent data, however, will still be entered because BOAST requires this information during the simulation.

Enter the following line, the beginning of the BOAST input:

Line 1 Title

This line is to mark the beginning of the intialization data. Choose any phrase (limited to 80 characters) for the title, e.g., the name of the reservoir being simulated.

Restart allows the user to continue a previous simulation after a given time to model performance without recalculating the history up to that time. BOAST saves the conditions of the simulation in a **restart record.** The user may establish up to five checkpoints during a simulation. Each time that the simulation reaches a checkpoint, it updates the restart record. Continuing (restarting) from the restart record, the simulation may proceed under the control of a new set of parameters.

Run summary provides a summary table and 15 user–specified plots for the simulation. The table shows rates and cumulatives for fluid production and injection, along with reservoir pressure and fluid ratios.

Line 1 Title

The title serves as a delineator in the data. Enter up to 80 characters for the title, e.g., *Restart Data.*

Line 2 Control codes

The first line of data is two values, IREOPT and IPOSTP, which control restart and run summary in the following manner:

IREOPT	Instruction to BOAST
–1	Initialize and run; do not write the restart record
0	Initialize and run; choose up to 5 checkpoints
1	Restart with previous initialization; choose up to 5 checkpoints in the run

IPOSTP	Instruction to BOAST
–1	Do not produce a table or plots
0	Produce the summary table
1	Produce the summary table and the user–specified plots

NOTE: If IREOPT = –1, proceed to Line 7, omitting Lines 3, 4, 5, and 6. In this case, no restart records will be written.

Line 3 **Restart parameters**

The restart parameters line consists of four values, which have the following meanings:

- IRNUM Number of restart records (up to five) to be written.
- IRSTRT A time–step number identifying the restart record from which to continue if this is a restart run. When IREOPT = 0 (an initialization run), set this variable = 0. Time–step numbers were output when restart records were written (from Line 5 or 6 in a previous run).
- NN Maximum number of time steps allowed during a restart run. When IREOPT = 0, set this variable = 0 .
- TMAX Maximum number of days to be simulated during a restart run. When IREOPT = 0, TMAX entered on Line 2 of Run Control Parameters, Chapter 10, supercedes this value.

The simulation will terminate after NN time steps or after TMAX days, whichever comes first.

Line 4 **Input and output data files**

Enter RESIN and RESOUT, the input and output files containing restart records. The names must begin with a letter and may not include a space. The last character of RESIN must lie in column 10; RESOUT must end in column 20. Restart records are written to RESOUT when IREOPT = 0 or 1. Restart records are read from RESIN when IREOPT = 1 (a restart run).

The file RESIN must exist before running a restart (IREOPT = 1). The output file for one run will be the input file for a subsequent restart run.

NOTE: The user may choose to write restart records with respect to either time steps (Line 5) or elapsed calendar time (Line 6). Either way, Lines 5 and 6 will each contain IRNUM values (see Line 3). One Line will be all zeros (IRNUM of them) and the other Line will contain IRNUM values. Thus, the choice is made.

Line 5 **Write restart records at specified time steps**

To update the restart record at given time steps, enter non–zero IRNUM time–step values (see Line 3) on this line. Each of these entries, up to five in the sequence, must be larger than the preceding one. If the record is to be written after specified elapsed times (days), enter IRNUM zero values. (These values are entered into the array IRETYM.)

Line 6 **Write restart records at specified elapsed times**

To update the record after elapsed times, specify non–zero IRNUM elapsed times (days) after which to write records. Each value must be greater than its predecessor.

If the restart record is to be updated with respect to time steps, enter IRNUM zero values on the line. (The array holding these values is called REDATE.)

NOTE: If IREOPT = –1 (do not write the restart record), the user should have reached this point without entering Lines 3, 4, 5, and 6.

Line 7 Plot codes

Include this line only if IPOSTP = 1.

The first value to enter is NPLINE, the number of plot lines to represent one time step along the time axis. A typical value for NPLINE is 1 or 2. Set NPLINE = 0 for BOAST to automatically calculate the plot lines per time step (see Chapter 17).

Increasing NPLINE will result in a graph lengthened in the direction of the time axis. Time is marked every ten plot lines along the axis. If ATSC is activated, the time–step size is variable during the simulation. In this case, the largest time step occurring during the simulation is used in determining the plot size.

For example, if the time step is 12 days, and NPLINE = 2, two plot lines represent one time step; one plot line covers six days. Thus, after ten plot lines, a time value (60 days) is displayed. The plot shows twenty lines before the 120–day mark.

For each of the following terms, enter a 1 or 0 (*plot* or *do not plot*). Recall the compact notation for 15 consecutive ones, 15*1, or 15 consecutive zeroes, 15*0. This is convenient for choosing all or none of the plots. Enter all 15 values on the same line with NPLINE:

PLOT CODES

Term	Variable
KOPR	Oil production rate
KGPR	Gas production rate
KWPR	Water production rate
KGOR	Producing gas/oil ratio
KWOR	Producing water/oil ratio
KGIR	Gas injection rate
KWIR	Water injection rate
KRESP	Pore volume weighted average reservoir pressure
KAIR	Aquifer influx rate
KAIC	Cumulative aquifer influx
KCOP	Cumulative oil production
KCGP	Cumulative gas production
KCWP	Cumulative water production
KCGI	Cumulative gas injection
KCWI	Cumulative water injection

0 —Do not plot
1 —Plot this variable

BOAST will not produce a plot if the variable did not change during the run.

NOTE: If IREOPT = 1, skip the rest of Initialization Data, and go to Recurrent Data, Section III.

Grid Dimensions (1)

In this section the user defines the three–dimensional grid that represents the study area. Note that when the grid blocks are smaller, the simulation usually will be more accurate, but it will also run slower.

Line 1 **Title**

Line 2 **Size of the model grid**

The variables II, JJ, and KK define the number of grid blocks in the x–, y–, and z–directions, respectively.

Line 3 **Title**

Line 4 **Control codes for grid input**

Four variables control the method of grid input, KDX, KDY, KDZ, and KDZNET. Using the following tables, enter the four codes:

KDX	Controls the input of x–direction grid dimensions
–1	The x–direction grid dimensions are the same for all blocks in the grid. Read only one value.
0	The grid dimensions in the x–direction are read for each block in one row. These x–direction dimensions are assigned to every row in the model grid. Read II values.
1	The x–direction dimensions are read for every grid block in one layer. Every layer in the model grid is assigned these x–direction dimensions. Read II × JJ values.

KDY	Controls the input of y–direction grid dimensions
–1	The y–direction grid dimensions are the same for all blocks in the grid. Read only one value.
0	The grid dimensions in the y–direction are read for each block in one column. These y–direction dimensions are assigned to every column in the model grid. Read JJ values.
1	The y–direction dimensions are read for every block in one layer. Every layer is the model grid is assigned these y–direction dimensions. Read II × JJ values.
KDZ	**Controls the input of z–direction gross grid–block thicknesses**
–1	The z–direction gross thicknesses are the same for all blocks in the grid. Read only one value.
0	A constant value of gross thickness is read for each layer in the grid. Each layer has a uniform thickness. Read KK values.
1	The z–direction gross thicknesses are read for every block in the model. Read II × JJ × KK values.
KDZNET	**Controls the input of z–direction net grid–block thicknesses**
–1	The z–direction net thicknesses are the same for all blocks in the grid. Read only one value.
0	A constant value of net thickness is read for each layer in the grid. Each layer has a uniform thickness. Read KK values.
1	The z–direction net thicknesses are read for every block in the model. Read II × JJ × KK values.

NOTE: The format for entering arrays is discussed under Entering Data in the Getting Started chapter.

Line 5 **X–direction grid dimensions (DX)**

If KDX = –1, enter the one value here.
If KDX = 0, enter a value for each column (II values).
If KDX = 1, enter a value for each block in layer one (II × JJ values).

Line 6 **Y–direction grid dimensions (DY)**

If KDY = –1, enter the one value here.
If KDY = 0, enter a value for each row (JJ values).
If KDY = 1, enter a value for each block in layer one (II × JJ values).

Line 7 **Z–direction gross grid–block thicknesses (DZ)**

If KDZ = –1, enter the one value here.
If KDZ = 0, enter a value for each layer (KK values).
If KDZ = 1, enter a value for each block in the grid (II × JJ × KK values).

Line 8 **Z–direction net grid–block thicknesses (DZNET)**

If KDZNET = –1, enter the one value here.
If KDZNET = 0, enter a value for each layer (KK values).
If KDZNET = 1, enter a value for each block in the grid (II × JJ × KK values).

Modifications to Grid Dimensions (2)

The user can resize the blocks in specified regions of the grid. The user specifies regions for each direction (x, y, or z) in which to change block lengths. Within each region, all the blocks will assume a new, common dimension (DX, DY, DZ, or DZNET).

Line 1 **Title**

Line 2 **Number of regions to be resized for each direction; print code**

Five values are entered on this line:

- NUMDX Number of regions in which to change DX.
- NUMDY Number of regions in which to change DY.
- NUMDZ Number of regions in which to change DZ.
- NUMDZN Number of regions in which to change DZNET.
- IDCODE Print code:
  0 – do not print any modified grid dimensions;
  1 – print the new grid dimensions.

NOTE: Modifying grid regions is discussed under Entering Data in the Getting Started chapter.

Any input in Lines 3–6 will begin with I_1, I_2, J_1, J_2, K_1, and K_2. These variables are interpreted as follows:

> $I_1 - I_2$ is the range in the x–direction of the region to modify
> $J_1 - J_2$ is in the y–direction
> $K_1 - K_2$ is in the z–direction

Line 3 **Modifications to DX (x–direction thickness)**

Omit this Line if NUMDX = 0.

Otherwise, enter NUMDX lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 DX,

where DX = the new x–direction thickness for all blocks in the region.

Line 4 **Modifications to DY (y–direction thickness)**

Omit this Line if NUMDY = 0.

Otherwise, enter NUMDY lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 DY,

where DY = the new y–direction thickness for all blocks in the region.

Line 5 **Modifications to DZ (z–direction thickness)**

Omit this Line if NUMDZ = 0.

Otherwise, enter NUMDZ lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 DZ,

where DZ = the new z–direction gross thickness for all blocks in the region.

Line 6 **Modifications to DZNET (z–direction net thickness)**

Omit this Line if NUMDZN = 0.

Otherwise, enter NUMDZN lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 DZNET,

where DZNET = the new z–direction net thickness for all blocks in the region.

Depths to Top of Grid Blocks (3)

Input the depths from datum to the top of the grid blocks in the top layer of the model. In BOAST, z–direction coordinates increase going down. Thus, depths are read as depths *below* user–selected datum, for example, the depth below sea level.

Line 1 Title

Line 2 Input code

Enter one variable, KEL:

KEL	Controls the input of depth to top of layer one of grid
0	Read one constant depth to the top of all blocks in layer one.
1	Read the depth to the top of each block in layer one (II × JJ values).
2	Read one constant value for each layer (KK values). This choice assumes horizontal layers.
3	Read a depth value for every grid block (II × JJ × KK values).

NOTE: The format for entering arrays is discussed under Entering Data in the Getting Started chapter.

Line 3 Depth value(s)

Enter ELEV, the depth in feet to the top of the grid blocks as follows:

If KEL = 0, enter the one value on this line.
If KEL = 1, enter the II × JJ values.
If KEL = 2, enter the KK values.
If KEL = 3, enter the II × JJ × KK values.

The simulator automatically calculates the depths to blocks in layers other than the top layer. BOAST determines the depth of a block in layer K by totaling the z–direction gross thicknesses of blocks in layers 1 through K–1, and adding this to the depth of the top block.

The node of a block is in the middle of the block. Where Top(I,J,K) is the depth to the top of the block and DZ(I,J,K) is the height of the block, the z–direction position of the node is given by:

$$EL(I,J,K) = Top(I,J,K) + (0.5 \times DZ(I,J,K)).$$

Here, the user specifies porosity and permeability throughout the study area. Porosity is the percentage of void space within a reservoir rock. Permeability measures the ability of a rock to transmit fluid through pore spaces.

Distributions (1)

Line 1 Title

Line 2 Control codes for input

The codes control the input of the corresponding parameters:

- KPH Porosity.
- KKX X–direction permeability.
- KKY Y–direction permeability.
- KKZ Z–direction permeability.

The codes operate as follows:

Code	Meaning
–1	The value is the same for all blocks in the grid. Read only one value.
0	A constant value is read for each layer in the grid. Each layer has a uniform value. Read KK values.
1	A value is read for every block in the model. Read II × JJ × KK values.

Line 3 Porosity value(s)

Porosity is read as a decimal, not as a percentage (e.g., 21% should be entered as 0.21 or .21).

If KPH = –1, enter the value here.
If KPH = 0, enter a value for each layer (KK values).
If KPH = 1, enter a value for each block in the grid (II × JJ × KK values).

Line 4 Permeability value(s) in the x–direction

Permeability (KX) is in millidarcies.

If KPX = –1, enter the value here.
If KPX = 0, enter a value for each layer (KK values).
If KPX = 1, enter a value for each block in the grid (II × JJ × KK values).

Line 5 **Permeability value(s) in the y–direction**

Permeability (KY) is in millidarcies.

If KPY = –1, enter the value here.
If KPY = 0, enter a value for each layer (KK values).
If KPY = 1, enter a value for each block in the grid (II × JJ × KK values).

Line 6 **Permeability value(s) in the z–direction**

Permeability (KZ) is in millidarcies.

If KPZ = –1, enter the value here.
If KPZ = 0, enter a value for each layer (KK values).
If KPZ = 1, enter a value for each block in the grid (II × JJ × KK values).

Modifications to Porosity and Permeability (2)

The user can redefine parameters for regions within the grid. Refer to the section on modifying grid regions under Entering Data in the Getting Started chapter.

Porosity is modified over regions of the grid; permeability is modified in each direction (x, y, and z) throughout grid regions.

Line 1 **Title**

Line 2 **Number of regions in which to change porosity/permeability; print code**

Enter five variables on this line:

- NUMP Number of regions in which to change porosity.
- NUMKX Number of regions in which to change x–direction permeability.
- NUMKY Number of regions in which to change y–direction permeability.
- NUMKZ Number of regions in which to change z–direction permeability.
- IPCODE Print code:
  0 – do not print modified distribution
  1 – print modified distributions.

NOTE: Any input in Lines 3–6 will begin with I_1, I_2, J_1, J_2, K_1, and K_2. These variables are interpreted as follows:

> $I_1 - I_2$ is the range in the x–direction of the region to modify;
> $J_1 - J_2$ is in the y–direction;
> $K_1 - K_2$ is in the z–direction.

Remember that porosity is read in decimal form; permeability is in millidarcies.

Line 3 **Modifications to porosity**

Omit this Line if NUMP = 0.

Otherwise, enter NUMP lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 PH1,

where PH1 = the new porosity value for all blocks in the region.

Line 4 **Modifications to x–direction permeability**

Omit this Line if NUMKX = 0.

Otherwise, enter NUMKX lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 KX,

where KX = the new x–direction permeability for all blocks in the region.

Line 5 **Modifications to y–direction permeability**

Omit this Line if NUMKY = 0.

Otherwise, enter NUMKY lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 KY,

where KY = the new y–direction permeability for all blocks in the region.

Line 6 **Modifications to z–direction permeability**

Omit this Line if NUMKZ = 0.

Otherwise, enter NUMKZ lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 KZ,

where KZ = the new z–direction permeability for all blocks in the region.

No–Flow Boundaries through Transmissibility Modifications (3)

Transmissibility describes the flow resistance between two grid blocks. The flow rate is equal to the transmissibility times the pressure (or flow potential) drop between the grid blocks. By setting the transmissibility in a given direction between grid blocks to 0, the user can define no flow boundaries.

NOTE: Keep in mind the directional convention for specifying transmissibility modification:

> TX(I, J, K) controls flow across the boundary between blocks I–1 and I.
>
> TY(I, J, K) controls flow across the boundary between blocks J–1 and J.
>
> TZ(I, J, K) controls flow across the boundary between blocks K–1 and K.

Line 1 **Title**

Line 2 **Number of grid blocks in which to change transmissibility; print code**

Enter four variables:

- NUMTX Number of grid blocks for which to change the x–direction transmissibility.
- NUMTY Number of grid blocks for which to change the x–direction transmissibility.
- NUMTZ Number of grid blocks for which to change the x–direction transmissibility.
- ITCODE Print code:
  0 – do not print any modified transmissibility values;
  1 – print the new transmissibility values

NOTE: Any input in Lines 3–6 will begin with I_1, I_2, J_1, J_2, K_1, and K_2. These variables are interpreted as follows:

> $I_1 - I_2$ is the range in the x–direction of the region to modify;
> $J_1 - J_2$ is in the y–direction;
> $K_1 - K_2$ is in the z–direction.

Line 3 **Modifications to x–direction transmissibility**

Omit this Line if NUMTX = 0.

Otherwise, enter NUMTX lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 TX,

where KX = the new x–direction transmissibility for all blocks in the region.

Line 4 **Modifications to y–direction transmissibility**

Omit this Line if NUMTY = 0.

Otherwise, enter NUMTY lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 TY,

where KY = the new y–direction transmissibility for all blocks in the region.

Line 5 **Modifications to z–direction transmissibility**

Omit this Line if NUMTZ = 0.

Otherwise, enter NUMTZ lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 TZ,

where KZ = the new z–direction transmissibility for all blocks in the region.

Rock and PVT Types

Chapter 5

Rock and PVT types relate to certain geologic properties. A rock type distinguishes a rock with certain properties—for example, permeability. A certain PVT type describes specific fluid properties and the relationship among pressure, volume, and temperature. Every grid block corresponds to one of up to five rock and PVT types.

This chapter associates grid blocks with rock and PVT types. The next two chapters define the actual parameters of these distinct types.

Line 1 Title

Line 2 Number of rock, PVT types

Enter two values, NROCK and NPVT. NROCK is the number of distinct rock types defined for the simulation; NPVT is the number of distinct PVT types. These values must be between one and five but do not have to be equal.

A separate set of saturation–dependent data will describe each rock type, and a separate set of pressure–dependent data will describe each PVT type. This data will be entered in tables in the next two chapters.

NOTE: For convenience, all grid blocks will be assumed initially to be rock type 1 and PVT type 1. Areas of the model that are of a type other than this default are specified in the remainder of this chapter.

Line 3 Title

Omit this Line if NROCK = 1.

Line 4 Number of grid regions with non–default rock type

Omit this Line if NROCK = 1.

Enter NUMROK, the number of grid regions over which to assign a rock type value other than 1, the default.

An entry of 0 indicates that the user will input a rock type for every grid block (II × JJ × KK). This implies that the distribution of the types of rock throughout the model is fairly complex.

Line 5 Rock type specifications

Omit this Line if NROCK = 1.

If NUMROK = 0, enter one rock type value for each block in the grid, II × JJ × KK values. Entries range from 1 to NROCK.

If NUMROK > 0, enter NUMROK of the following lines:

I_1 I_2 J_1 J_2 K_1 K_2 IVAL,

where IVAL is the index number of the rock type assigned to this grid region.

> $I_1 - I_2$ is the range in the x–direction of the region to modify
> $J_1 - J_2$ is in the y–direction
> $K_1 - K_2$ is in the z–direction

Line 6 **Title**

Omit this Line if NPVT = 1.

Line 7 **Number of grid regions with new PVT type**

Omit this Line if NPVT = 1.

Enter NUMPVT, the number of grid regions over which to assign a PVT type value other than 1, the default.

An entry of 0 indicates that the user will input a PVT type for every grid block (II × JJ × KK). This implies that the distribution of the types of PVT throughout the model is complex.

Line 8 **PVT type specifications**

Omit this Line if NPVT = 1.

If NUMPVT = 0, enter a PVT type value for each block in the grid, II × JJ × KK values. Entries range from 1 to NPVT.

If NUMPVT > 0, enter NUMPVT of the following lines:

I_1 I_2 J_1 J_2 K_1 K_2 IVAL,

where IVAL is the index number of the PVT type assigned to this grid region.

In this chapter, enter a separate set of saturation–dependent data for each rock type. This data is entered in tables containing relative permeability and capillary pressure. One table is entered for each rock type.

Line 1 Title

It is convenient to use this line as a header for the table of entries to follow. Type the variable names in the order in which the variables will be entered.

Line 2 Relative permeability and capillary pressure tables

For each rock type (there are NROCK, Chapter 5, Line 2), input a table as below:

SAT_1	$KROW_1$	KRW_1	KRG_1	$KROG_1$	$PCOW_1$	$PCGO_1$
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
SAT_n	$KROW_n$	KRW_n	KRG_n	$KROG_n$	$PCOW_n$	$PCGO_n$

Enter the following seven variables on each line of the table. The phase saturation is the independent parameter. The values of the saturation–dependent parameters will be linearly interpolated from the table entered. It is often convenient to extend the table 10% beyond the physical limits of fluid saturation to insure interpolation of a value that is meaningful even if the numerical solution is becoming unstable. Input the SAT values and permeabilities as decimals, not as percentages:

- SAT Phase saturation ($SAT_1 = -0.10 \ldots SAT_n = 1.10$)
- KROW Oil relative permeability in the presence of SAT_i oil saturation.
- KRW Water relative permeability in the presence of SAT_i water saturation.
- KRG Gas phase relative permeability in the presence of SAT_i gas saturation.
- KROG Oil relative permeability in the presence of SAT_i total liquid (irreducible water plus oil) saturation.
- PCOW Oil/water capillary pressure (psi) in the presence of SAT_i water saturation.
- PCGO Gas/oil capillary pressure (psi) in the presence of SAT_i gas saturation.

The values on a given line in the table are relative to the saturation (SAT_i) on that line. If $SAT_i = 0.20$, the values following on that line are in the presence of 20% saturation.

NOTE: KROG is used only when three–phase oil relative permeability is calculated (Line 4 below). When ITHREE = 1, KROW and KRW represent a water–oil system while KROG and KRG represent a gas–oil system. See Appendix C for more information.

Line 3 **Title**

Line 4 **Three–phase oil relative permeability option; irreducible water saturation**

The first of two variables on this line is ITHREE. To make a three–phase calculation of oil relative permeability, set ITHREE = 1; otherwise, set ITHREE = 0.

The second variable, SWR, is the decimal form of the irreducible water saturation.

Fluid Data Tables

Chapter 7

This chapter describes how to enter a separate set of pressure–dependent data for each PVT type. An entire set of set data, Lines 1 through 19, is entered for each PVT type. Enter the entire sequence NPVT times (NPVT, Chapter 5, Line2).

Line 1 **Title**

Line 2 **Bubble point data**

Enter three values:

- PBO Initial oil bubble point pressure (psia).
- PBODAT Depth datum (ft) at which bubble point applies.
- PBGRAD Constant bubble point pressure gradient (psi/ft).

The bubble point pressure at a grid block (I,J,K) is computed as below:

PBOT(I,J,K) = PBO + (PBODAT – EL(I,J,K)) × PBGRAD.

Line 3 **Title**

Line 4 **Undersaturated oil properties, PVT constraint, bubble point algorithm**

The five variables for this line are:

- VSLOPE The slope of the oil viscosity vs. pressure curve for undersaturated oil (oil at a pressure greater than the initial bubble point). This value ($\Delta\mu_o/\Delta P$) is in cp/psi.
- BSLOPE The slope of the oil formation volume factor vs. pressure curve for undersaturated oil. This value ($\Delta B_o/\Delta P$) is in RB/(STB × psi). This value should be negative and is not the same as undersaturated oil compressibility.
- RSLOPE The slope of the solution gas/oil ratio vs. pressure curve for undersaturated oil. This value ($\Delta RSO/\Delta P$) is in SCF/(STB × psi). This will normally be zero.
- PMAX The maximum pressure entry allowed in all PVT tables.
- IREPRS Code for setting repressurization algorithm on or off:
  0 – algorithm is on,
  1 – algorithm is off (see Appendix D).

NOTE: The repressurization algorithm calculates a new bubble point when the reservoir is repressured, for example, by waterflooding. However, the algorithm is based on correlations and does not always gives an accurate representation of the reservoir fluid of interest.

Line 5 **Title**

Line 6 **Oil PVT table**

Enter the following table, which describes oil properties at various pressures:

P_1	MUO @ P_1	BO @ P_1	RSO @ P_1
.	.	.	.
.	.	.	.
.	.	.	.
P_{max}	MUO @ P_{max}	BO @ P_{max}	RSO @ P_{max}

The variables have the following meanings:

- P Pressure (psia). P_1 normally is 14.7. The entries must increase from P_1 to P_{max}. The last entry, P_{max}, must be equal to PMAX given in Line 4.
- MUO Saturated oil viscosity (cp).
- BO Saturated oil formation volume factor (RB/STB).
- RSO Saturated oil solution gas/oil ratio (SCF/STB).

NOTE: The bubble point tracking routine in BOAST requires that the above oil properties be entered as saturated oil data over the entire pressure range. BOAST will use the saturated oil data above the initial bubble point only if the local reservoir pressure rises above the initial bubble point pressure and if free gas is introduced into the region (e.g., pressure maintenance by gas injection into the oil zone).

Laboratory saturated oil data generally will have to be extrapolated above the measured bubble point pressure to cover the entire pressure range anticipated for the simulation run.

Line 7 **Title**

Line 8 **Water PVT table**

Enter the following table, which describes water properties at various pressures:

P_1	MUW @ P_1	BW @ P_1	RSW @ P_1
.	.	.	.
.	.	.	.
.	.	.	.
P_{max}	MUW @ P_{max}	BW @ P_{max}	RSW @ P_{max}

The variables in the water PVT table have the following meanings:

- P Pressure (psia). P_1 normally is 14.7. The entries must increase from P_1 to P_{max}. The last entry, P_{max}, must be equal to PMAX given in Line 4.
- MUW Water viscosity (cp).
- BW Water formation volume factor (RB/STB).
- RSW Water solution gas/water ratio (SCF/STB).

NOTE: For many Reservoirs, the solubility of gas in the water often is assumed to be negligible. In this case, set RSW = 0.0 for all pressures.

BOAST uses the water PVT table for such situations as when geopressured aquifers produce gas or whenever gas solubility in water contributes significantly to the available gas.

Line 9 Title

Line 10 Gas properties code

Enter a value for KGCOR to prepare BOAST for the correct table. The variable also activates the gas correlation option.

If KGCOR = 0, BOAST will read the gas and rock properties table.

If KGCOR = 1, the gas correlation option (see Appendix D) is activated, and BOAST will read the rock compressibility versus pressure table.

Line 11 Title

Line 12 Gas and rock properties table

Omit this Line if KGCOR = 1.

Enter the following table, which describes gas properties and rock compressibilities at various pressures:

P_1	MUG @ P_1	BG @ P_1	PSI @ P_1	CR @ P_1
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
P_{max}	MUG @ P_{max}	BG @ P_{max}	PSI @ P_{max}	CR @ P_{max}

The variables in the gas and rock properties table have the following meanings:

- P Pressure (psia). P_1 normally is 14.7. The entries must increase from P_1 to P_{max}. The last entry, P_{max}, must be equal to PMAX given in Line 4.
- MUG Gas viscosity (cp).

- BG Gas formation volume factor (RCF/SCF).
- PSI Gas pseudo–pressure ($psia^2/cp$).
- CR Pressure–dependent rock compressibility (psi^{-1}).

NOTE: If KGCOR = 1, proceed to Line 18.

Line 13 Gas PVT correlation parameters

Omit this Line if KGCOR = 0.

Enter four variables as follows:

- KODEA Gas property description. Enter a value according to the table below.
- MPGT Number of entries in the gas PVT table. The value must be from 2 to 25.
- TEM Reservoir temperature (°F).
- SPG Gas specific gravity (air = 1.0)

KODEA	Specify one of four general gas compositions to be entered on Line 14. Enter 1, 2, 3, or 4.
1	Sweet gas. Input mole fractions as: 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.
2	Sour gas. Input the mole fractions of H_2S CO_2 N_2 y_1 0. 0. 0. 0. 0. 0. 0. 0. where the mole fractions of $(H_2S + CO_2 + N_2) + y_1 = 1.0$.
3	Sweet or sour gas. Enter the mole fractions of these 12 components: H_2S CO_2 N_2 C_1 C_2 C_3 iC_4 nC_4 iC_5 nC_5 C_6 C_{7+}.
4	The same as when KODEA = 3, but also enter three more variables: critical pressure, critical temperature, and the molecular weight of C_{7+}.

Line 14 **Gas composition**

Omit this Line if KGCOR = 0.

Enter the mole fractions (in decimal form) of the following 12 components:

H_2S CO_2 N_2 C_1 C_2 C_3 iC_4 nC_4 iC_5 nC_5 C_6 C_{7+}

The sum of the 12 mole percentages must be 1.0. The array name is FRCI.

Line 15 **C_{7+} properties**

Omit this Line if either KGCOR = 0 or KODEA ≠ 4.
(Enter this Line only if KGCOR = 1 and KODEA = 4.)

Enter three properties about C_{7+}:

- PRSCI Critical pressure (psia).
- TEMCI Critical temperature (°R)
- RMWTI Molecular weight

Line 16 **Title**

Omit this Line if KGCOR = 0.

Line 17 **Rock compressibility**

Omit this Line if KGCOR = 0.

The user may set rock compressibility to be either constant or pressure dependent. To specify constant rock compressibility, input the following two values:

- PMAX Maximum table pressure from Line 4.
- CR Rock compressibility @ PMAX (1/psia).

To specify pressure–dependent rock compressibility, input a table of MPGT lines (see Line 13) as shown below:

P_1	CR @ P_1
.	.
.	.
.	.
P_{max}	CR @ P_{max}

Line 18 **Title**

Line 19 **Stock tank fluid densities**

Enter the density for each of the three fluids as follows:

- RHOSCO Stock tank oil density (lb/cu ft) at standard conditions.
- RHOSCW Stock tank water density (lb/cu ft) at standard conditions.
- RHOSCG Gas density (lb/cu ft) at standard conditions.

NOTE: Enter Lines 1 through 19 NPVT times.

BOAST contains two options for pressure initialization and two options for saturation initialization.

The pressure distribution can be calculated by the program for equilibrium conditions given the location of the gas/oil contact and the oil/water contact, as well as the pressure at both contacts. Otherwise, the user can enter the initial pressure distribution block–by–block; this would be non–equilibrium initialization.

Saturations for oil, gas, and water (SO, SG, and SW) can be read as constant values over the entire grid, or saturation distributions can be read block–by–block; in this case SO and SW are entered, and BOAST calculates SG as 1.0 – SO – SW.

Line 1 Title

Line 2 Initialization codes, depth of datum, fluid gradient

The four variables on this (one) line are KPI, KSI, PDATUM, and GRAD. The first two variables operate as shown:

KPI	Pressure initialization code
0	BOAST calculates equilibrium initialization. Input will be pressures at the gas/oil and water/oil contacts, and the depths at those two contacts.
1	Initialize block–by–block (non–equilibrium initialization). Read II × JJ × KK values.
KSI	**Saturation initialization code**
0	Initial oil, gas, and water saturations are constant over the entire grid. Enter three values: SOI, SGI, and SWI.
1	Define saturation initialization block–by–block. Enter the array SO, then an array SW. Each array has II × JJ × KK values. BOAST automatically calculates SG.

- PDATUM The depth of the pressure datum (ft).
- GRAD Fluid gradient for pressure corrections to PDATUM. Given in psi/ft. Every time that a pressure map is printed, the simulator will also print a map of pressures corrected to PDATUM. This holds true unless GRAD = 0; then BOAST will not print a map of corrected pressures. See Appendix F.

Line 3 **Equilibrium pressure initialization**

Omit this Line if KPI = 1.

Enter NROCK lines each with these four variables:

- PWOC Pressure (psia) at the water/oil contact.
- WOC Depth (ft) to the water/oil contact.
- PGOC Pressure (psia) at the gas/oil contact.
- GOC Depth (ft) to the gas/oil contact.

NOTE: Enter NROCK lines, each containing the four variables (one line for each rock type). The first line defines pressure in rock type 1, and so on. The last line defines pressure in rock type NROCK.

Line 4 **Non–equilibrium pressure initialization**

Omit this Line if KPI = 0.

Enter an initial pressure (psia) for each block in the grid (II × JJ × KK values).

Line 5 **Constant saturation initialization**

Omit this Line if KSI = 1.

Enter NROCK lines each with these three variables:

- SOI Initial oil saturation (a decimal) to be assigned to all blocks of this rock type.
- SWI Initial water saturation (decimal) to be assigned to all blocks of this rock type.
- SGI Initial gas saturation (decimal) to be assigned to all blocks of this rock type.

NOTE: The three variables must be entered NROCK times, once for each rock type. The first line defines saturation in rock type 1; the second line defines it in rock type 2, etc. The last line defines saturation in rock type NROCK.

Line 6 **Block–by–block saturation initialization**

Omit this Line if KSI = 0.

Enter two arrays, SO and SW.

SO contains II × JJ × KK initial oil saturation values (decimals); SW contains II × JJ × KK initial water saturation values (also decimals).

In every block, gas saturation is calculated automatically as follows:

$$SG_{ijk} = 1.0 - SO_{ijk} - SW_{ijk}.$$

Diagnostic Output

Chapter 9

BOAST provides several diagnostic features for debugging a simulation. These options do not check the input file. Rather, they simply provide information about variables during the run.

NOTE: Normally, each of these codes should be set to zero (0). Activating any of the options generates an extremely large volume of output.

Line 1 **Title**

Line 2 **Codes for diagnostic output**

Four codes, KSN1, KSM1, KCO1, and KCOF, control diagnostic output for the following options. Enter four values:

- KSN1 SOR parameters.
- KSM1 Solution matrix.
- KCO1 Compressibility and volume factor.
- KCOF Density and saturation.

If a code is set to 1, BOAST will print diagnostic output for that option.
If a code is set to 0, there is no output for that option.

The run control section of the data defines the boundary conditions under which the program runs.

Two variables discussed below, FACT1 and FACT2, constitute automatic time–step control (ATSC). This feature helps the simulation to run more efficiently by increasing the time–step size when changes are occurring slowly in the model and by decreasing the time–step size when model conditions are changing rapidly.

Line 1 Title

Line 2 Run control parameters

Enter the following eight variables:

- NMAX — Maximum number of time steps executed before the simulation terminates. TMAX ≥ 1.
- FACT1 — Factor for increasing time–step size under automatic time–step control. FACT1 ≥ 1.0. If FACT1 = 1.0, time–step size will not increase. Commonly, FACT1 = 1.25.
- FACT2 — Factor for decreasing time–step size under automatic time–step control. FACT2 ≤ 1.0. If FACT2 = 1.0, time–step size will not decrease. Commonly, FACT2 = 0.5.
- TMAX — Maximum calendar time (days) to be simulated in the run. TMAX ≥ 1. The simulation terminates after TMAX days.
- WORMAX — Maximum water/oil ratio for a producing oil well (see Appendix G). To establish a single maximum water/oil ratio, set WORMAX > 0.0. Otherwise, to establish a maximum water/oil ratio for each rock type, set WORMAX = 0.0, and notice Line 3.
- GORMAX — Maximum gas/oil ratio (SCF/STB) for a producing oil well (see Appendix G). To establish a single maximum gas/oil ratio, set GORMAX > 0.0. To establish a maximum gas/oil ratio for each rock type, set GORMAX = 0.0, and notice Line 4.
- PAMIN — Minimum field average pressure (psia). PAMIN ≥ 1. The run terminates when the average reservoir pressure falls below PAMIN.
- PAMAX — Maximum field average pressure (psia). PAMAX > PAMIN. The run is terminated when the average reservoir pressure exceeds PAMAX.

NOTE: For fixed time steps, set both FACT1 and FACT2 = 1.0. Do not set just one of the factors = 1.0 because BOAST must be able to offset any change in the size of the time step. See Appendix H for further discussion.

Line 3 **Limit the water/oil ratio in each rock type**

Omit this Line if WORMAX > 0.0.

WORMAX has been set to the special value 0.0. Here, enter the maximum acceptable water/oil ratio for each rock type (up to five). The array name is WOROCK.

Enter NROCK values of WOROCK, all on one line. The first value limits the water–oil ratio in rock type 1; the second value limits the water–oil ratio in rock type 2, and so on.

NOTE: If a well is completed in more than one rock region, the well will be limited by the greatest acceptable water/oil ratio of all the rock types penetrated by that well.

Line 4 **Limit the gas/oil ratio in each rock type**

Omit this Line if GORMAX > 0.0.

GORMAX has been set to the special value 0.0. Here, enter the maximum acceptable gas/oil ratio for each rock type (up to five). The array name is GOROCK.

Enter NROCK values of GOROCK, all on one line. The first value limits the gas–oil ratio in rock type 1; the second value limits the gas–oil ratio in rock type 2, and so on for the (up to five) rock types.

NOTE: If a well is completed in more than one rock region, the well will be limited by the greatest acceptable gas/oil ratio of all the rock types penetrated by that well.

Here, the user specifies the algorithm for BOAST to use to solve systems of linear equations. BOAST offers two direct solution methods (BAND and D4 algorithms) and several iterative solution methods (successive overrelaxation [SOR] algorithms). The user also sets several controls under which BOAST will proceed to a solution.

Line 1 Title

Line 2 Solution method parameters

The first entry on this line is the code for the solution method, KSOL. BOAST offers eight algorithms. Set KSOL to one of the following (1–8):

1	BAND algorithm.
2	D4 algorithm.
3	LSORX (Line SOR with x–direction tridiagonal algorithm).
4	LSORY (Line SOR with y–direction tridiagonal algorithm).
5	LSORZ (Line SOR with z–direction tridiagonal algorithm).
6	L2SORX (Two–line SOR with x–direction tridiagonal algorithm).
7	L2SORY (Two–line SOR with y–direction tridiagonal algorithm).
8	L2SORZ (Two–line SOR with z–direction tridiagonal algorithm).

Guidelines for when to apply the various methods:

BAND algorithm — for one–dimensional problems.

D4 algorithm — for small two–dimensional problems or very small three–dimensional problems.

Line SOR methods (3, 4, or 5) — for larger two–dimensional problems and almost every three–dimensional problem (see the discussion below). Almost all field–sized problem requires an SOR method. Storage requirements make the direct solutions impractical for large matrices.

The particular SOR method selected should be aligned in the direction of maximum flux. For example, in a three–dimensional model with communication between layers, the greatest transmissibility is usually in the z–direction; therefore, the best SOR method usually is LSORZ. LSORX and LSORY are suitable most often for horizontal two–dimensional models.

NOTE: The two–line SOR solutions (6, 7, and 8) are not recommended for typical field problems. They are included for use by simulation researchers and for possible expansion later. Refer to Appendix B for more on choosing the solution method.

So far, only one value (KSOL) is on the line. Add the following values:

- MITER — Maximum number of SOR iterations per time step. MITER ≥ 1. A typical value is 100.
- OMEGA — Initial SOR acceleration parameter. OMEGA should be between 1.0 and 2.0, depending on the particular problem. Typically, when TOL1 = 0.0, OMEGA = 1.7, and when TOL1 ≠ 0.0, OMEGA = 1.5. The program will optimize OMEGA as the simulation proceeds.
- TOL — Maximum acceptable pressure change for SOR convergence.
- TOL1 — Parameter for determining when to change OMEGA. If TOL1 = 0.0, OMEGA will not change from its initial value (hence, OMEGA will not be optimized). A typical value for TOL1 is 0.001.
- DSMAX — Maximum acceptable change (fraction) in saturation over a time step. If the saturation (of oil, gas, or water) in any grid block changes by more than DSMAX over one time step, the time–step size will be reduced by FACT2, and the time step will be repeated. 0.0 < DSMAX ≤ 1.0. A typical value for DSMAX is 0.05.
- DPMAX — Maximum acceptable change (psi) in pressure over a time step. If the pressure change in any grid block exceeds DPMAX over one time step, the time–step size will be reduced by FACT2, and the time step will be repeated. DPMAX ≥ 1. A typical value for DPMAX is 100.

Line 3 Title

Line 4 Codes for numerical dispersion and formulation

Enter the following three variables on one line:

- NUMDIS — Numerical dispersion code:
  0 – One–point upstream weighting (standard);
  1 – Two–point upstream weighting*.
- IRK — Formulation code:
  0 – Standard IMPES;
  1 – Stabilized IMPES (see Appendix A).
- THRUIN — When IRK = 1 (stabilized IMPES), this value will limit the throughput per grid block. 0.0 < THRUIN ≤ 1.0. A value between 0.5 and 1.0 is recommended.

* Watkins, R. W. SPE Paper No. 10513.

Aquifer Model

Chapter 12

In this chapter, the last in Initialization Data, the user specifies an aquifer model and parameters (there is a choice in case there is no aquifer to model). BOAST offers three analytic models. The three models are the pot aquifer, the steady–state aquifer, and the Carter–Tracy aquifer. For any given model, the user may choose the aquifer influx strength.

Line 1 Title

Line 2 Aquifer model

Enter only one variable, IAQOPT, a number representing the choice for the model. The choice may be to use no aquifer model (IAQOPT = 0). Choose IAQOPT according to the following table:

IAQOPT	Choice of aquifer
0	No aquifer model will be used.
1	Pot model.
2	Steady–state model.
3	Carter–Tracy model with $r_e/r_w = 1.5$.
4	Carter–Tracy model with $r_e/r_w = 2.0$.
5	Carter–Tracy model with $r_e/r_w = 3.0$.
6	Carter–Tracy model with $r_e/r_w = 4.0$.
7	Carter–Tracy model with $r_e/r_w = 5.0$.
8	Carter–Tracy model with $r_e/r_w = 6.0$.
9	Carter–Tracy model with $r_e/r_w = 8.0$.
10	Carter–Tracy model with $r_e/r_w = 10.0$.
11	Carter–Tracy model with $r_e/r_w = \infty$.

NOTE: If IAQOPT = 0, go to the Recurrent Data section.
If IAQOPT = 1, go to Line 3.
If IAQOPT = 2, go to Line 5.
If IAQOPT ≥ 3, go to Line 7.

Line 3 Number of pot aquifer regions

Enter NAQEN, the number of grid regions containing a pot aquifer.

NOTE: Input in Lines 4, 6, and 10 will begin with I_1, I_2, J_1, J_2, K_1, and K_2. These variables are interpreted as follows:

> $I_1 - I_2$ is the range of the aquifer in the x–direction;
> $J_1 - J_2$ is in the y–direction;
> $K_1 - K_2$ is in the z–direction.

Line 4 **Pot aquifer parameters**

Enter NAQEN lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 POT,

where POT is the pot aquifer coefficient (SCF/psi).

Go to the next section, Recurrent Data.

Line 5 **Number of steady–state aquifer regions**

Enter NAQEN, the number of grid regions containing a steady–state aquifer.

Line 6 **Steady–state aquifer parameters**

Enter NAQEN lines, each containing the following 7 values:

I_1 I_2 J_1 J_2 K_1 K_2 SSAQ,

where SSAQ is the steady–state aquifer coefficient (SCF/day/psi).

Go to the next section, Recurrent Data.

Line 7 **Number of distinct types of Carter–Tracy aquifers**

Enter NAQREG, the number of distinct Carter–Tracy aquifers.

This line does not describe any number of regions but rather the number of unique Carter–Tracy aquifers. Each of these aquifers will have a unique set of parameters; there may be multiple occurrences for each type of Carter–Tracy aquifer.

NOTE: Lines 8, 9, and 10 are entered NAQREG times, once for each unique Carter–Tracy aquifer. Line 10 is entered once for each occurrence of a particular Carter–Tracy type (from Line 8).

Line 8 **Carter–Tracy aquifer parameters**

Enter the following 8 values:

- AQCR Aquifer rock compressibility (1/psia).
- AQCW Aquifer water compressibility (1/psia).

- AQMUW Aquifer water viscosity (cp).
- AQK Aquifer permeability to water (md).
- AQPHI Aquifer porosity (fraction).
- AQH Aquifer net thickness (ft).
- AQS Aquifer/reservoir boundary interface ($0.0 \leq AQS \leq 1.0$).
- AQRE External radius of aquifer (ft).

Line 9 **Number of Carter–Tracy aquifer regions as described in Line 8**

This value, NAQEN, is the number of regions of the aquifer with the particular parameters described in Line 8.

Line 10 **Carter–Tracy aquifer boundaries**

Enter NAQEN lines, each with the following 6 values:

I_1 I_2 J_1 J_2 K_1 K_2.

NOTE: See the Note above Line 8. The Initialization Data section ends here.

Recurrent Data III

Recurrent data differs from initialization data in two ways: 1) it is retrieved in segments throughout the simulation, and 2) it describes the current running conditions of the simulation. (Initialization data was read in all–at–once at the beginning.)

For example, the recurrent input may comprise several data sets. BOAST can read a data set, operate under those conditions for the specified time, then read the next data set for new conditions. Each set describes time–step control, well completions and performance, and specifications for the frequency and type of output.

The two chapters in the Recurrent Data section describe one data set:

13. Time–step and output control
14. Well Information

Repeat Chapters 13 and 14 for each additional data set.

Enter the following title line:

Line 1 Title

This line is to mark the beginning of the recurrent data. Choose any phrase (limited to 80 characters) for the title, e.g., *Recurrent Data.*

Do not repeat this line for each data set. This line is entered only once.

Time–Step and Output Control Chapter 13

This chapter controls the advancement of the time step and specifies the frequency and type of output.

Line 1 Title

Line 2 Time–step and output parameters

Enter the following three values on this line:

- ICHANG The number of time steps over which time–step and output control will apply (when IOMETH = 0). If IOMETH ≠ 0, ICHANG is ignored.
- IWLCNG Code for BOAST to read or not to read the well information lines:
  0 – Do not read the well information lines (Chapter 14);
  1 – BOAST will read the well information lines during the first time step of this interval. These lines may not be omitted.
- IOMETH Code for the output method:
  0 – Output is based on time steps;
  ≥1 – Output is based on elapsed time.

HOW LONG WILL A DATA SET APPLY?

When IOMETH = 0:
If ATSC is not activated, the current data set will control the simulation for exactly ICHANG time steps. If ATSC is activated, the data set will operate for ICHANG × DT days. (The variable DT is in Line 6.)

When IOMETH = 1, 2, . . . :
The current data set applies through the last elapsed time in the FTIO array on Line 4.

Line 3 Frequencies of output for well reports and summary reports (see next page)

Enter this Line only if IOMETH = 0 (output is based on time steps).

Enter the two values described below:

- IWLREP The number of time steps between the output of each well report.
- ISUMRY The number of time steps between the output of each time–step summary.

The *well report* summarizes cumulative production and injection data for each layer and for the entire field. The *summary report* is a concise description of total field production and injection, aquifer influx formation, time–step and material balance information, the locations where maximum pressure and saturation changes are occurring in the reservoir. Refer to Chapter 16, Recurrent Output.

Line 4 **Elapsed times at which to output well and summary reports**

Enter this Line only if IOMETH ≥ 1 (output is based on elapsed time of the run).

Enter IOMETH elapsed–time values (days) into the array FTIO. When the elapsed time of the run equals an FTIO value, BOAST will output both the well report and the time–step summary report. Enter the elapsed–time values in increasing order.

Line 5 **Map print codes**

The user may instruct BOAST to print certain values present in the grid. Enter a code (0, 1, or 2) for each of the following six options.

PRINT CODES

Term	Variable to be mapped
IPMAP	Grid block pressures
ISOMAP	Grid block oil saturations
ISWMAP	Grid block water saturations
ISGMAP	Grid block gas saturations
IPBMAP	Grid block saturation pressures (normally = 0)
IAQMAP	Grid block aquifer influx in each grid block

0 — Do not print
1 — Print the array
2 — Print array and printer plot

Note: These maps are printer output and part of the summary report.

Line 6 **Time–step control**

Enter the following three variables:

- DT — The size of the initial time step, related in days (e.g., 0.5 days). Automatic time–step control (ATSC) may adjust this value depending on the changes in pressure and saturation.
- DTMIN — The minimum allowable time step size. ATSC cannot decrease the time–step size below this value. (Common value = 0.1 days.)
- DTMAX — The maximum allowable time–step size. ATSC cannot increase the time–step size beyond this value. (Common value = 30 days.)

If IWLCNG = 0, skip this chapter and go to the Output Evaluation section. Otherwise (IWLCNG = 1) proceed into this chapter. The variable IWLCNG is from Chapter 13, Line 2.

This chapter contains the schedule of well completions and performance.

Line 1 **Title**

Line 2 **Number of well changes**

Enter the following two variables:

- NWELLN The number of new wells for which complete well information is to be read. Lines 4 through 8 must be are entered NWELLN times.
- NWELLO The number of previously defined wells for which new rates and/or rate controls are to be read. Lines 10 and 11 must be entered NWELLO times.

NOTE: If NWELLN = 0, proceed to Line 9

Line 3 **Title — complete information for new well(s)**

Omit this Line if NWELLN = 0.

Line 4 **Well characteristics**

Omit this Line if NWELLN = 0.

Enter the following six variables:

- WELLID A well name of up to five characters, entered within the first five positions on the line.
- IDWELL A unique well identification number. Start this entry at or beyond position six on the line. If two wells are defined with the same number, the characteristics of the most recently entered well are used.
- I The x–coordinate of the well's grid location.
- J The y–coordinate of the well's grid location.
- PERFl The number of the uppermost completion layer for this well.
- NLAYER The number of consecutive completion layers beginning with and including PERFl.

Line 5 **Well flow index**

Omit this Line if NWELLN = 0.

Enter a layer flow index for each layer for rates in STB/D. There will be NLAYER entries. The array name is PID. Estimate the layer flow index as follows:

$$PID = \left[\frac{0.00708 \times K \times h}{\ln\left(\frac{0.121\sqrt{DX \times DY}}{r_w} \right) + S} \right]$$

The equation is interpreted as indicated below:

- K Layer absolute permeability (md).
- h Layer thickness (ft).
- DX X–direction grid block dimension (ft).
- DY Y–direction grid block dimension (ft).
- r_w Wellbore radius (ft).
- S Skin factor.

Line 6 **Flowing bottom–hole pressures**

Omit this Line if NWELLN = 0.

Enter the flowing bottom–hole pressure (FBHP) for each layer for this well. There will be NLAYER entries, given in psia. The first entry corresponds to the shallowest layer, and so on. The array name is PWF.

NOTE:

1. The values in PWF will be used only if KIP (Line 7) is negative; however, these variables must be entered in any case.

2. Once a well has been completed in a certain layer, the user must continue specifying that layer in all succeeding Lines, even if the layer or the entire well is shut in.

3. To shut in a layer, set the PID of that layer to 0. To shut in an entire well, set the PID of every layer in the well to 0.

Line 7 **Rate control and information**

Omit this Line if NWELLN = 0.

This Line contains seven variables. The first three are described below:

- WELLID A well name of up to five characters, entered within the first five positions on the line.
- IDWELL A unique well identification number. Start this entry at or beyond position six on the line. If two wells are defined with the same number, the characteristics of the more recently entered well are used.
- KIP A code that specifies both the well type and how the well's performance (production or injection) is determined, either by specified rates or specified FBHP.

Choose a value for KIP according to the table below:

KIP	RATE–CONTROLLED WELLS
3	Gas well.
2	Water well.
1	Production well.

	BHP–CONTROLLED WELLS (with optional rate constraint)
–1	Oil and/or water production well — PI and FBHP control.
–2	Water well — PI and FBHP control.
–3	Gas injection well — PI and FBHP control.
–4	Gas production well — LIT representation.
–11	Production well — PI and FBHP control.
–12	Water well — PI and FBHP control.
–13	Gas well — PI and FBHP control.

The final four variables on this Line follow in the order in which they must be entered:

- QO
- QW
- QG
- QT

The entries for these variables depend on the value of KIP. Using the tables below, find the appropriate KIP value (already entered on this Line), and enter four corresponding rates. Remember to enter them in the order shown above.

RATE–CONTROLLED WELLS

KIP	Variable Assignments
3	QG (gas rate) > 0 MCF/D; QO = QW = QT = 0.
2	QW (water rate) > 0 STB/D; QO = QG = QT = 0.
1	QO (oil rate) > 0 STB/D; QW = QG = QT = 0. or QT (total fluid voidage rate) > 0 RB/D; QO = QW = QG = 0.

NOTE:

1. The BHP–controlled wells (on the following page) have a rate constraint option. With this option, the user limits daily production or injection rates. However, the user can deactivate the constraint for any given rate by setting that rate to 0.0.

2. Production rates are signed positive (withdrawal is positive); injection rates are negative. Therefore, QW in both KIP = –2 and KIP = –3 should be negative (or 0.0 to deactivate the constraint).

BHP–CONTROLLED WELLS
(with optional rate constraint)

KIP

–1 QO = minimum oil production rate required (STB/D).
QW = maximum oil production rate allowed (STB/D).
QG = 0.0.
QT = maximum liquid withdrawal rate allowed (STB/D).

If the calculated oil production rate drops below QO, the well is shut in. If the calculated oil production rate exceeds QW, calculated production from each layer is reduced by the ratio of allowed to calculated oil production rates. If the sum of oil and water production exceeds QT, production from each layer is reduced by the ratio of allowed to calculated liquid withdrawal rates.

Note that the units for QT here are STB/D — not RB/D as with KIP ≥ 1.

–2 QO = 0.0
QW = maximum water injection rate allowed (STB/D).
QG = 0.0.
QT = 0.0

If the calculated water injection rate exceeds QW, calculated water injection into each layer is reduced by the ratio of allowed to calculated water injection rates.

Remember that QW is negative (or 0.0 to deactivate the constraint).

–3 QO = 0.0
QW = 0.0.
QG = maximum gas injection rate allowed (MCF/D).
QT = 0.0

If the calculated gas injection rate exceeds QG, calculated gas injection into each layer is reduced by the ratio of allowed to calculated gas injection rates.

Remember that QG is negative (or 0.0 to deactivate the constraint).

SHUT-IN WELLS

KIP	Variable Assignments
–4	QO = QW = QG = QT = 0.
–11	QO = QW = QG = QT = 0.
–12	QO = QW = QG = QT = 0.
–13	QO = QW = QG = QT = 0.

Line 8 **Pressure–constrained gas well (LIT representation)**

Omit this Line if NWELLN = 0 or if KIP ≠ –4.

Two of the variables on this Line, ALIT and BLIT, are parameters for the LIT model. Values for these entries can be obtained from the program GASDEL. The program is available through **K&A Energy Consultants, Inc.**

Enter the following four variables:

- WELLID A well name of up to five characters, entered within the first five positions on the line.
- IDWELL A unique well identification number. Start this entry at or beyond position six on the line. If two wells are defined with the same number, the characteristics of the more recently entered well are used.
- ALIT The *a* coefficient for the LIT method of gas well analysis.
- BLIT The *b* coefficient for the LIT method of gas well analysis.

The units for ALIT are: $\frac{\text{MMSCF/D}}{\text{psia}^2/\text{cp}}$

The units for BLIT are: $\frac{(\text{MMSCF/D})^2}{\text{psia}^2/\text{cp}}$

NOTE: Recall from Line 2 that Lines 4 through 8 must be entered NWELLN times.

Line 9 **Title — new rates for previously defined well(s)**

Omit this Line if NWELLO = 0.

Line 10 Rate control and information for previously defined wells

Omit this Line if NWELLO = 0.

The content of this Line is identical to that in Line 7; therefore, the tables are not repeated. Refer to the tables in Line 7 for entering values for KIP, QO, QW, QG, and QT.

This Line contains seven variables. The first three are described below:

- WELLID A well name of up to five characters, entered within the first five positions on the line.
- IDWELL A unique well identification number. Start this entry at or beyond position six on the line. If two wells are defined with the same number, the characteristics of the more recently entered well are used.
- KIP A code that specifies both the well type and how the well's performance (production or injection) is determined, either by specified rates or specified FBHP.

Choose a value for KIP according to the first table in Line 7.

The final four variables on this Line follow in the order in which they must be entered:

- QO
- QW
- QG
- QT

The entries for these variables depend on the value of KIP. Using the tables from Line 7, find the appropriate KIP value (already entered on this Line), and enter four corresponding rates.

NOTE:

1. The BHP–controlled wells have a rate constraint option. With this option, the user limits daily production or injection rates. However, the user can deactivate the constraint for any given rate by setting that rate to 0.0.

2. Production rates are signed positive (withdrawal is positive); injection rates are negative. Therefore, QW in both KIP = –2 and KIP = –3 should be negative (or 0.0 to deactivate the constraint).

Line 11 **Pressure–constrained gas well (LIT representation)**

Omit this Line if NWELLO = 0 or if KIP ≠ –4.

Two of the variables on this Line, ALIT and BLIT, are parameters for the LIT model. Values for these entries can be obtained from the program GASDEL. The program is available through **K&A Energy Consultants, Inc.**

Enter the following four variables:

- WELLID A well name of up to five characters, entered within the first five positions on the line.
- IDWELL A unique well identification number. Start this entry at or beyond position six on the line. If two wells are defined with the same number, the characteristics of the more recently entered well are used.
- ALIT The *a* coefficient for the LIT method of gas well analysis.
- BLIT The *b* coefficient for the LIT method of gas well analysis.

NOTE: Recall from Line 2 that Lines 10 and 11 must be entered NWELLN times.

Output Evaluation IV

This section discusses all types of BOAST program output—from the input data to plots to error messages. Four chapters describe what output the user can obtain and where the output is controlled in the input file:

15. Initialization Output
16. Recurrent Output
17. Summary Plots
18. Initialization Error Messages

Initialization output is produced once and reflects the original state of the simulation. BOAST produces this output at the beginning of the run. For every run, BOAST produces output on the following topics:

- Grid block size
- Node (midpoint) elevations
- Porosity distribution
- Permeability distribution
- Rock– and PVT–type distribution
- Relative permeability and capillary pressure tables for each rock type
- PVT tables for each PVT region
- Slopes calculated from PVT data for each PVT type
- Time–step control data
- Aquifer model parameters
- Initial well information
- Initial fluid volumes–in–place
- Initial pressure and saturation arrays

In three places in the input, the user may modify a previously entered array. In each case, BOAST reads a print code for whether or not to output the modified array. This output would be in addition to the above list of topics. See IDCODE, 3.2–2, IPCODE, 4.2–2, and ITCODE, 4.3–2.

Recurrent Output

Chapter 16

Recurrent output is derived while the simulation is underway and may be produced repeatedly. The four types of recurrent output are discussed below.

Time–step summary

The user cannot deactivate this feature, which displays run–control information at every time step. This output provides important information about the progress and stability of the run. This material includes fluid production and injection rates over a time step, the pore volume weighted average reservoir pressure, material balance checks, and the locations and magnitudes of maximum saturation and pressure changes.

Well report

The well report summarizes cumulative production/injection data for each layer as well as for the entire field. The user controls the frequency of output for the well report on Line 3 or 4 of Chapter 13, depending on the value chosen for IOMETH in Line 2. GOR is reported in SCF/STB, WOR in STB/STB.

Summary report

This report is the most valuable for monitoring model performance. It is a concise summary of total field production and injection, aquifer influx formation, time–step and material balance information, and a determination of where maximum pressure and saturation changes are occurring in the reservoir. Summary reports are produced at the same times as well reports (above).

Grid parameter values

BOAST can also provide output on parameter distributions throughout the grid. For each grid block, the printouts show pressure, oil saturation, gas saturation, water saturation, saturation pressure, and aquifer influx. For each parameter, the user may print the actual array or both the actual array and an array plot. The control of this output is described in Chapter 13, on Line 5. User–selected distribution arrays (and array plots) are printed at the same times as well and summary reports (above).

The array plot simplifies the representation of the distribution of a parameter. This form of output converts the original parameter array into an array of eleven possible code values (–, 1, 2, 3, 4, 5, 6, 7, 8, 9, and T). The code for each position in the array plot (AOUT) represents a small range of values from the original array. AOUT highlights trends in parameter values and lets the user visually monitor pressure pulses and saturation fronts, for example. The output array is printed such that it can be used for drawing a rough contour map.

BOAST uses the maximum (AMAX) and minimum (AMIN) values from the original array to construct the new array using a normalized parameter, AV:

$$AV_{ijk} = (APLOT_{ijk} - AMIN)/ADIF,$$

where

$$ADIF = AMAX - AMIN > 0.001.$$

The codes displayed in the array plot (AOUT) are defined as follows:

AOUT Code	Condition
–	ADIF ≤ 0.001 or AV < 0.005
1	0.05 ≤ AV < 0.15
2	0.15 ≤ AV < 0.25
3	0.25 ≤ AV < 0.35
4	0.35 ≤ AV < 0.45
5	0.45 ≤ AV < 0.55
6	0.55 ≤ AV < 0.65
7	0.65 ≤ AV < 0.75
8	0.75 ≤ AV < 0.85
9	0.85 ≤ AV < 0.95
T	0.95 ≤ AV

Summary Plots

Chapter 17

At the end of Chapter 2, the user enters 15 codes for activating summary plots. These plots show various field–wide parameters as functions of the time–step number. Because they provide field–wide data in such a compact format, the summary plots are especially useful in quickly evaluating history–match runs.

Activated plots will appear at the end of the output. If a parameter did not change during the run, the program will not produce a summary plot for the variable.

The variables to plot are:

- Oil production rate
- Gas production rate
- Water production rate
- Producing gas/oil ratio
- Producing water/oil ratio
- Gas injection rate
- Water injection rate
- Pore volume weighted average reservoir pressure
- Aquifer influx rate
- Cumulative aquifer influx
- Cumulative oil production
- Cumulative gas production
- Cumulative water production
- Cumulative gas injection
- Cumulative water injection

The number of plot lines per time step along the time axis may be either input by the user or determined automatically, when NPLINE = 0. If the number of plot lines per time step is determined automatically, the total number of time steps run (ITSMAX) is used to determine the plot length as follows:

If $1 \leq$ ITSMAX ≤ 50, the plot length is 50 lines.
If $51 \leq$ ITSMAX ≤ 100, the plot length is 100 lines.
If $101 \leq$ ITSMAX ≤ 250, the plot length is 150 lines.
If $251 \leq$ ITSMAX ≤ 400, the plot length is 200 lines.
If $401 \leq$ ITSMAX, the plot length is 250 lines.

Initialization Error Messages

Chapter 18

BOAST checks the input data file for entries that are not feasible, for example, a porosity value less than 0 or greater than 1. This chapter explains BOAST's error messages. The outline below presents the error or warning message from BOAST, then its meaning.

Fatal Errors

Type 1 Tests applied to distributions (arrays)

GRID BLOCK DX ERROR AT IJK =
GRID BLOCK DY ERROR AT IJK =
GRID BLOCK DZ ERROR AT IJK =
GRID BLOCK DZNET ERROR AT IJK =

Meaning: grid block size is negative.

POROSITY ERROR AT IJK =

Meaning: porosity is negative or greater than 1.0.

GRID BLOCK KX ERROR AT IJK =
GRID BLOCK KY ERROR AT IJK =
GRID BLOCK KZ ERROR AT IJK =

Meaning: permeability is negative.

GRID BLOCK TX ERROR AT IJK =
GRID BLOCK TY ERROR AT IJK =
GRID BLOCK TZ ERROR AT IJK =

Meaning: transmissibility is negative.

INIT P ERROR AT GRID BLOCK IJK =

Meaning: initial pressure is negative.

INIT SO ERROR AT GRID BLOCK IJK =
INIT SW ERROR AT GRID BLOCK IJK =
INIT SG ERROR AT GRID BLOCK IJK =
INIT SAT SUM ERROR AT GRID BLOCK IJK =

Meaning: initial saturation is negative or greater than 1.0; or sum of initial saturations exceeds 1.0.

Type 2 **Tests applied to saturation–dependent properties (Appendix C)**

SAT ERROR FOR ROCK REGION . . .
FIRST SAT ENTRY FOR ROCK REGION . . .
LAST SAT ENTRY FOR ROCK REGION . . .

Meaning: saturations for relative permeability and capillary pressure tables should start at –0.1 and end at 1.10. All other saturations should be between 0.0 and 1.0, in increasing order. If these constraints are not met, a message is printed.

KROW ERROR FOR ROCK REGION . . .
KRW ERROR FOR ROCK REGION . . .
KRG ERROR FOR ROCK REGION . . .
KROG ERROR FOR ROCK REGION . . .
3–PHASE SWR ERROR FOR ROCK REGION . . .

Meaning: the relative permeability values and three–phase irreducible water saturation values should be between 0.0 and 1.0; if not, a message is printed.

PCOW ERROR FOR ROCK REGION . . .
PCGO ERROR FOR ROCK REGION . . .

Meaning: capillary pressure is negative.

Type 3 **Tests applied to pressure–dependent properties (Appendix D)**

POT ERROR FOR PVT REGION . . .
PWT ERROR FOR PVT REGION . . .
PGT ERROR FOR PVT REGION . . .
ROCK PRES ERROR FOR PVT REGION . . .

Meaning: pressure is negative or entry I is less than entry I–1.

MUO ERROR FOR PVT REGION . . .
BO ERROR FOR PVT REGION . . .
RSO ERROR FOR PVT REGION . . .
MUW ERROR FOR PVT REGION . . .
BW ERROR FOR PVT REGION . . .
RSW ERROR FOR PVT REGION . . .
MUG ERROR FOR PVT REGION . . .
BG ERROR FOR PVT REGION . . .
PSEUDO–PRES ERROR FOR PVT REGION . . .
CR ERROR FOR PVT REGION . . .

Meaning: property is negative.

ZANDC ERROR (IERR = . . .)
VISCY ERROR (IERR = . . .)

Meaning: error occurred in calculating gas properties from standard correlations.

OIL DENSITY FOR PVT REGION . . .
WATER DENSITY FOR PVT REGION . . .
GAS DENSITY FOR PVT REGION . . .

Meaning: density at standard conditions is negative.

OIL COMP ERROR FOR PVT REGION . . .
WATER COMP ERROR FOR PVT REGION . . .
GAS COMP ERROR FOR PVT REGION . . .

Meaning: a negative compressibility is computed for the given PVT data.

BUBBLE POINT PRESSURE ERROR IN . . .

Meaning: bubble point pressure is negative.

Type 4 Tests applied to codes

NO. OF X–DIRECTION BLOCKS CANNOT EXCEED . . .
NO. OF Y–DIRECTION BLOCKS CANNOT EXCEED . . .
NO. OF Z–DIRECTION BLOCKS CANNOT EXCEED . . .

Meaning: maximum number of user–specified grid blocks in the given direction exceeds the allowed number.

GAUSID ERROR: NX CANNOT EXCEED . . .
GAUSID ERROR: NY CANNOT EXCEED . . .
GAUSID ERROR: NZ CANNOT EXCEED . . .

Meaning: number of user–specified blocks in the given direction exceeds the allowed number of blocks for the KSOL = 1 solution method.

D4 ERROR: MAX NO. OF GRID . . .

Meaning: number of user–specified grid blocks exceeds the allowed number of blocks for KSOL = 2 solution method.

LSORX ERROR: NX CANNOT EXCEED . . .
LSORY ERROR: NY CANNOT EXCEED . . .
LSORZ ERROR: NZ CANNOT EXCEED . . .

Meaning: number of user–specified grid blocks in the given direction exceeds the allowed number of blocks for the KSOL = 3, 4, or 5 method.

LSOR ERROR: MAX NO. OF GRID . . .
L2SOR ERROR: MAX NO. OF GRID . . .

Meaning: total number of user–specified grid blocks exceeds the allowed number of blocks for the LSOR or L2SOR solution method.

L2SORX ERROR: NO. OF GRID BLOCKS IN Y–Z PLANE . . .
L2SORY ERROR: NO. OF GRID BLOCKS IN X–Z PLANE . . .
L2SORZ ERROR: NO. OF GRID BLOCKS IN X–Y PLANE . . .

Meaning: number of user–specified grid blocks in the given plane exceeds the allowed number of blocks for the KSOL = 6, 7, or 8 solution method.

L2SOR ERROR: NX MUST EXCEED 1
L2SOR ERROR: NY MUST EXCEED 1
L2SOR ERROR: NZ MUST EXCEED 1

Meaning: L2SOR methods (KSOL = 6, 7, and 8) cannot be used except for three–dimensional problems.

MAX # OF TIME STEPS ERROR

Meaning: the maximum number of time step is less than 1.

FACT1 ERROR
WORMAX ERROR
GORMAX ERROR

Meaning: parameter is negative.

TMAX ERROR
PAMIN ERROR
DPMAX ERROR

Meaning: parameter is less than or equal to 0.0.

DSMAX ERROR
FACT2 ERROR

Meaning: parameter is less than or equal to 0.0, or parameter is greater than 1.0.

Warning Messages

Type 1 Tests applied to PVT data (Appendix D)

VSLOPE FOR PVT REGION . . .
BSLOPE FOR PVT REGION . . .
RSLOPE FOR PVT REGION . . .

Meaning: undersaturated PVT property slope has an atypical value.

Type 2 **Tests applied to codes**

SOR DEBUG OUTPUT ON
SOLN METHOD DEBUG OUTPUT ON
COMP AND FVF DEBUG OUTPUT ON
DEN AND SAT DEBUG OUTPUT ON

Meaning: KSN1, KSM1, KCO1, or KCOF is not equal to 0. Substantial output may be generated.

ALLOWED # OF SOR ITER LESS THAN 1

Meaning: MITER is 0 or negative.

OMEGA LT 1 OR OMEGA GT 2

Meaning: SOR acceleration parameter OMEGA has an atypical value.

TOL1 IS NEGATIVE

Meaning: SOR parameter for changing OMEGA has an atypical value.

Examples V

Eight samples runs of BOAST are briefly described in this section. The corresponding input and output files are provided on the diskettes included with this manual.

19.	Sample Runs
Ex. 1	Linear Buckley–Leverett Waterflood Displacement
Ex. 2	Single Well Primary Depletion of Undersaturated Reservoir
Ex. 3	Waterflood Displacement with Multizone Completion
Ex. 4	Five–Spot Waterflood of the Mother Lode Sand
Ex. 5a	Faulted Reservoir with Multiple PVT and Rock Regions
Ex. 5b	Restart Run for Faulted Reservoir with . . . Regions (5a)
Ex. 6	Production from a Gas Reservoir with Aquifer Influx
Ex. 7	Production from an Oil Reservoir by Gas Injection

Sample Runs

Chapter 19

This chapter presents eight input sets describing a range of simulation problems. Following each data set are excerpts from the output of the run. These examples cover varied grid configurations and reservoir conditions. The following guide previews each example.

Ex. 1 **One–dimensional, linear model showing Buckley–Leverett waterflood displacement**

Block 10 × 1 contains an oil–rate specified production well. Block 1 × 1 contains a rate–specified water injection well.

Initial reservoir pressure	4787 psi
Initial oil saturation	0.80
Initial water saturation	0.20
Porosity	0.25
Permeability	200 md
Solution method	BAND
Maximum simulation time	360 days

Ex. 2 **Two–dimensional, areal model showing primary depletion of an undersaturated reservoir by a single production well**

Block 5 × 5 contains an oil–rate–specified production well.

Initial reservoir pressure	5383 psi
Initial oil saturation	0.88
Initial water saturation	0.12
Permeability	100 md
Solution method	D4
Maximum simulation time	90 days

Ex. 3 **Two–dimensional, X–Z cross–sectional model showing waterflooding of a layered system with multi–zone completions**

Block 10 × 1 contains an explicit–pressure–specified production well. Block 1 1 contains a rate–specified water injection well.

Initial oil saturation	0.80
Initial water saturation	0.20
Porosity	0.25
Permeability	200 md
Solution method	LSORZ
Maximum simulation time	300 days

Ex. 4 **Three–dimensional model of a heterogeneous 40–acre, five–spot pattern area showing primary depletion by fluid expansion and solution gas drive**

Primary depletion is followed by drilling of additional wells and waterflooding of the partially depleted reservoir. This input features several recurrent data sets.

Block 5 × 5 contains an oil–rate–specified production well. Rate–specified water injection wells are located in blocks 1 × 1, 1 × 9, 9 × 1, and 9 × 9.

Initial oil saturation	0.88
Initial water saturation	0.12
Solution method	LSORX
Maximum simulation time	1460 days

Ex. 5a **Three–dimensional model of a faulted reservoir with multiple rock and PVT types**

This run writes restart records and prints a summary table. Explicit–pressure–specified production wells are located in blocks 5 × 3, 5 × 7, 2 × 4 and 9 × 4.

Solution method	LSORX
Maximum simulation time	180 days

Ex. 5b **A restart of Ex. 5a continuing at 90 days**

The simulation continues from the 90–day mark from the 5a run. Maximum simulation time 365 days

Ex. 6 **Three–dimensional model of production from a gas reservoir with aquifer influx**

Block 2 × 8 contains a gas–rate–specified production well. Block 8 × 2 contains an explicit–pressure–specified gas production well.

Initial reservoir pressure	4787 psi
Initial oil saturation	0.88
Initial water saturation	0.12
Solution method	LSORX
Maximum simulation time	1095 days

Ex. 7 **Three–dimensional model of oil production by gas injection**

The repressurization algorithm is activated.

A rate–specified gas injection well is located in block 1 × 1, and block 10 × 10 contains an explicit–pressure–specified production well.

Initial oil saturation	0.88
Initial water saturation	0.12
Porosity	0.30
Solution method	LSORZ
Maximum simulation time	3650 days

Appendices VI

The Appendices provide supplementary information to the data input chapters. There are 14 Appendices (A–N):

A. Simulator Formulation
B. Choosing a Solution Method
C. Three–Phase Relative Permeability
D. PVT (Pressure Dependent) Data
E. Aquifer Models
F. Initialization
G. Well Models
H. Special Topics
I. Redimensioning Instructions
J. Laboratory Measurements of Saturation Dependent Data
K. Index of Program Variables
L. Sample Job Control Language
M Preprocessor Programs
N. Post Processor Programs

Simulator Formulation

Appendix A

This chapter states the flow equations that drive the simulation and discusses the following topics: IMPES iteration, the general solution procedure, the design of the model, and the concept of transmissibility.

For a more detailed discussion of BOAST's formulation, refer to Fanchi, J. R., K. J. Harpole, and S. W. Bujnowski. 1982. BOAST: a three-dimensional, three-phase black oil applied simulation tool. 2 vols. Bartlesville Energy Technology Center Report DOE/BC/10033-3.

Equations

BOAST describes reservoir fluids in three phases (oil, gas, and water) of constant composition, with physical properties that depend on pressure only. The mass conservation equations of the black oil simulator are written succinctly as follows:

FLOW EQUATIONS

OIL

$$-\bar{v}\frac{\bar{v}_o}{B_o} - \frac{q_o}{\rho_{osc}} = \frac{\delta}{\delta t}\left(\phi\frac{S_o}{B_o}\right)$$

WATER

$$-\bar{v}\frac{\bar{v}_w}{B_w} - \frac{q_w}{\rho_{wsc}} = \frac{\delta}{\delta t}\left(\phi\frac{S_w}{B_w}\right)$$

GAS

$$-\bar{v}\left[\frac{\bar{v}_g}{B_g} + \frac{R_{so}}{B_o}\bar{v}_o + \frac{R_{sw}}{B_w}\bar{v}_w\right] - \frac{q_g}{\rho_{gsc}} = \frac{\delta}{\delta t}\left\{\phi\left[\frac{S_g}{B_g} + \frac{Rso}{B_o}S_o + \frac{R_{sw}}{B_w}S_w\right]\right\}$$

The equations are interpreted using the list below when the subscript i indicates oil, gas, or water:

- B_i Formation volume factor of phase i.
- q_i Mass flow rate per unit reservoir volume of phase i.

- R_{so} Solubility of gas in oil.
- R_{sw} Solubility of gas in water.
- S_i Saturation of phase i.
- v_i Darcy velocity of phase i.
- ρ_{isc} Density of phase i at standard conditions.
- ϕ Porosity.

BOAST employs the following three equations, called auxiliary equations, to solve the preceding flow equations:

AUXILIARY EQUATIONS

$$S_o + S_w + S_g = 1.0 \qquad P_{cow}(S_w) = P_o - P_w$$

$$P_{cgo}(S_g) = P_g - P_o$$

When i represents oil, gas, or water, the variables in the auxiliary equations are interpreted as follows:

- S_i Saturation of phase i.
- P_i Pressure of phase i.
- P_{cow} Oil–water capillary pressure.
- P_{cgo} Gas–oil capillary pressure.

Stabilized IMPES

BOAST is an implicit pressure, explicit saturation (IMPES) numerical simulator. This means that BOAST implicitly solves for pressures throughout the grid, then incorporates these pressure values into equations to explicitly solve for saturation. The new pressure and saturation values are then considered to be the present values. Thus the solutions to the flow equations may be approximated for an arbitrarily long time.

As long as the simulation remains stable, the time–step size can be increased with a Runge–Kutta–based numerical method known as stabilized IMPES. The IMPES calculations occur as described below:

Determine the number of stabilized IMPES cycles K
Begin stabilized IMPES loop (1 to K)
    Evaluate properties
    Set up pressure matrix
    Solve for implicit pressures
    Solve for explicit saturations
    Form modified saturations
End stabilized IMPES loop

The modified saturations are given by

$$S^J_{mod} = S^o + C^J_K (S^J - S^o)$$

where

S^J = explicit saturations calculated on cycle J,
S^o = saturation at the beginning of the tie step, and
C^J_K = Runge–Kutta coefficients*.

On the last cycle, $C^K_K = 1$ and $S^K_{mod} = S^K$.

The number of cycles used (K) is calculated from the relation

$$K = 1 + \sqrt{(Q_{grid}/Q_{user})}$$

where normal FORTRAN integer rounding applies when

Q_{grid} = maximum throughput in a grid block over the preceding time step, and
Q_{user} = user specified maximum throughput.

If K > 7, the time–step size is reduced. If K = 1, the standard IMPES procedure is being used. The value of K is output in the time–step summary under ITERATION Output.

* Vinsome, P. K. W. Table 2. SPE Paper 5248.

General Solution Procedure

The progress of the simulation can be summarized as below:

Initialization stage

1. Read grid geometry.
2. Read porosity and permeability distributions.
3. Calculate geometric part of transmissibilities.
4. Read fluid properties and relative permeability data.
5. Initialize pressure and saturation distributions.
6. Specify solution method.

Recurrent stage

*1. Read time–step and well information.
*2. Calculate well rates for rate constraint option
*3. Calculate well rates for explicit pressure constraint option.
4. Calculate coefficients of pressure equation.
*5. Alter coefficients for implicit pressure constraint option.
6. Solve pressure equation.
*7. Calculate well rates for implicit pressure constraint option.
8. Solve for oil saturation S_o.
9. Solve for water saturation S_w.
10. $S_g = 1 - S_o - S_w$.
*11. Test automatic time step. Repeat time step if necessary.
*12. Calculate floating bubble point pressure.
*13. Write well report.
*14. Calculate material balance.
*15. Write summary report.
16. Update pressure and saturation arrays.
*17. Read time–step and well information.

* Performed according to user instructions.

Transmissibility

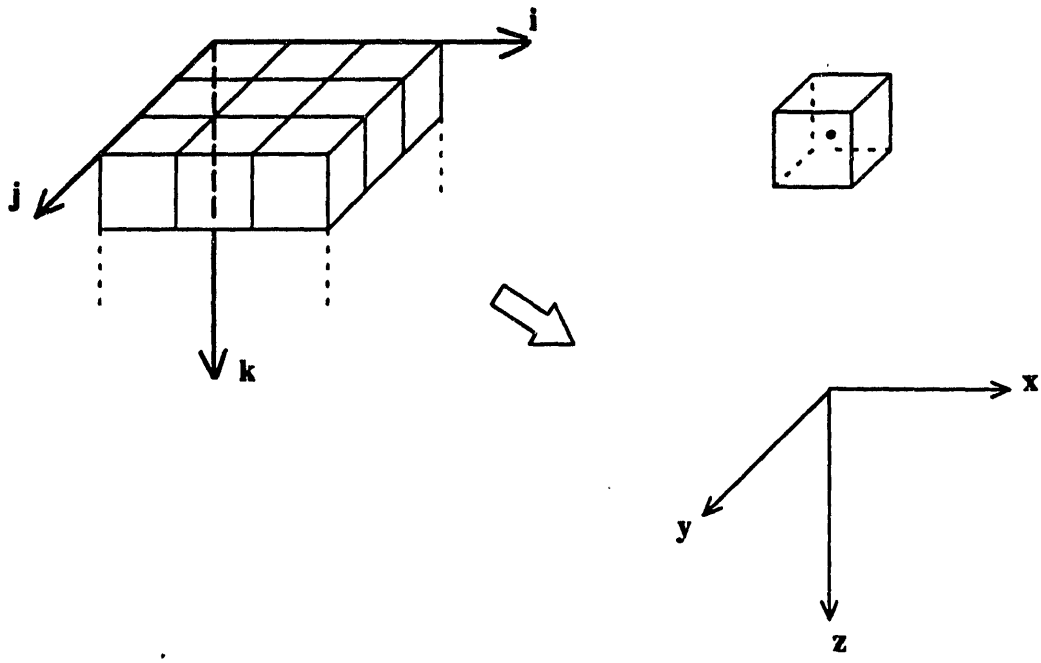
The simulator offers no–flow boundary conditions, allowing the user to prevent any flux between specified grid blocks in chosen directions. The user can then accurately model situations in which flow is not allowed equally in opposing directions. The no–flow conditions are implemented by setting transmissibilities at boundary interfaces to zero. The *Transmissibility Modifications* section in Chapter 4 describes the directional conventions for transmissibility in the model.

Flow between neighboring blocks is treated as a series application of Darcy's law. A transmissibility term is defined using average values of relative permeability times absolute permeability times cross–sectional area divided by the product of viscosity and formation volume factor. The transmissibility to each phase is determined using a harmonic average calculation of the product of absolute permeability and cross–sectional area at the interface between neighboring blocks. An arithmetic

averages of phase viscosities and formation volume factors is used. The average relative permeability is determined using an upstream weighted averaging technique.

Reservoir Model

The BOAST reservoir model assumes a block–centered grid with the axes aligned using a right–handed coordinate reference, as below. The top layer (K = 1) is shown. The second layer (K = 2) is under the K = 1 layer, and so on.



The following equations compare the work (measured by the number of multiplications and divisions performed) for the three algorithms per layer:

DIRECT	ITERATIVE
$W_{BAND} \propto I \times J^3$	$W_{LSOR} \propto I \times J \times N_{iter}$
$W_{D4} \propto \frac{I \times J^3}{2} - \frac{J^3}{4}$	

The equations measure the work but not the storage requirements of the algorithms. Storage quickly becomes a deciding factor as matrices get larger. There are no set rules or cutoffs for choosing a solution method. But, for example, a matrix as small as 10×10 would probably require an SOR algorithm.

Therefore, almost any three–dimensional problem should be solved by an iterative method. A three–dimensional matrix would have to be very small (a laboratory–scale model, for example) to be efficiently solved with a direct method.

The control for choosing a solution method is described in Chapter 11.

Three–Phase Relative Permeability Appendix C

NOTE: In principle, three–phase relative permeabilities should be used when oil, water, and gas are flowing simultaneously. However, using three–phase permeabilities is often impractical because they are difficult to measure accurately. It is commonly sufficient to work with two–phase relative permeabilities only.

Despite the shortcomings of three–phase relative permeabilities, a user may wish to perform a simulation using a set of three–phase curves. BOAST contains an option for computing a three–phase relative permeability curve using water–oil and gas–oil relative permeability curves.

This option was designed assuming that:

- the water relative permeability curve (k_{rw}) obtained for a water–oil system depends only on water saturation, and that
- the gas relative permeability curve (k_{rg}) obtained for a gas–oil system depends only on gas saturations.

The validity of these assumptions depends on such factors as wettability and degree of consolidation (see, for example, Schneider and Owens [1970]). Given the assumptions below, k_{rw} (for water–oil systems) and k_{rg} (for gas–oil systems) are also valid for a water–gas–oil system. The three–phase oil relative permeability k_{ro3} is calculated (Stone*, Model II of Dietrich and Bondor**) as follows:

$$k_{ro3} = \frac{(k_{row} + k_{rw})(k_{rog} + k_{rg})}{k^*_{row}} - (k_{rw} + k_{rg})$$

where

k_{row} = oil relative permeability for water–oil system,
k_{rog} = oil relative permeability for gas–oil system, and
k^*_{row} = oil relative permeability for water–oil system evaluated at the oil saturation corresponding to the irreducible water saturation.

When the three–phase calculation is activated, the user must be sure that the relative permeability curves entered are realistic. For example, the constraint

$$k_{row}(1-S_{wr}) = k_{rog}(S_o + S_w = 1.0)$$

must be satisfied because $S_g = 0$ in both cases.

The control for this option is on Line 4 of Chapter 6, Rock Data Tables.

* Stone, H. L. 1973. Estimation of three–phase relative permeability and residual oil data. J. Can. Pet. Tech. (October–December): 53ff.

* Dietrich, J. K., and P. L. Bondor. 1976. Three–phase oil relative permeability models. SPE Paper 6044, presented at 51st Fall Technical Conference and Exhibition of SPE of AIME, 3–6 October, New Orleans Louisiana.

Extrapolating Saturation Curves

Reservoir fluid properties (PVT data) include fluid viscosities, densities, formation volume factors, and gas solubilities. These data are usually obtained by laboratory analyses of fluid samples from reservoir.

Often the PVT data are not known over the wide range of pressures used in a simulation. However, the user can broaden the data base by complementing the laboratory data with correlations and by extrapolating the laboratory data.

Guidelines for extrapolating PVT data to pressures above the measured saturation pressure are presented below.

1. The B_g versus pressure curve is strongly non–linear, and an extrapolation of this curve to small B_g values at high pressures can result in errors. For most natural gases, the relationship $1/B_g$ versus pressure will be very nearly linear, especially at moderate to high pressures. Plotting $1/B_g$ versus pressure and extrapolating to PMAX should provide realistic values of B_g at higher pressures.

2. Once the B_g versus P curve is fixed, R_{so} versus P and B_o versus P curves must be extrapolated to avoid calculating a negative oil compressibility over any pressure increment. To ensure that the program will not calculate negative oil compressibilities, use the following test. For and pressure increment P_1 to P_2, where $P_2 > P_1$, the following relationship should hold:

$$0 \le B_{o2} - B_{o1} \quad \frac{B_{g2}(R_{so2} - R_{so1})}{5.615}$$

where the units of B_o, B_g, and R_{so} are RB/STB, RCF/STB, and SCF/STB, respectively.

NOTE: This test applies only to saturated oil PVT data.

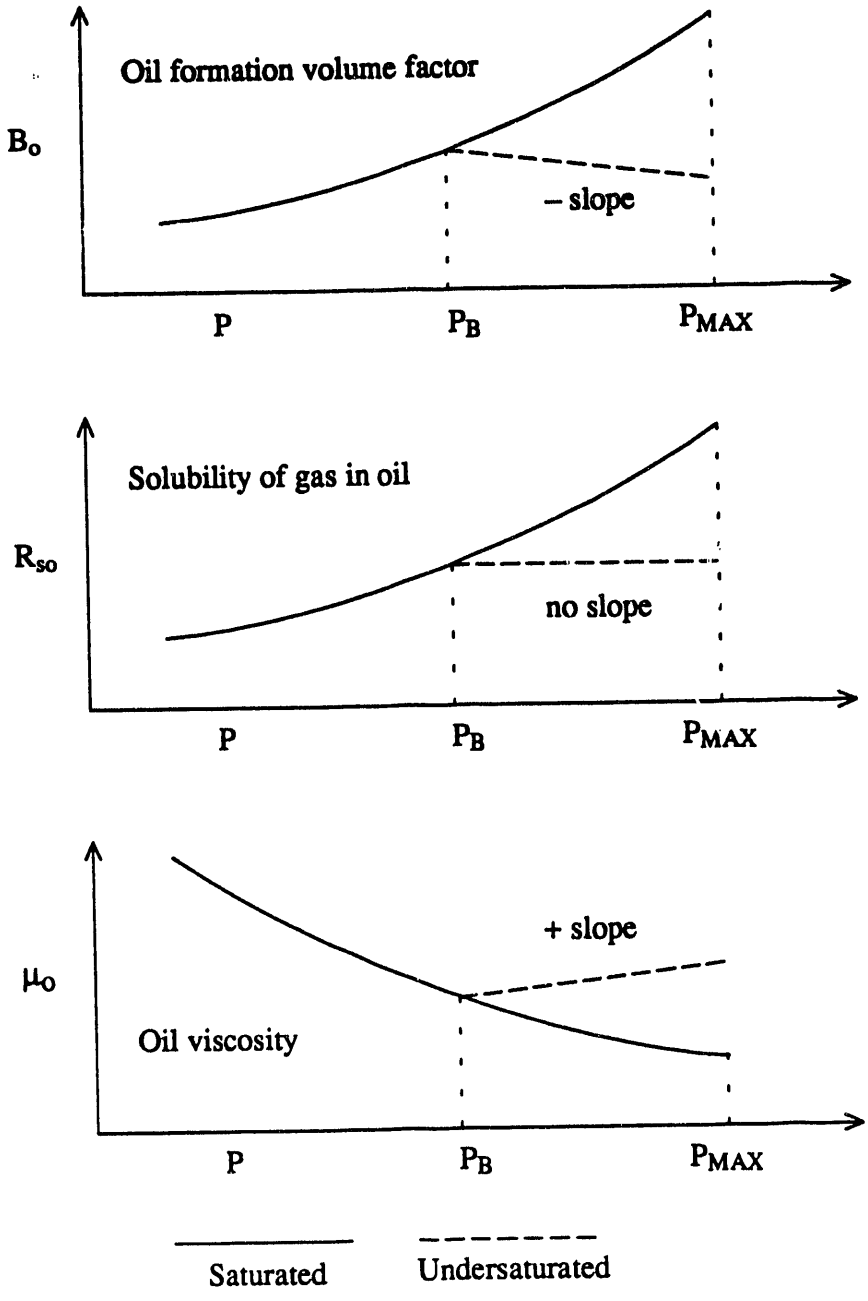
3. The above concepts also apply to the water PVT data. However, for most simulations, it can be assumed that $R_{sw} = 0.0$, thus, B_w reflects the effects of liquid compressibility and thermal expansion.

Bubble Point Tracking

When the pressure in the reservoir drops below the bubble point, a free gas saturation will form. If the free gas saturation exceeds the critical gas saturation, gas becomes mobile and can be produced. Production of the gas will change the local total gas/oil ratio, and, as a consequence, the local saturation pressure (bubble point) will change.

If the reservoir is repressured, for example, by waterflooding, the initial bubble point will no longer be correct. The following procedure, the BOAST repressurization option, accounts for this effect. Use this option carefully because the algorithm may cause discrepancies in the calculated reservoir pressures. The control for the option is the variable IREPRS on Line 4 of Chapter 7, Fluid Data Tables.

The following figure shows saturated curves as solid lines, and undersaturated curves, for bubble point P_B, as dashed lines. The three oil properties, oil formation volume factor B_o, solubility of gas in oil R_{so}, and oil viscosity μ_o, are functions of pressure. As the reservoir is depleted, the value for each oil property moves along the appropriate curve in the direction of decreasing pressure. The user specifies the saturated curve as well as the slopes of the straight lines representing the undersaturated curves.



The purpose of the repressurization option is to estimate the pressure at which free gas will completely dissolve in the saturated oil in the model grid. An estimation for the new gas solubility (from the volumes of gas and oil in the block) allows the use of the saturated R_{so} versus P curves to find a new bubble point. The new solubility is estimated to be the sum of the dissolved gas and the free gas in the block, divided by the stock tank oil in the block, as shown below.

$$R_{so}^{new} = \frac{R_{so} + S_g B_o}{S_o B_g} = \text{estimate of new gas solubility}$$

The new bubble point pressure P_B^{new} is the saturation pressure that corresponds to R_{so}^{new}. Undersaturated curves for pressures above the new bubble point pressure are parallel to the initial undersaturated curves.

Gas Correlation Option

The control for this option is on Line 10 of Chapter 7, Fluid Data Tables.

Aquifer Models

Appendix E

BOAST offers three analytic models to represent aquifer influx: the pot aquifer model, the steady–state aquifer model, and the Carter–Tracy aquifer model. Analytic models require less computer storage and computing time than models that use grid blocks to define the aquifer.

Each section below states what conditions are assumed when the aquifer is activated and presents the equations on which the aquifer is based. See Chapter 12.

Pot Aquifer

Aquifer influx is calculated assuming the aquifer is both small and bounded. The pot aquifer influx rate q_{wp} is dependent on the pressure change over a time step for a specified grid block:

$$q_{wp} = -\left[POT \frac{(P^n - P^{n+1})}{\Delta t} \right]; POT \geq 0 \qquad (E.1)$$

where P^n, Δt^n are grid block pressure and time step at the present time level n; P^{n+1}, Δt^{n+1} are grid block pressure and time step at the future time level n+1; and POT is the pot aquifer coefficient. The minus sign preceding the bracketed term indicates water is entering the block when $P^n > P^{n+1}$.

Steady–State Aquifer

The steady–state aquifer model is based on Schilthuis' assumption that the water influx rate q_{wss} is proportional to the pressure difference between the aquifer and the hydrocarbon reservoir. It is further assumed that the aquifer is sufficiently large that it experiences no net pressure change throughout the producing life of the reservoir. With these assumptions we compute steady–state aquifer influx into a specified grid block as

$$q_{wss} = -\left[SSAQ\,(P^o - P^{n+1}) \right]; SSAQ \geq 0 \qquad (E.2)$$

where P^{n+1} is the grid block pressure at the future time level n+1; P^o is the initial grid block pressure; and SSAQ is the proportionality constant. The minus sign preceeding the bracketed term indicates water is entering the block when $P^o > P^{n+1}$.

Carter–Tracy Aquifer

The Carter–Tracy[1] modification of the Hurst–van Everdingen[2] unsteady–state aquifer influx calculation is available in BOAST II. The Carter–Tracy aquifer influx rate q_{wct} for a specified grid block is

$$q_{wct} = -\left[A - B \left(P^{n+1} - P^{n} \right) \right] \qquad \text{(E.3)}$$

where P^{n} , P^{n+1}, are grid block pressures at time levels n and n+1, respectively. The coefficients A and B are given by

$$A = K_t \left[\frac{\beta(P^{o} - P^{n}) - W_e {}^{n}P'^{n+1}_{id}}{DENOM} \right] \qquad \text{(E.4)}$$

$$B = K_t \beta / DENOM \qquad \text{(E.5)}$$

with

$$DENOM = P^{n+1}_{tD} - t_D {}^{n}P'^{n+1}_{tD} \qquad \text{(E.6)}$$

$$P'^{n+1}_{tD} = \left[\frac{dP_{tD}}{dt_D} \right]^{n+1} \qquad \text{(E.7)}$$

$$K_t = 0.00633\ k/(\phi \mu c r_e^2) = \text{AQ PAR 1} \qquad \text{(E.8)}$$

$$\beta = 2\ \pi \phi h c r_e^2 s = \text{AQPAR2} \qquad \text{(E.9)}$$

1. Carter, R. D. and Tracy, G. W. (1960); "An Improved Method for Calculating Water Influx," Trans. of AIME, Vol. 219, pp. 415–417.

2. van Everdingen, A. F. and Hurst, W. (1949); "The Application of the Laplace Transformation to Flow Problems in Reservoirs," Trans. of AIME, Vol. 186, pp. 305–324.

and

$$c = c_r + c_w \qquad \text{(E.10)}$$

The quantities t_D and P_{tD} are dimensionless time and pressure, respectively, with

$$t_D = K_t t$$

and P_{tD} is the Carter–Tracy influence function for the constant terminal rate case. The functions P_{tD} and P'_{tD} are numerically represented by the regression equations shown in the Table E–1. All remaining parameters are defined as follows:

c_r = rock compressibility, psi^{-1}

c_w = water compresibility, psi^{-1}

h = aquifer net thickness, ft

k = aquifer permeability, md

r_e = external aquifer radius, ft

r_w = external reservoir radius, ft

s = $\theta/360^o$ where θ is the angle of aquifer/reservoir interfase

W_e^n = cumulative water influx at time level n, SCF

μ = aquifer water viscosity, cp

ϕ = aquifer porosity

TABLE E–1: Carter–Tracy Influence Function Regression Coefficients for the Constant Terminal Rate Case

Regression Equation:

$$P_{tD} = a_o + a_1 t_D + a_2 \ln t_D + a_3 (\ln t_D)^2$$

$$P'_{tD} = a_1 + \frac{a_2}{t_D} + 2a_3 \frac{\ln t_D}{t_D}$$

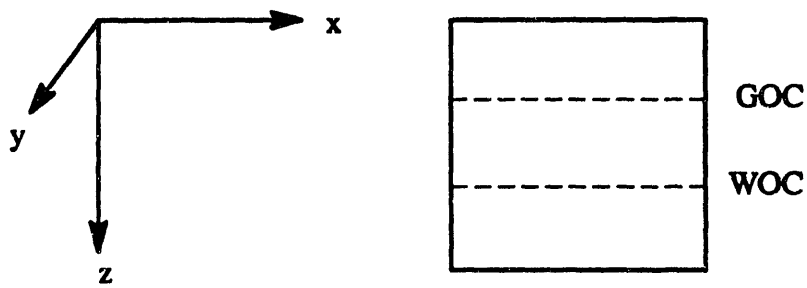
Case	Regression Coefficients				
r_e/r_w	a_0	a_1	a_2	a_3	Correlation range of t_D
1.5	0.10371	1.66657	–0.04579	–0.01023	0.06–0.6
2.0	0.30210	0.68178	–0.01599	–0.01356	0.22–5.0
3.0	0.51243	0.29317	0.01534	–0.06732	0.52–5.0
4.0	0.63656	0.16101	0.15812	–0.09104	1.5–10.0
5.0	0.65106	0.10414	0.30953	–0.11258	3.0–15.0
6.0	0.63367	0.06940	0.41750	–0.11137	4.0–30.0
8.0	0.40132	0.04104	0.69592	–0.14350	8.0–45.0
10.0	0.14386	0.02649	0.89646	–0.15502	12.0–70.0
∞	0.82092	–0.000368	0.28908	0.28817	0.01–1,000.0

Pressure Initialization

It is important when making cross–section or 3–D runs that the pressures in the model are correctly initialized. If not, phase potential differences due to gravity terms could cause fluid migration even though no wells are active. Consequently, a simple pressure initialization algorithm is used in BOAST II.

Consider a grid block which may have a gas–oil contact and a water–oil contact as in Figure F–1.

Figure F–1. Grid Block i,j,k, for Pressure Initialization.



We assume the pressure in the block is dominated by the density of the phase at the block midpoint and that there are no transition zones between different phases initially. The pressure and depth at the gas–oil contact are PGOC and GOC, respectively. Similarly, for the water–oil contact we have PWOC and WOC.

The initial pressure assigned to the grid block in Figure F–1 is determined by the depth of the node (midpoint) relative to the respective contact elevations.

Let us define the depth of the block midpoint from datum as EL_{ijk}. With this definition, the pressure in the block is given by the following algorithm.

a. If $EL_{ijk} < GOC$ then $\rho_g = \rho_{gsc}/B_g$

$$\text{and } P_{ijk} = PGOC + \rho_g (EL_{ijk} - GOC)/144 \qquad (F.1)$$

b. If $EL_{ijk} > WOC$ then $\rho_w = \frac{1}{B_w}(\rho_{wsc} + R_{sw} \cdot \rho_{gsc})$

$$\text{and } P_{ijk} = PWOC + \rho_w (EL_{ijk} - WOC)/144 \qquad (F.2)$$

C. If $GOC \leq EL_{ijk} \leq WOC$ then

$$\rho_o = \frac{1}{B_o}(\rho_{osc} + R_{so} \cdot \rho_{gsc}) \qquad \text{(F.3)}$$

and $P_{ijk} = PWOC + \rho_o (EL_{ijk} - WOC)/144$

The above algorithm should be reasonable for systems with initial transition zones that are small relative to the total thickness of the formation.

Saturation Initialization

Initial oil, water, and gas saturations may be specified under option KSI=0 as SOI, SWI, and SGI respectively. Given the initial saturations, the following constraints apply:

a. If $EL_{ijk} < GOC$; then

$SO_{ijk} = 0.0$

$SW_{ijk} = SWR$

$SG_{ijk} = 1 - SW_{ijk}$

where SWR is the irreducible water saturation for the rock region containing grid block ijk.

b. If $EL_{ijk} > WOC$; then

$SO_{ijk} = 0.0$

$SW_{ijk} = 1.0$

$SG_{ijk} = 0.0$

c. If $GOC \leq EL_{ijk} \leq WOC$; then

$SO_{ijk} = SOI$

$SW_{ijk} = SWI$

$SG_{ijk} = SGI$

where

$$SGI = 1 - SOI - SWI \qquad (F.4)$$

Output of Pressures Corrected to Datum

Pressure P(I,J,K) of grid block I, J, K with mid–point elevation EL(I,J,K) may be corrected to a datum depth PDATUM by specifying a pressure gradient GRAD. The pressure at datum is given by

$$PDAT(I,J,K) = P(I,J,K) + (PDATUM - EL(I,J,K))*GRAD \qquad (F.5)$$

Rate Constraint Representation

Case 1: Oil Production Rate Q_o Specified

In this representation, rates may be specified for injectors or producers. Assuming the well may be completed in K layers, as we will throughout, the production rates of layer k for a specified oil rate are:

Oil

$$Q_{ok} = Q_o \frac{[(PI)\lambda_o/B_o]_k}{\sum_{k=1}^{K} [(PI)\mu_o/B_o]_k} \qquad (G.1)$$

Water

$$Q_{wk} = Q_{ok}\left(\frac{\lambda_w/B_w}{\lambda_o/B_o}\right)_k \qquad (G.2)$$

and Gas

$$Q_{gk} = \left(\frac{\lambda_g/B_g}{\lambda_o/B_o}\right)_k Q_{ok} + (R_{so})_k Q_{ok} + (R_{sw})_k Q_{wk} \qquad (G.3)$$

Notice that PI's may be specified by layer. This capability lets the BOAST II user take into account permeability contrast.

Case 2: Water Production Rate Q_w Specified

Assuming the well may be completed in K layers, the production rates of layer k for a specified water rate are:

Water

$$Q_{wk} = Q_w \frac{[(PI)\lambda_w/B_w]_k}{\sum_{k=1}^{K}[(PI)\lambda_w/B_w]_k} \qquad (G.4)$$

Oil

$$Q_{ok} = Q_{wk}\left(\frac{\lambda_o/B_o}{\lambda_w/B_w}\right)_k \qquad (G.5)$$

and Gas

$$Q_{gk} = \left(\frac{\lambda_g/B_g}{\lambda_w/B_w}\right)_k Q_{wk} + (R_{sw})_k Q_{wk} + (R_{so})_k Q_{so} \qquad (G.6)$$

Case 3: Gas Production Rate Q_g Specified

Assuming the well may be completed in K layers, the production rates of layer k for a specified gas rate are:

Gas

$$Q_{gk} = Q_g \frac{[(PI)\lambda_g/B_g]_k}{\sum_{k=1}^{k}[(PI)\lambda_g/B_g]_k} \qquad (G.7)$$

Oil

$$Q_{ok} = Q_{gk} \left(\frac{\lambda_o/B_o}{\lambda_g/B_g} \right)_k \qquad \text{(G.8)}$$

Water

$$Q_{wk} = Q_{gk} \left(\frac{\lambda_w/B_w}{\lambda_g/B_g} \right)_k \qquad \text{(G.9)}$$

Notice that solution gas in both oil and water is neglected when a gas production rate is specified. This will be a reasonable assumption for wells producing primarily free gas.

Case 4: Total Production Rate Specified

When the total reservoir voidage rate Q_T is specified, we first compute the phase mobility ratio for all layers:

Oil Mobility Ratio

$$\alpha_{oT} = \sum_{k=1}^{K} \left(\frac{\lambda_o}{\lambda_o + \lambda_w + \lambda_g} \right)_k \qquad \text{(G.10)}$$

Water Mobility Ratio

$$\alpha_{wT} = \sum_{k=1}^{K} \left(\frac{\lambda_w}{\lambda_o + \lambda_w + \lambda_g} \right)_k \qquad \text{(G.11)}$$

and Gas Mobility Ratio

$$\alpha_{gT} = \sum_{k=1}^{K} \left(\frac{\lambda_g}{\lambda_o + \lambda_w + \lambda_g} \right)_k \qquad \text{(G.12)}$$

We now compute the total oil rate

$$Q_o = \left(\frac{\alpha_{oT}}{\alpha_{oT} + \alpha_{wT} + \alpha_{gT}} \right) \frac{Q_T}{\bar{B}_o} \qquad \text{(G.13)}$$

where

$$\bar{B}_o = \frac{1}{K} \sum_{k=1}^{K} (B_o)_k \qquad \text{(G.14)}$$

is the average oil formation volume factor for all layers in which the well is completed.

Given Equation (G.13), we simply proceed as in Equations (G.1) through (G.3) above.

Case 5: Injection Rate Specified

If the well is a water or gas injector, the user must specify the total water or gas injection rates Q_w or Q_g, respectively, and well injectivity indexes (WI) for each layer. The injection rate for each layer is then allocated as follows:

Water Injection Rate

$$Q_{wk} = Q_w \frac{\left[WI(\lambda_o + \lambda_w + \lambda_g) \right]_k}{\sum_{k=1}^{K} \left[WI(\lambda_o + \lambda_w + \lambda_g) \right]_k} \qquad \text{(G.15)}$$

Gas Injection Rate

$$Q_{gk} = Q_g \frac{[WI(\lambda_o + \lambda_w + \lambda_g)]_k}{\sum_{k=1}^{K} [WI(\lambda_o + \lambda_w + \lambda_g)]_k} \qquad (G.16)$$

It is important to note that allocation of injection fluids is based on total mobilities, and not just injected fluid mobility. This is necessary for the following reason.

If an injector is placed in a block where the relative permeability to the injection fluid is zero, then the simulator using injection fluid mobility only would prohibit fluid injection even though a real well would allow fluid injection. A common example would be water injection into a block containing oil and irreducible water. To avoid the unrealistic result of no fluid injection, it is assumed the total mobility of the block should be used. For most cases, the error of this method will only persist for a few time steps because, in time, the mobile fluid saturation in the block will be dominated by the injected fluid.

NOTE: A rate constrained oil production well can also be used as an injector by making the oil rate negative. In this case, BOAST II uses total fluid mobility as in Equations (G.15) and (G.16). Furthermore, the injected fluid rate applies only to oil (Equations (G.2) and (G.3) are skipped when Q_{ok} is negative). This option is useful if the BOAST II user is interested in tracking three injected fluids.

Implicit Pressure Constraint Representation

The source/sink terms in the black oil simulator fluid flow equations (Appendix A) may be written as

$$Q_{pk} = \left[PID\left(\frac{\lambda_p}{B_p}\right)\right]_k^n (P^{n+1} - PWF) \qquad (G.17)$$

where the subscript p signifies the appropriate oil, water, or gas phase. If the well is a producer, PID = PI and $P^{n+1} > PWF$ where, PWF is the well flowing pressure P_{wf}. If the well is an injector, PID = WI and $P^{n+1} < PWF$.

Substituting Equation (G.17) into the fluid flow equations, we can implicitly solve for pressure. The computed pressure P^{n+1} is then replaced in Equatipn (G.17) to yield rates.

Explicit Pressure Constraint Representation

The most commonly used method of calculating rates when wells are under pressure constraint is based on the method described below:

Case 1: Oil and/or Water Production Wells

We assume that flowing bottomhole pressures (PWF) and well PI's are specified for a pressure constrained well. The oil and water rates in STB/D for layer k are given by

$$Q_{ok} = \left[PID \frac{\lambda_o}{B_o} \right]_k^n (P^n - PWF)_k \qquad (G.18)$$

and

$$Q_{wk} = \left[PID \frac{\lambda_w}{B_w} \right]_k^n (P^n - PWF)_k \qquad (G.19)$$

where PID = PI and the explicit pressure P^n is used. If $P^n < PWF$, the well is shut in. When $P^n > PWF$, Q_{ok} and Q_{wk} are calculated and then substituted into Equation (G.3) to find Q_{gk}.

Case 2: Gas Production Well

The laminar–inertial–turbulent (LIT) method may be used to represent a gas production well. The LIT method entails fitting gas well test data to the equation

$$\Delta\psi = aQ_g + bQ_g^2 = \psi_R - \psi_{wf} \qquad (G.20)$$

where

ψ_R = pseudo–pressure corresponding to shut–in reference pressure P_R, $psia^2/cp$

Ψ_{wf} = pseudo–pressure corresponding to a user–specified well flowing pressure P_{wf}, $psia^2/cp$

aQ_g = term characterizing laminar flow and well conditions

and

bQ_g^2 = term characterizing inertial and turbulent flow.

BOAST II employs user–specified values of a, b, P_{wf} and pseudo–pressure vs. pressure to compute the total gas well production rate as

$$Q_g = \frac{-a + \sqrt{a^2 + 4b\Delta\psi}}{2b} \qquad \text{(G.21)}$$

where ψ_R is the pseudo–pressure corresponding to the nodal pressure P^n. Rates for each phase in layer k are computed by mobility allocation as shown in Eqs. (G.7) through (G.9).

Case 3: Injection Wells

The injection rate for a water or gas injection well is computed from

$$Q_{pk} = \left[PID\left(\frac{\lambda_o + \lambda_w + \lambda_g}{B_p}\right) \right]_k^n (P^n - PWF)_k \qquad (G.22)$$

where the subscript p denotes water or gas, and PID=WI. Fluid injection occurs when $P^n < PWF$. If $P^n > PWF$, the injection well is shut in. Also note that total mobility is used for the injection well rate calculation. The reason for this was discussed under Case 5 of the Rate Constraint Representation section.

Estimating Layer Flow Index (PID)

A value of the layer flow index PID can be estimated from a formula derived by Peaceman[3]

$$PID_k = \left[\frac{0.00708\ Kh}{\ln\left(\frac{0.121\Delta L}{r_w}\right) + S} \right]_k \qquad (G.23)$$

where $\Delta L = \Delta x$ for a square well block, and $\Delta L = \sqrt{\Delta x \Delta y}$ for a rectangular well block. The subscript k in Equation (G.23) denotes the k^{th} layer. The remaining parameters are defined as:

K = air permeability of layer k, md,

h = thickness of layer k, feet,

r_w = wellbore radius, feet,

and S = dimensionless skin factor.

3. Peaceman, D. W. (1978); "Interpretation of Well-Block Pressures in Numerical Reservoir Simulation," Soc. Pet. Eng. J., pp. 183–194. See also Peaceman, D.W. (1983); "Interpretation of Well-Blocks Pressures in Numerical Reservoir Simulation with Nonsquare Grid Blocks and Anisotropic Permeability," Soc. Pet. Eng. J., pp. 531–543.

In principle, the layer flow index could be related to measured values. In practice, however, the terms r_e, S, and $k_{ro}/\mu_o B_o$ are seldom well known, especially for a multiphase flowing well. As a matter of expediency, therefore, Equation (G.23) is often used to compute an initial estimate of PID. This value can then be improved by adjusting it until the simulator computed well rates match the initial observed well rates.

GOR/WOR Constraints

Maximum gas/oil and water/oil ratios (GORMAX, WORMAX respectively) are input by the user and apply to every oil production well. GOR for a well is defined as total gas production divided by total oil production for all active well completion intervals. If GOR for the well exceeds GORMAX, then the completion interval (layer) with the highest GOR will be shut–in. If more than one layer has the same maximum GOR, the shallowest layer will be shut–in first. The procedure is repeated until GOR is less than GORMAX or until the well is shut–in.

The ratio WOR is defined as total water production divided by total oil production for all active well completion intervals. If WOR for the well exceeds WORMAX, then the completion interval (layer) with the highest WOR will be shut–in. If more than one layer has the same maximum WOR, the deepest layer will be shut–in first. The procedure is repeated until WOR is less that WORMAX or until the well is shut–in.

Fluid Withdrawal Constraints

Fluid withdrawal from explicit pressure controlled production wells can be constrained as follows:

a. A minimum oil production rate can be specified;
b. A maximum oil production rate can be specified; and
c. A maximum liquid (water plus oil) withdrawal rate can be specified.

A positive value of QO for a pressure controlled production well is used as the minimum allowed oil production rate. If the calculated oil production rate drops below the minimum allowed value, the well is shut–in.

A positive value of QW for a pressure controlled production well is used as the maximum allowed oil production rate. If the calculated oil production rate exceeds the maximum allowed value, calculated production will be reduced to the allowed value. Production from each layer is proportionally reduced by the ratio of allowed to calculated oil production rates.

A positive value of QT for a pressure controlled production well is used as the maximum allowed liquid withdrawal rate. If the sum of oil and water production exceeds the maximum allowed value, calculated production is reduced to the allowed value. The reduction is made by multiplying production from each layer by the ratio of allowed to calculated liquid withdrawal rates. IMPORTANT: The units of QT when used to control total liquid withdrawal are STB/Day.

Fluid Injection Constraints

Fluid injection into explicit pressure controlled injection wells can be constrained as follows:

a. A maximum water injection rate can be specified; and
b. A maximum gas injection rate can be specified.

A negative value of QW for a pressure controlled water injection well is used as the maximum allowed water injection rate. If the calculated water injection rate exceeds the allowed value, calculated water injection will be reduced to the allowed value. Water injection into each layer is proportionally reduced by the ratio of allowed to calculated water injection rates.

A negative value of QG for a pressure controlled gas injection well is used as the maximum allowed gas injection rate in direct analogy to the water injection rate constraint described above.

Special Topics Appendix H

This chapter covers three topics: Automatic Time–Step Control, Material Balance Check, and Numerical Dispersion.

Automatic Time–Step Control

The user has an option regarding the time–step size. BOAST offers either a fixed–size time step or automatic time–step control (ATSC). Under ATSC, the simulator increases the the time–step size when conditions are changing slowly, or decreases the size for accuracy when conditions are changing rapidly.

Specifically, BOAST determines when to increase or decrease step size according to the maximum changes in pressure (DPMC) and saturation (DSMC) throughout the grid. BOAST compares these changes with the user–specified maximum acceptable changes in pressure, DPMAX, and saturation, DSMAX, (see Chapter 11, Line 2). If either of these limits is exceeded, the time–step size is reduced by FACT2, and the time step is repeated. On the other hand, if both DPMC and DSMC are less than their respective maximum acceptable values, the time–step size is increased by FACT1.

The user also sets a minimum and a maximum time step (Chapter 13, Line 6, DTMIN and DTMAX). ATSC will modify the size of the time step only within these boundaries. The initial value for the time step (DT) is also given at this point.

The controls for ATSC are discussed in Chapter 10, Run Control Parameters. The variables FACT1 and FACT2 activate ATSC. FACT1 is the factor for *increasing* the time step under ATSC. FACT2 is the factor for *decreasing* the time step under ATSC. Commonly, FACT1 = 1.25 and FACT2 = 0.5. For fixed–size time steps, set both FACT1 and FACT2 = 1.0.

Chapter 10 warns not to set only one of the variables = 1.0. If the program were allowed to increase (but not decrease) the time–step size, the simulation could proceed quickly but inaccurately. On the other hand, if the simulation were allowed only to decrease the time–step size, the program would be far to slow.

NOTE: If FACT2 = 1.0, the step size will not decrease, nor will BOAST repeat a time step. The user should monitor the material balance calculations to check that the run has maintained stability when FACT2 = 1.0.

Material Balance Calculation

BOAST performs a simple, instantaneous material balance calculation so that the user may check the accuracy of the finite–difference calculations. The simulator compares the volume–in–place of oil, gas, and water at the beginning and at the end of the time step, taking into consideration injected and produced fluids. The percent change over the time step for each phase is reported in both the Time–Step Summary and the Summary Report. See Chapter 13, Time–Step and Output Control.

Numerical Dispersion

BOAST II, like most simulators in use today, solves the black oil fluid flow equations by replacing derivatives with finite difference approximations. The truncation error

introduced by the approximation is small for many systems of practical interest and the approximate solutions of the finite difference equations are accurate enough for engineering purposes. For some systems, however, the truncation error cannot be ignored. Such systems are convection–dominated with sharp displacement fronts. Examples include miscible floods and immiscible floods in which the ratio of capillary to viscous forces is small.

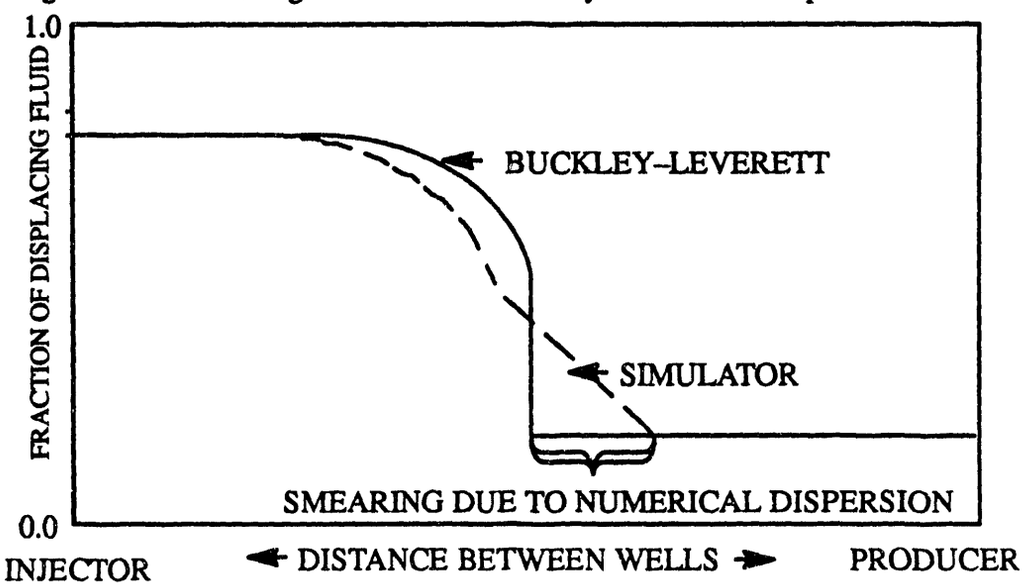
A donimant characteristic of numerical truncation error is the smearing of otherwise sharp saturation fronts. This is readily demonstrated when a simulator is used to model the Buckley–Leverett problem. Illustrative results are depicted in Figure H–1. The truncation error makes the saturation front appear more dispersed than it should, hence the truncation error is known as numerical dispersion or numerical diffusion.

Lantz[4] showed that the total dispersion D^{tot} in a one–dimensional simulation is the sum of physical dispersion and numerical dispersion:

$$D^{tot} = D^{phy} + D^{num}. \qquad \text{(H.1)}$$

An extension of Lantz's method to three dimensions by Fanchi[5] demonstrated the validity of Equation (H.1) for multidimensions, and also showed multidimensional numerical dispersion (MND) can cause rotation of the principle flow axes for a given system. Fanchi's paper contains a method for estimating the magnitude of the rotation effect. Computation of the magnitude of the front smearing and rotation effects can help the user of a finite difference simulator evaluate the validity of the simulation.

Figure H–1. Smearing of Saturation Fronts by Numerical Dispersion



4. Lantz, R. B. (August 1971); "Quantitative Evaluation of Numerical Diffusion," Soc. Pet. Eng. J., pp. 269–276.
5. Fanchi, J. R. (February 1983); "Multidimensional Numerical Dispersion," Soc. Pet. eng. J., pp. 143–151.

Redimensioning Instructions Appendix I

The arrays in BOAST II can be redimensioned to the most efficient size for a given problem to minimize computer storage costs. Redimensioning was made relatively easy by defining a file containing all of the PARAMETER statements, which is included in the source file using the INCLUDE statement. To redimension BOAST II, simply change the user-specified parameters, defined on the following page, is the PARAMETER file to the appropriate values. After editing the PARAMETER file, delete the previous object code and recompile the BOAST II program.

The form of the INCLUDE statement in the FORTRAN source code, Appendix K, as well as that of the PARAMETER statement in the PARAMETER file, is installation dependent. The form of these statements may need to be changed according to the installation specifications. If there is not an equivalent to the INCLUDE statement, the PARAMETER statements from the PARAMETER file should replace the INCLUDE statement in the main program and in each subroutine. In this case, to redimension BOAST II, the appropriate PARAMETER statements in each subroutine must be edited and the source code recompiled.

If there is not an equivalent to the PARAMETER statement, all dimensioning must be done using the DIMENSION and COMMON statements. In the main program, and in each subroutine, the INCLUDE statement must be deleted. All of the variables (LP1–LP23) defined in the PARAMETER statements and used to dimension variables in COMMON and DIMENSION statements must be changed in all COMMON and DIMENSION statements in the program to the specific values desired for the problem at hand before compiling the BOAST II program.

LP1	NX	Max X-direction grid blocks
LP2	NY	Max Y-direction grid blocks
LP3	NZ	Max Z-direction grid blocks
LP7	NROCK	Max ROCK regions
LP8	NPVT	Max PVT regions
LP9	NTE	Max number of input data table entries per region (applies to both ROCK and PVT regions)

LP10	NRPMAX	Larger of NROCK and NPVT
LP11	NW	Max number of allowed wells
LP12	ITSMAX	Max number of time steps included in post–plot options
LP14	TOT2D	Max blocks using 2D or 3D direct solution methods
LP15	TOT1D	Max blocks using 1D direct solution methods
LP16	MAX2D	Max blocks in 2D direct solution calculation of L2SOR subroutine
LP17	NRST	Max number of allowed restart records
LP18		= 0.5 * LP14; half of TOT2D.

DERIVED PARAMETERS:

LP4	NXP	= NX + 1
LP5	NYP	=NY + 1
LP6	NZP	= NZ + 1
LP13		= NX + NY + NZ
LP19	NXMAX	=NXP * NY * NZ
LP20	NYMAX	=NX * NYP * NZ
LP21	NZMAX	= NX * NY * NZP
LP22	NMAX	= NX * NY * NZ
LP23		= NW * NZ

Fluid Injection Constraints

The most critical of these parameters are LP1, LP2, LP3, LP14, LP16, and LP18. For efficient dimensioning, these parameters should be set to the specifid dimensions

of the problem at hand. When the solution method (KSOL) being used is 1, 3, 4, or 5, LP14, LP16, and LP18 can be set to one. Otherwise, they should be dimensioned as follows:

LP14 = NX * NY * NZ

LP16 = NX * NY * NZ

LP18 = 0.5 (NX * NY * NZ)

Array Dimensions

The dimensions of the arrays in BOAST II are summarized below. The indices used for array dimensions are defined as follows:

NX	=	Max X–direction grid blocks
NY	=	Max Y–direction grid blocks
NZ	=	Max Z–direction grid blocks
NMAX	=	NX * NY * NZ
NROCK	=	Max ROCK regions
NPVT	=	Max PVT regions
NRPMAX	=	Larger of NRock and NPVT
NTE	=	Max no. of input data table entries per region
NXP	=	NX + 1
NYP	=	NY + 1
NZP	=	NZ + 1
NXMAX	=	NXP * NY * NZ
NYMAX	=	NX * NYP * NZ
NZMAX	=	NX * NY * NZP
NW	=	Max no. of allowed wells
ITSMAX	=	Max no. of time steps included in post–plot options
TOT1D	=	Total blocks using 1D direct solution methods
TOT2D	=	Total blocks using 2D or 3D direct solution methods
MAX1D	=	Max no. of blocks in 1D for LSOR
MAX2D	=	Max no. of blocks in 2D for L2SOR
NRST	=	Max no. of allowed restart records

COMMON BLOCK ARRAYS

COMMON/BUBBLE/

VSLOPE(NPVT)
BSLOPE(NPVT)
RSLOPE(NPVT)
RHOSCO(NPVT)
RHOSCW(NPVT)
RHOSCG(NPVT)
MSAT(NROCK)
MPOT(NPVT)
MPGT(NPVT)
MPWT(NPVT)
PBOT(NX,NY,NZ)

COMMON/COEF/

AW(NX,NY,NZ)
AE(NX,NY,NZ)
AN(NX,NY,NZ)
AS(NX,NY,NZ)
AB(NX,NY,NZ)
AT(NX,NY,NZ)
E(NX,NY,NZ)
B(NX,NY,NZ)

Laboratory Measurements of Saturation–Dependent Data

Appendix J

Relative Permeability Tests

General Comments:

Relative permeability tests can be conducted on "native–state" cores, "fresh–state" cores, and "restored–state" cores. Native–state cores are those drilled with non–aqueous coring fluids (lease crude, No. 1 diesel oil, oil–base mud, gas, etc). These cores are normally tested without cleaning, as are fresh–state cores (those drilled with water–base fluids). Restored–state cores have been previously cleaned, and simulated connate water saturations must be established prior to testing. Native–state cores are preferred for relative permeability testing because the coring procedure preserves the reservoir wettability and connate water saturations. With regard to core wettability, bland brines and refined oils are employed to avoid alterations in wettability.

Core preparation procedures are as follows. Native–state and fresh samples are flushed at reservoir temperature to remove crude oil, gas, mobile water, and any precipitated waxes and asphaltenes. This treatment leaves a test sample saturated with oil and a near–irreducible water saturation. Restored–state samples, clean and dry, are saturated with brine and flushed with a 25 cp oil under a high pressure differential to produce a near–irreducible water saturation. For unsteady–state relative permeability tests GO, GWO, and WO (pages J.4 and J.5), the 25 cp oil is displaced by injecting either introgen or brine. However, for steady–state water–oil tests, this oil is replaced by a 1.5 cp oil. In all of these tests, strong brines are normally used to avoid clay dispersion problems.

With regard to core sample size, unsteady–state relative permeability tests will be performed on one and one–half inch and two inch diameter samples. Lengths are variable but will be greater than the diameter selected. Steady–state flow tests can be run on smaller samples. Minimum diameter is one inch and length is one and one–half inches.

Endpoint Flood Tests

Water–Oil Endpoint Test (WT):

Endpoint flood tests are less expensive than unsteady–state relative permeability tests but usually provide only endpoint values of saturation and relative permeability because low viscosity water and oil are used. Also, if sample permeability is very low (less than one md), only endpoint tests can be completed in a reasonable time. Preparation procedures are the same as for relative permeability tests. An oil–displacing–water endpoint condition can also be obtained if desired. In addition, the test fluids can be tailored to reflect the reservoir viscosity ratio condition.

Frequently, additional details at the endpoint condition are needed when waterflood residual oil saturations depend on water throughput. These data can easily be

collected at the end of the test. If there is sufficient saturation change, relative permeability data can be calculated.

Water–Gas and Oil–Gas Endpoint Tests (WGF and OGF):

These tests are similar to the oil–water endpoint tests with the exception that gas is the phase displaced by the injected liquid. To minimize expansion effects, the tests are conducted at an elevated pore pressure of 200 psig. The water–gas test data are applicable to a gas reservoir undergoing water influx. In contrast, the oil–gas test data are applicable to a gas condensate reservoir. Here the production of high pressure gas reduces reservoir fluid pressure which results in the formation of hydrocarbon liquid within the formation pore spaces.

The data of interest in both tests is the residual gas saturation trapped by liquid. Test results include initial and final residual gas saturations and relative permeabilities.

Residual Gas Determination (RG):

An inexpensive test providing only residual gas saturation data is conducted by immersing clean, dry core samples in water or oil. Frequent weighing provides the gas (i.e. air) saturation existing at the end of spontaneous imbibition. Residual gas saturation, porosity, air permeability, and depth are obtained for each sample tested.

Unsteady–State Relative Permeability Tests

Gas–Oil Relative Permeability Test (GO):

Gas–oil relative permeability data (gas saturation increasing) aid prediction of oil recovery and producing rates for the following reservoir production mechanisms: solution gas drive, gas cap expansion and gravity drainage of oil. In the laboratory flow test, nitrogen is injected into an oil–saturated core sample (frequently containing a connate water saturation) to displace a 25 cp mineral oil. Displacement time, produced gas and oil volumes, and pressure differential data permit calculation of gas saturation and gas and oil relative permeabilities.

Water–Oil Relative Permeability Test (WO):

Water–oil relative permeability data (water saturation increasing) aid prediction of oil recovery and producing rates for the following reservoir production mechanisms: water influx and field waterflooding. In the laboratory test, use of a 25cp mineral oil lowers the water saturation obtained at water breakthrough, and thereby increases the amount of two–phase production from which relative permeability data are calculated. An initial water saturation should be present or established in the core sample, because the measured flow behavior is dependent on the initial saturation condition of the sample. Displacement time, produced water and oil volumes, and pressure differential permit calculation of water saturations and water and oil relative permeabilitites.

Gas–Water Relative Permeability Test (GW):

In this test, gas displaces water from a fully water –saturated core. The test data can be used to compute the restoration time of effective gas permeability of a wellbore

zone, or fractured zone, invaded by water. The laboratory test procedure is the same as that used for gas–oil flow test GO except that water is substituted for oil.

Gas–Water Relative Permeability Test with Irreducible Oil Present (GWO):
This test is designed to follow water–oil flow test WO and, with the exception of the presence of the residual oil saturation, is performed using the procedure used for the test GW. The collected data are applicable to a tertiary, alternate gas–water injection process.

Steady–State Relative Permeability Tests

Steady–State Water–Oil Relative Permeability Test (WOSS):
Water–oil relative permeability data (water saturation increasing) aid prediction of oil recovery and producing rates for the following reservoir production mechanisms: water influx and field waterflooding. Relative permeability testing using the steady–state method offers several advantages compared to unsteady–state or waterflood tests. The saturation range covered depends only on the water–oil injection ratio rather than fluid viscosities, and the flow test data are collected under conditions of flow and wetting equilibrium. Further, calculations of relative permeabilities require only the Darcy flow equation, a relationship that underlies all modern reservoir engineering principles.

The flow test begins at the initial water saturation, usually an irreducible value, and stepwise changes in the volumetric water–oil injection ratio (e.g., 0.1, 0.4, 2, and 10) produce increasing levels of water saturation. All data are collected after attaining flow equilibrium. The test ends at a water floodout condition when only water is injected into the test sample. Saturations are measured during the flow test using an x–ray absorption technique.

Steady–State Oil–Water Relative Permeability Test (OWSS):
Oil recovery for tertiary processes may involve the buildup of an oil bank which then displaces flood water ahead of it. Oil–displacing–water relative permeability data aid in predicting the performance of this fluid movement. The test can be conveniently performed immediately following Test WOSS.

Steady–State Water–Gas Relative Permeability Test (WGSS):
This test is similar to a steady–state water–oil flow test (WOSS) except that gas is substituted for oil. To minimize gas expansion effects, the test is conducted at a static pressure of 200 psig. Since complete relative permeability relationships are generated by the steady–state method (rather than only endpoint flow data), the water and gas relative permeability curves can be used with mathematical models to describe the flow of water and gas for CO_2 and alternate gas–water injection processes.

Steady–State Oil–Gas Relative Permeability Test (OGSS):
This test is the same as Test WGSS except that oil is substituted for water. An interstitial water saturation is established in the test sample. The results are

applicable to a gas condensate reservoir undergoing pressure decline. Production of high pressure gas reduces reservoir pressure which results in the drop out of hydrocarbon liquid within the formation pore spaces. The liquid reduces reservoir gas permeability. The steady–state procedure provides complete relative permeability curves for use in mathematical model recovery predictions.

Capillary Pressure Tests

Mercury Injection Test (MI):

Mercury injection tests provide basic capillary pressure and pore size characteristics (both drainage and inhibition) and aid in identifying formation rock types. The tests involve the step–wise injection of instrument grade mercury into a clean, evacuated core sample using a series of pressures between 0 and 2,000 psia. Higher injection pressures can be requested. The amount of mercury which enters the sample is measured at each predetermined pressure point. Pore throat radii and pore size distribution can be subsequently calculated. Additionally, these data can be converted to air–brine relationships applicable to gas reservoir systems.

Centrifuge Capillary Pressure Test (CPC):

This test uses a centrifuge to determine capillary pressure curves for small core samples. The advantages of using a centrifugal method are the excellent test accuracy obtained, the availability of high pressure differences between phases, and rapid establishment of saturation equilibrium.

Saturated core samples are placed in a core holder with a bottom well for fluid displacement mesurement. The samples are then subjected to step–wise increases in rotational speed, and the volume of displaced liquid is measured at each equilibrium state. A tachometer and calibrated stroboscope are used for determining centrifuge speed and fluid volume. Centrifuge rpm is converted to pressure for determination of capillary pressure (psi). Various two–phase combinations are available including air/brine and oil/brine systems. Capillary pressure is obtained as a function of percent mercury saturation.

Porous–Plate Capillary Pressure (PCP):

This test involves the use of a semi–permeable ceramic plate to determine the capillary pressure characteristics of core samples. Although this type of capillary pressure test requires a longer testing time than other methods offered, it provides results on samples as large as two inches in diameter.

Test samples are cleaned, measured for air permeability and porosity, and then saturated with the prescribed fluid. The samples are then placed on the ceramic plate and are subjected to step–wise air pressure increases. Sample liquid saturations are gravimetrically measured at each equilibrium point. This test can be performed using air displacing either oil or water from the samples. The maximum pressure possible is approximately 200 psi in the air–water system and approximately 70 psi in the air–oil system. Capillary pressure is obtained as a function of percent liquid saturation.

Index of Program Variables

Appendix K

The index provides an alphabetical listing of the program variables that the user may enter.

Sample Job Control Language — Appendix L

BOAST runs in three versions, one for MVS, one for VM, and one for AIX.

MVS

Below is an example of the JCL to initiate a BOAST run under MVS. The input data is a file called LAF3900B. The user is requesting 50 units of computer time. Try to request the smallest amount of time that will complete the job so that the mainframe will not delay the job for too long.

```
//BOASTII JOB (1309,73811,50,200),'PECLAY',CLASS=H,
//          TIME=50,REGION=74M,
//          MSGLEVEL=(1,1),MSGCLASS=S,
//          NOTIFY=PECLAY
//*************************************************************
//JOBLIB DD DSN=VSF2.VSF2LOAD,DISP=SHR
//*************************************************************
//*IEFBR14 EXEC PGM=IEFBR14
//*DD2 DD UNIT=3380,DSN=PECLAY.BOASTII.RESTART,DISP=(MOD,DELETE),
//*                 SPACE=(3200,(1,1))
//*************************************************************
//GO       EXEC  PGM=B2MTFVM,REGION=8M,TIME=50,
//          PARM=('NOXUFLOW,NODEBUG,AUTOTASK(B2MTFVM,1)')
//STEPLIB   DD    DSN=PECLAY.BOASTII.LOAD,DISP=SHR
//AUTOTASKDD  DSN=PECLAY.BOASTII.LOAD,DISP=SHR
//FT20F001  DD    DSN=PECLAY.BOAST.DATA(LAF3900B),DISP=SHR
//FT06F001  DD    SYSOUT=*
//FT16F001  DD    SYSOUT=*
//FT07F001  DD    SYSOUT=*
//FT24F001  DD    DSN=PECLAY.B2MTFV.RESTART,DISP=SHR
//FT25F001  DD    DSN=PECLAY.B2MTFV.RESTART,DISP=SHR
//FTERR001  DD    SYSOUT=*,DCB=(RECFM=F)
//FTERR002  DD    SYSOUT=*,DCB=(RECFM=F)
//FTERR003  DD    SYSOUT=*,DCB=(RECFM=F)
//FTERR004  DD    SYSOUT=*,DCB=(RECFM=F)
//FTERR005  DD    SYSOUT=*,DCB=(RECFM=F)
//FTERR006  DD    SYSOUT=*,DCB=(RECFM=F)
```

VM

Below is an example of an EXEC file used to initiate a BOAST run in VM.

```
/* THIS IS A REX EXEC FOR THE BOASTIV RUN */
'FI 6 DISK BOAS2X58 OUTPUT A (LRECL 132'
'FI 7 TER'
'FI 20 DISK BOAST2X3C DATA A'
'FI 25 DISK BRESTART DATA A'
'LOAD BOASTVI (START RMODE ANY AMODE ANY'
/*  */
```

AIX

Below is a sample command line for performing a BOAST run on the IBM RISC SYSTEM 6000, which runs under the AIX operating system.

```
XLF -lessl -Pv -Pk -Wp,-opreprocessed.i  -o BOAST1t -O -NQ 40000
      -NA 50000 -NT 50000 <file>
```

The options used are:

-lessl: This calls the "lessl" library. Very few subroutines from this library are used by BOAST: the subroutines DGTF and DGTS are used to perform the LU factorization of the matrixes.

-Pv -Pk: This calls the 2 pre-processors available on the RISC/6000. Usually a pre-processor deals with instructions such as "INCLUDE" statements or the optimization of empty loops, cache optimization, or even declaration optimization. For example, when an "INCLUDE" statement is met by the pre-processor, the content of the file referenced by the statement is added in the source file. But the pre-processors on this machine can also be a powerful tool to increase the speed of the final executable file. Most of the speed increase is performed by the new Pk pre-processor (released in December, 1991), which can perform control and data flow analysis for the entire program unit. For example, the RISC/6000 contains a system that can execute two non-floating point operations at the same time. Then the pre-processor transforms the source code before compiling so that this capability is efficiently used. It also allows optimizations, such as global register assignments, so that variables are retained in registers where possible to eliminate unnecessary LOAD/STORE operations. For example the following loop:

```
DO I=1,50
      A(I)=0
ENDDO
```

is transformed into:

```
DO I=1,50,2
      A(I)=0
      A(I+1)=0
ENDDO
```

-Wp,-opreprocessed.i,-F=file.i : Generates 2 files, "pre-processed.i" and "file.i," which contain the source code after all pre-processing modifications are completed. The -F flag introduces the name of the source file pre-processed with the Pk pre-processor, and the -o flag introduces the source file pre-processed by using the Pv pre-processor.

-o BOAST: Gives the executable the name "BOAST"

-O: Starts the optimization at the end of the compilation.

-NQ 40000 -NA 50000 -NT 50000: Indicates the size of the compiler's tables. As the default sizes are too small for this source code file, we have to specify another range of sizes.

Compiling and Optimizing

The compiler does not optimize programs by default. The optimization request has to be made explicitly in the command line. In BOAST, some variables were uninitialized or initialized inconsistently with the program code and caused initial problems. For example, the program code which follows can be compiled with an executable file occurring on run-time, but it cannot be optimized by processing:

```
I=0
...
...
A(I)=1
```

Such a code is incorrect, but it compiles, and the inconsistency (which did not modify any of the program results) was found because of a preprocessing failure during compilation. Also, compiling with optimization takes longer, because the compiler must restructure the code to apply the many optimization techniques. These techniques are described in the following paragraph. One problem is that the debugger will not work properly with the optimization option.

The optimization Techniques:

Techniques used are as follows:

–Inlining: Inlining is the process of replacing a subroutine call or function reference with the text of subroutine or function. By default this option is off and has to be evoked explicitly by the programmer. It also checks between formal and real parameters.

–Dead Code Elimination: The compiler can eliminate code for unnecessary calculations. For example the following loop:

```
DO I=1,50
A=7
ENDDO
```

will be eliminated. Nested loops that are not dependant on each other are found.

–Dead Store Elimination: Since main memory is a basic component of the RISC architecture improvements, it has to be managed efficiently. Usually when a member cell has been statically or semi–dynamically bound to a variable, the binding remains until the procedure has finished. This provides a dynamic elimination for the variables which are not going to be used any longer in the run time.

–Code motion: If variables used in a computation within a loop are not altered within the loop, the calculation may be performed outside of the loop and the results used within the loop.

–Straightening: Straightening is rearranging the program code to minimize the branching logic and to combine physically separate blocks of code.

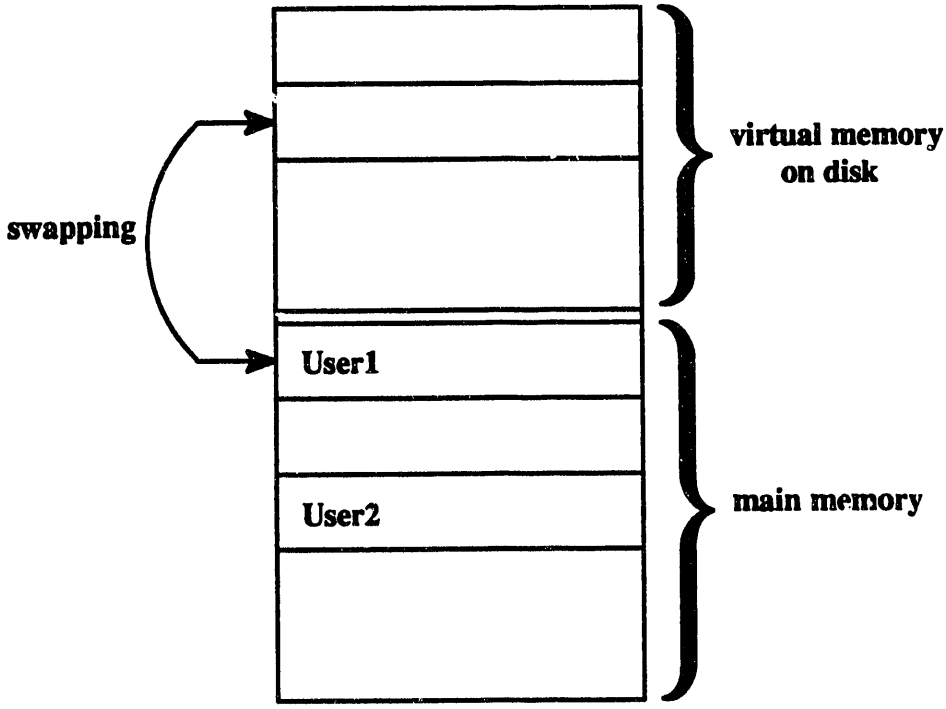
–Common expression elimination: In common expression elimination, the same value is recalculated in subsequent expressions. The duplicate expression can be eliminated by using the previous value. For example, if the program contains the following statements:

```
10 A=C+D
.....
.....
20 F=C+D+E
```

then the result of the first operation will be used to calculate F.

An IBM RISC/6000 can reach the limits of its efficiency only if it has enough memory available in RAM, because the Virtual Memory Manager (VMM) starts "Swapping." A great deal of Input/Output is created for the CPU calculation. These executions of BOAST are very long relative to the average time spent on the CPU.

Such situations should be avoided because the RISC/6000 pages so much that the CPU is not used and the performance of the machine decreases.



virtual memory

Management of the RISC Memory

Computers like the IBM 3090 cannot directly access their entire memory, but with statements such as "Dynamic COMMON," they can. The RISC/6000 has the capability for accessing up to two gigabytes of RAM. The smallest memory configuration for an IBM RISC/6000 is not recommended for BOAST, because after an application reaches the real memory capacity of the machine, the performance decreases progressively. BOAST requires up to 52 megabytes of paging space for a typical field–wide simulation.

CHANGES TO THE ORIGINAL BOAST:

Two important changes have been made to the BOAST version for the 3090 to run the RISC SYSTEM/6000:

1) **The file names:**

   A file name, for example "fort.31" was changed to "oilsat.31." This file is the "oil saturation file." The number 31 is the unit number for the FORTRAN compiler. The water saturation filet was renamed to "watersat.32," etc.

2) **FORTRAN instructions not standard:**

Subroutines such as CLOCKX or CPUTIME which give the time and CPU time inside the program were removed. They are not accepted by XLF (the FORTRAN compiler on the RISC/6000). They have been replaced by equivalent statements available in the ESSL or BLAS libraries. For example, the SUBROUTINE CLOCKX was replaced by the function CLOCKM (). Parts of the original program have been commented out because statements did not have an equivalent. All changes are clearly marked with comments.

Pre–processing Programs

Appendix M

A series of four PC–based FORTRAN programs are available to help prepare BOAST II input. These programs and sample input files are provided on a diskette included with this manual. Four programs assist in converting well field descriptions into a reservoir model. In additional there are two auxiliary programs for manipulating and contouring model matrices. The programs are illustrated by a model of the *TEST* reservoir.

Primary Pre–processors:	
GRID_FIT	uses digitized reservoir outline and key well locations to align the model with the reservoir and define a grid with the highest resolution being near the key wells.
KRIG	interpolates and extrapolates irregularly spaced data by kriging to fill an II x JJ x KK matrix.
CON_FILL	interpolates between digitized contours to fill an II x JJ matrix.
BOAST_MK	compiles all interpolated matrices into a BOAST II input file.
Auxiliary Pre–processors:	
MATMOD	allows the modification of an II x JJ x KK matrix, Z_{old}, by: $Z_{new} = C_{slp} * (Z_{old} - Z_{ref})^{Power} + DZ_{ref}$ where all of the variables can be either a constant or the name of a file with an II x JJ x KK matrix.
CONTOUR	contours data from irregularly spaced II x JJ matrices based on the grid spacings stored in *.GRD.

The compile command line, with appropriate options, has been listed for each program since several of the programs call external libraries. The source codes for two of the external libraries, QDISSPLA and PVT, are included in this package. QDISSPLA contains a series of basic plotting routines that mimic ISSCO's DISSPLA graphics calls. PVT contains fluid property correlations for oil, water, and gas.

All programs with graphics calls must be compiled with MICROSOFT's compiler and linked with MICROSOFT's graphics library (version 5.0 or greater) as follows:

```
fl GRID_FIT.for                 /link graphics.lib
fl KRIG.for qdisspla.obj        /link graphics.lib
fl CON_FILL.for
fl BOAST_MK.for pvt.obj
fl MATMOD.for
fl CONTOUR.for                  /link graphics. lib
```

The file nomenclature and contents of each file type are summarized in Table 1.

TABLE 1. — DEFINITION OF FILE NOMENCLATURE AND CONTENTS

MANDATORY Files that must exist:

*.KEY[6]	digitizer unit/foot scale, reference well, and names of files with digtized reservoir outline and well locations.
TEST.dig[7]	digitized reservoir outline and basemap of ***TEST***.
wells.dig	digitized locations of key wells.
TEST_krg.thk	X,Y,Z data for kriging reservoir thicknesses.
TEST_ci.tos	digitized contours that define the top of sand.
TEST.PRD	production data in ASCII format as printed from dBASE.
TEST_WEL.LOC	well names and locations for reservoir TEST.
TEST.BII	description of miscellaneous reservoir properties for reservoir TEST.
BOAST_II.DEF	default description of miscellaneous reservoir properties.

OPTIONAL files:

TEST_krg.phi	basic X,Y,Z data for kriging reservoir porosity.
TEST_krg,kxy	basic X,Y,Z data for kriging reservoir permeability.
*.PRS	pressure history in ASCII format.
CONTOUR,DEF	default contouring options for various data types.

Suffixes taken from input files to denote output files with matrices:

*.THK	gross thickness of all sand and clay layers within a model.
*.SO	initial oil saturation (an II x JJ x KK matrix)
*.SW	initial water saturation (an II x JJ x KK matrix)
*.TOS	top of sand (an II x JJ matrix)
*.PHI	porosity distribution (an II x JJ x KK matrix)
*.KXY	XY–permeability distribution (an II x JJ x KK matrix)
*.KX	X–permeability distribution (an II x JJ x KK matrix)
*.KY	Y–permeability distribution (an Ii x JJ x KK matrix)
*.KZ	Z–permeability distribution (an II x JJ x KK matrix)
*.P3D	pressure distribution (an II x JJ x Kk matrix)
*.GRD	grid spacing and locations in X and Y. Also contains a matrix that delineates the reservoirs lateral extent.
*.MAP	digitized base map file after being rotated and scaled to model orientation and dimensions.
*.INP	BOAST II input file produced by BOAST_MK.

6. The reservoir name code is substituted for all files with a * as their prefix.
7. Full names are completely arbitrary.

GRID_FIT

GRID_FIT defines an II x JJ grid that is aligned with the reservoir outline and is finest near key wells. It also scales and rotates any digitized files to produce new files in the selected model scale and orientation.

REQUIRED INPUT FILES:

TEST.key:
defines basic scaling from digitizer units to feet, identifies reference well, and names files with digitized reservoir outline and well locations.

TEST.dig:
digitized reservoir outline and basemap of TEST.

TEST.dig:
digitized reservoir outline and basemap of TEST.

WELLS.dig:
digitized locations of key wells.

Sample *.key file for the *TEST* Reservoir.

```
FEET =        DIGITIZER UNITS
5000.000000  20000.000000
<x,y> coordinate of reference well i  digitizer units.
30000.        40000.
Arbitrarily selected <x,y> coordinate of reference well in feet.
0000.         10000.
Angle of rotation in degrees.  If angle > 360, queries for data rotation.
400
List of files with <x,y> data pairs to be transformed.....
test.dig<first entry should be model outline.>
wells.dig
>>>> End file input list <<<<
Minimum grid block width (dx) and depth (dy).
   200.          150.
File with <x,y> coordinates of higher resolution locations.
wells.MAP
II & JJ size.  If II or JJ equals 0, queries for computing new grid.
   0      0
^^^^^^^^^^^^^^^^^^End of File^^^^^^^^^^^^^^^^^^^
```

Sample TEST.dig file for the *TEST* Reservoir.

```
???? CI:-11588              C:BLUE    Oil/Water contact
 38041.99    39672.91
 37869.85    39650.81
 37154.68    39542.50
     .          .
 38041.99    39672.91
?? Control points           C:ORANGE  M:C,S,0.45,7
 30000.00    40000.00
     .          .
```

Note : Question marks serve as delimiters which indicate a change in the type of data.

In addition to the model outline, the file of digitized data usually contains structural features such as the top of sand.

```
     .          .
 31914.10    40927.00
101610.20    59781.52
 85206.13    64083.41
?????END
^^^^^^^^^^^^^^^^End of File^^^^^^^^^^^^^^^^^^
```

WELLS.dig file for the *TEST* Reservoir.

```
?? Production               C:PURPLE  M:C,S,0.45,12
   33948.50    47788.01
   30000.00    40000.00
   22939.41    42593.86
?????END
^^^^^^^^^^^^^^^^^^End Of  File^^^^^^^^^^^^^^^^^^
```

OPTIONAL INPUT FILES:

Any other digitized data sets that need to be rotated and scaled.

OUTPUT FILES:

TEST.GRD:
contains grid spacing, node locations, and a matrix that delineates the reservoir's lateral extent.

TEST.MAP:
digitized basemap file after being rotated and scaled to model orientation and dimensions.

T_EDGES.GRD:
is a file of <x,y> pairs that can be appended to the basemap file to show the model grid.

KRIG

KRIG interpolates and extrapolates irregularly spaced data by kriging (Karlingerand Skrivan, 1980)* to fill an II x JJ x KK matrix. This program is primarily for constructing a full three–dimensional description of a reservoir's sand and clay (shale) distribution. If adequate data is available, ***KRIG*** can also be used to construct matrices of porosity, permeability, pressure, or saturation distributions.

Structure maps can be constructed using ***KRIG***, but it is ill–advised. The scant quantity of data usually available produces maps that are considerably different from hand–contoured maps. Alternate means to construct a top of sand matrix are provided in ***CON_FILL.***

Spatial interpolation by kriging is based on the internal structure of the data set (autocorrelation). this structure, if present, is examined by plotting differences in Z squared (gamma) against the distance between the two values of Z. The number of unique comparisons, ntot, is:

$$ntot = n * (n-1) / 2$$

where,

n = the number of <x,y> locations

If 20 <X,Y> locations and their associated Z values are used, this results in 190 unique comparisons.

The scatter plot of gamma vs. distance for clay and sand thickness in the TEST reservoir (Fig. M1a) shows a general increase in gamma as pairs become further apart. An overall structure, however, is hard to determine since a great degree of variability exists at any given point. A structure is made more apparent by averaging gamma over intervals (Fig. M1b). The resulting plot is commonly referred to as an empirical semi–variogram. The empirical semi–variogram shows that differences in thickness increase with distance up to 9,000 ft apart, but beyond that gamma remains near 50 ft^2. (Thickness differences are about 7 ft.)

*Karlinger, M.R. and Skrivan, J.A., 1980, Kriging Analysis of Mean Annual Precipitation, Powder River Basin, Montana and Wyoming, U.S. Geolgoical Survey, Water Resources Investigations 80–80, 25 p.

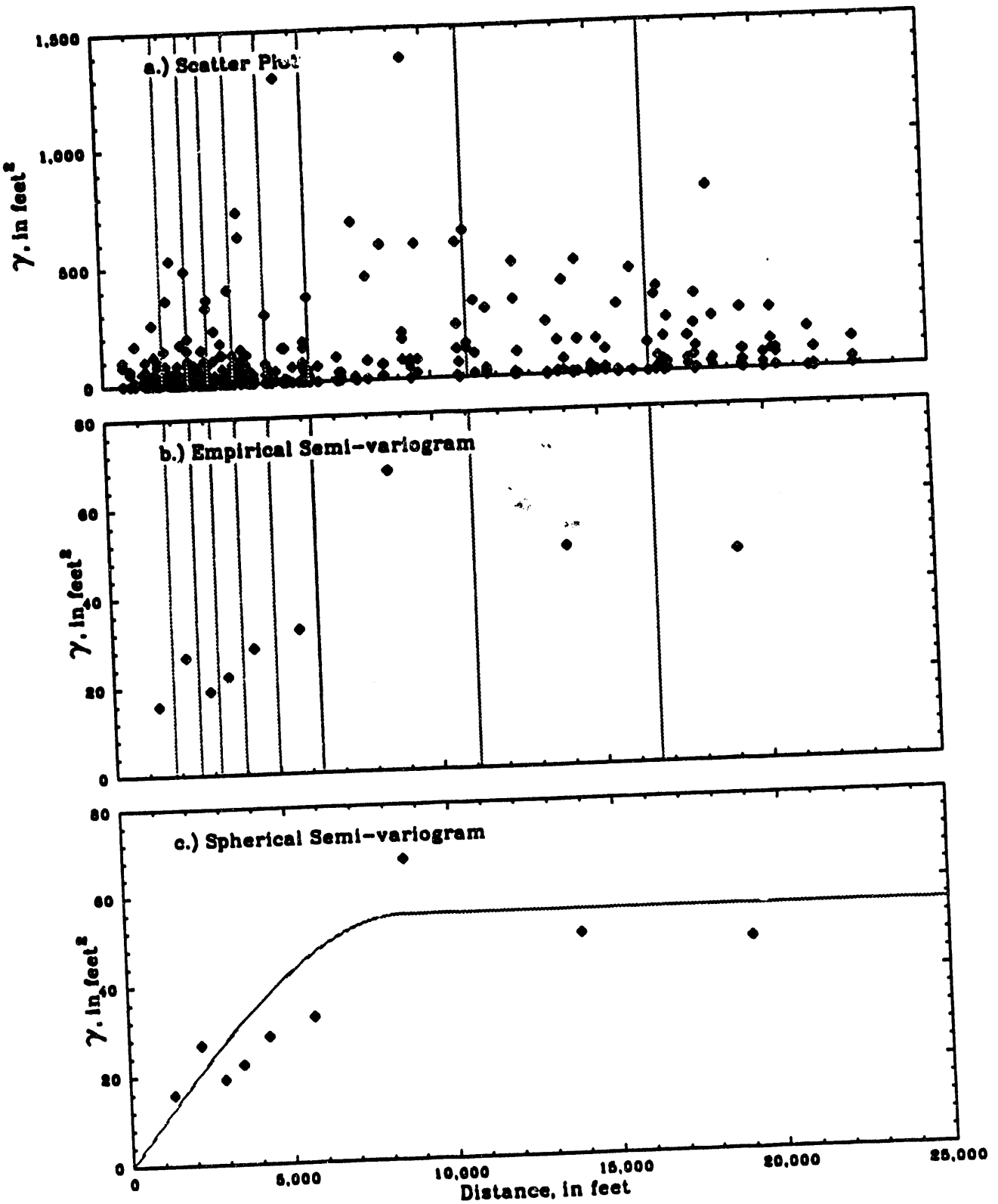


Figure 1.--Semi-variogram for sand and clay thickness in the TEST reservoir.

A theoretical semi–variogram provides a functional form of the relation between gamma and distance, which is needed for kriging. The spherical form best describes the example distribution (Fig. M1c) and most other thickness data. Several other theoretical forms have also been implemented in ***KRIG*** and are illustrated in Figure M2 (Strivan and Karlinger, 1980)*.

The theoretical semi–variogram governs the emphasis placed on each Z value when interpolating; as gamma decreases, the emphasis increases. Theoretical semi–variograms are defined by:

the maximum distance that gamma varies,
the minimum value of gamma, the nugget, and
the maximum value of gamma, the sill.

If there is no uncertainty associated with the measured Z values, gamma @ x=0 will be equal to zero. This makes kriging an exact interpolator when the nugget equals zero.

Obvious trends (a.k.a.: drift in kriging papers) are accounted for by a user defined polynomial equation:

$$Z_{drift} = \sum_{j=1}^{NT} a_j \left[x^{ex(j)} * y^{ey(j)}\right]$$

where,

ex = j^{th} drift term in x, user defined,
ey = j^{th} drift term in y, user defined,
a = j^{th} drift coefficient, solved for by kriging system, and
NT = number of drift terms, user defined.

The power terms are supplied by the user and the kriging system solves for the coefficients. No trend will be present for most data sets, especially thickness data. For these cases, NT=1 and ex=ey=0. This reduces Z_{drift} to a_1 which is equal tot he average value of Z.

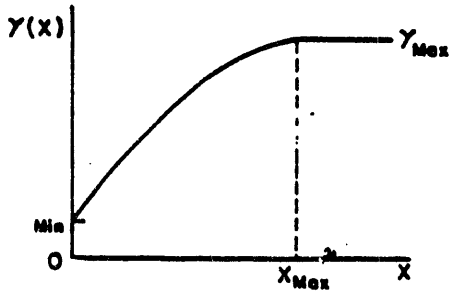
*Skrivan, J.A. and Karlinger, M.R., 1980, Semi–variogram Estimation and Universal Kringing Program, U.S. Geological Survey, Computer contribution, 98 p.

SPHERICAL

$\gamma(X) = 0 \Longrightarrow X = 0$

$\gamma(X) = \gamma_{Min} - \gamma_{Min}\left[\frac{3}{2}\frac{X}{X_{Max}} - \frac{1}{2}\left(\frac{X}{X_{Max}}\right)^3\right] + \gamma_{Min} \Longrightarrow 0 < X < X_{Max}$

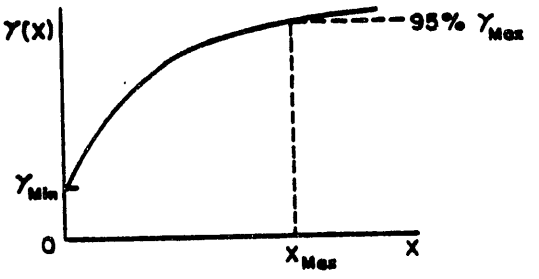
$\gamma(X) = \gamma_{Max} \Longrightarrow X = a$



EXPONENTIAL

$\gamma(X) = 0 \Longrightarrow X = 0$

$\gamma(X) = (\gamma_{Max} - \gamma_{Min})\left(1 - e^{-\left(\frac{X}{3X_{Max}}\right)}\right) + \gamma_{Min} \Longrightarrow X > 0$

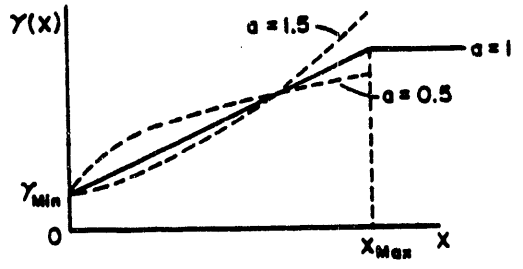


LINEAR, PARABOLIC OR ROOT

$\gamma(X) = 0 \Longrightarrow X = 0$

$\gamma(X) = (\gamma_{Max} - \gamma_{Min})\left(\frac{X}{X_{Max}}\right)^2 + \gamma_{Min} \Longrightarrow 0 < X < X_{Max}$

$\gamma(X) = \gamma_{Max} \Longrightarrow X > X_{Max}$



GAUSSIAN

$\gamma(X) = 0 \Longrightarrow X = 0$

$\gamma(X) = (\gamma_{Max} - \gamma_{Min})\left(1 - e^{-\left(\frac{X}{1.73X_{Max}}\right)^2}\right) + \gamma_{Min} \Longrightarrow X > 0$

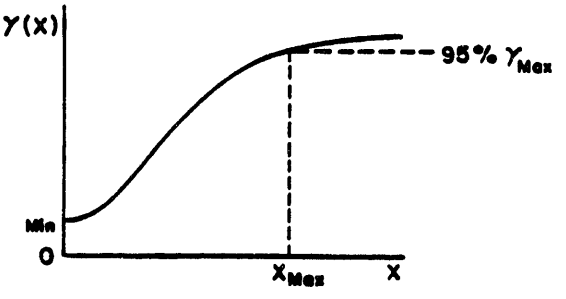


Figure M2 — Theoretical semi–variogram types available in KRIG

The interpolated–Z value at each model node is the sum of all Z values times a corresponding set of weights plus the drift term or:

$$Z_{interpolated} = \sum_{i=1}^{n} w_i * z_i$$

where,

$w_i = i^{th}$ weight solved for the kriging system.

For each <x,y> location, n+NT simultaneous equations are solved to determine the weights, w_i, and the drift coefficients, a_j. This analysis is limited to examining variation in only one layer.

Semi–variogram construction and interpolation by kriging has been extended to multiple layers assuming that *the variation within each layer is of similar magnitude and an average semi–variogram for all layers is adequate.* These assumptions holdup best for thickness data where variations in sand and clay thickness are usually complementary. A set of weights and drift coefficients is calculated for each <x,y> location and is used for all layers.

REQUIRED INPUT FILES:

TEST.GRD:
contains grid spacing, node locations, and a matrix that delineates the reservoirs lateral extent.

TEST_KRG.THK:
supplies reservoir name, grid information, and basic X,Y, Z data for kriging reservoir thicknesses.

OUTPUT FILE:

TEST.THK:
contains the initial data, the theoretical semi–variogram description, and the resulting sand and clay thickness matrices.

Sample thickness file for the *TEST* Reservoir.

```
                    Kriged sand and clay thicknesses in the TEST reservoir.
                          Model grid information is in: (only prefix is important)

                    test.grd
Number of layers to be Kriged is:
 3
Semi-Variogram type, Dist. Max, gamma @ d=0, gamma @ d=Max, (Power term)
Spherical (If unknown, type = ?)   8000.                0.                55.
   X              Y              drift terms added from 1-10 (Recommend1 to 3)
   0.             0.
Enter X,Y thickness:              S1          C1-2        S2b
   10000.00   10000.00            12          5           13
    7042.67   10248.43            7           2           15
      .                .          .           .           .
   14000.90    8590.64            7           0           25
Enter end point values.....
? or use values INSTEAD to override automatic endpoint selection.
End of input.....
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^End Of File^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

CON_FILL

CON-FILL interpolates between digitized contours to fill an II x JJ matrix. This program is an alternate method of generating matrices that define reservoir structure.

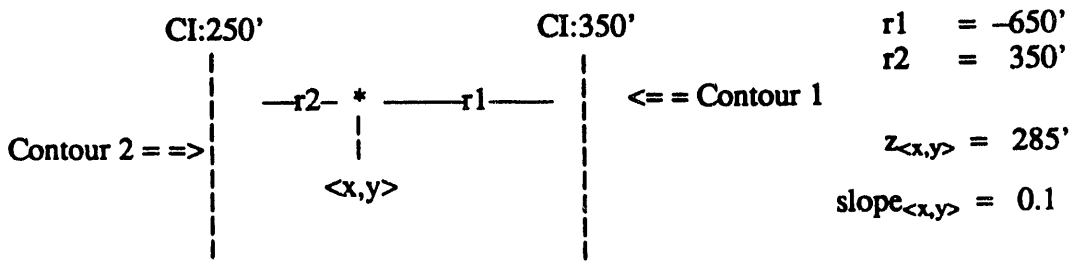
Initially, the program scans the entire contour file to locate the highest contour. Matrices of distances, r, between simulator nodes and the highest and second highest contours are generated (Halford, 1992)*. Distances located updip of a contour are positive while those that are downdip are negative.

*Halford, Keith J., 1992, Incorporating Reservoir Characteristics for Automatic History matching, Baton Rouge, Louisiana, Louisiana State University, Department of Petroleum Engineering, unpublished Ph.D. dissertation, 150 p.

Interpolation occurs at locations where the distance to the higher contour is negative and the distance to the lower contour is positive. Z is interpolated by:

$$z_{<x,y>} = (CI1 - CI2)\ \frac{r2}{(r2 - r1)} + CI2$$

where,



This procedure is repeated for each successively lower contour until the matrix is filled or there are no more contours. A by–product of this procedure is an additional matrix with the slopes of the surface.

REQUIRED INPUT FILES:

TEST.GRD:
contains grid spacing, node locations, and a matrix that delineates the reservoirs lateral extent.

TEST_CI.TOS:
digitized structure map file after being rotated and scaled to model orientation and dimensions.

OUTPUT FILES:

TEST.TOS:
contains top of sand an slope of surface matrices.

T.TRA:
A trash file with the distance matrix and a matrix that identifies the contributing segment.

Sample test ci.tos file for the *TEST* Reservoir.

```
?????            CI: -11550    <= = = Contour interval sequence is unimportant.
   10046.72      10059.44
   10142.42      10097.70
      .             .
      .             .
   10046.72      10059.44      <= = = Repeat first <x,y> pair for closed contours.

????             CI: -11545    .
                                  Note: Contour is assumed to be closed if
   10250.        10740.          its endpoints are within 0.4 ft of one
   10240.        10730.          another.
   10260.        10730.
   10250.        10740.
 XXXX            CI: -11575    < = CI: is tag that identifies the contour interval.
   08630.4       09387.5
   08920.6       09182.1
      .             .
   08229.2       09799.4
   08630.4       09387.5
 ???End
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^End Of File^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

BOAST_MK

BOAST_MK compiles all interpolated matrices into a BOAST II input file. Wherever matrix input is not available or not appropriate (miscellaneous information such as reservoir temperature, initial pressure, oil gravity, depth to oil/water contact, etc.), input is read from *.BII, BOAST_II.DEF, or the screen. *.BII file contains the data specific to a given reservoir. BOAST_II.DEF contains default values that will be used if not found in *.BII. If no information is available in either file, the user is queried for the data.

A few special options are noted:

–Permeability and porosity data can be supplied as a full II x JJ x KK matrix, by layer, or as a constant.

–Clay (shale) layers can be taken out by incorporating their vertical resistance to flow in a KZ matrix (McDonald and Hargaugh, 1988, pp. 2–29 to 2–35).*

–Variable initial oil and water saturations are easily accommodated.

REQUIRED INPUT FILES:

TEST.GRD:
contains grid spacing, node locations, and a matrix that delineates the reservoir's lateral extent.

TEST.thk:
contains the initial data, the theoretical semi–variogram description, and the resulting sand and clay thickness matrices.

TEST.TOS
contains top of sand and slope of surface matrices.

TEST.PRD:
production data for the ***TEST*** reservoir as written to an ASCII file.

TEST.BII:
a control file with a reservoir description for ***TEST*** reservoir.

BOAST_II.DEF:
a file with generic reservoir descriptions if test.BII not present.

TEST_WEL.LOC:
a file with well names and locations.

OPTIONAL INPUT FILES:

TEST.KX:
a full II x JJ x KK matrix of X–permeability values.

TEST.KY:
a full II x JJ x KK matrix of Y–permeability values.

TEST.KXY:
a full II x JJ x KK matrix of XY–permeability values.

*McDonald, M.G., and Harbaugh, A.W., 1988, A modular three–dimensional finite–difference ground–water flow model: U.S. Geolgoical Survey TWRI, book 6, chap. A1, 576 p.

TEST.KZ:
a full II x JJ x KK matrix of vertical permeability values.

TEST.PHI:
a full II x JJ x KK matrix of porosity values.

TEST.SW:
a full II x JJ x KK matrix of initial water saturation values.

TEST.SO:
a full II x JJ x KK matrix of initial oil saturation values.

OUTPUT FILES:

TEST.INP:
finished BOAST II input file

Sample *.BII or BOAST II.DEF file for the *TEST* Reservoir

= indicate numeric values follow.

: indicate characters follow.

[ ] – Start default description .....

X–permeability	=	100.	300	400	md
Y–permeability	=	100.			md
Z–permeability	=	20.	40	40	md
Porosity	=	0.25			
thickness minimum	=	0.25			feet
K–clay	=	5.			md
Overall minimum permeability	=		1.		md
Clay identifier:			C		
Treatment of clay layers:		Quasi–3D			
Oil Saturation, initial	=		0.68		
Irreducible oil saturation	=		0.27		
Water saturation, initial	=		0.32		

Note: Descriptor preceding IIxJJ matrices in TEST.thk file.

Initial water saturation is assumed to be irreducible also.

Gas saturation, Irreducible	=	0.36
Initial reservoir pressure	=	8000. psia
api	=	38. oil gravity
Gravity of gas	=	0.65 (air = 1.0)
Salinity of brine	=	20 ppt
Fahrenheit temperature of reservoir	=	150. degrees
Datum of oil/water contact	=	11588 ft BLS

Aquifer is: edge–water drive, other choices are bottom–water or none.

Sample *.BII or BOAST II.DEF file for the *TEST* Reservoir Continued

Production well locations are in: TEST_WEL.LOC

Qtype: Total – specifies how production is specified in BOAST II. Can specify Oil, Water, Gas, or Total voidage, For history matching, total voidage is recommended.

Change well rates = 182.5 days

Time steps = 0.75 minimum 1.00 initial 90. maximum in days

Start and stop time conventions are – Month / Year

The year can be denoted with either 2 or 4 digits, e.g. 62 or 1962.

Start simulation = 5 / 62 (Leave spaces before and after each number.) No date or a date earlier than discovery will make the earliest date, minus a month, be time 0.

Stop simulation = 6 / 1999 (Leave spaces before and after each number.) No date or a date later than the last production entry will make the latest date be time 0.

[ ] – end

^^^^^^^^^^^^^^^^^^^^^^^^^^^^End Of File^^^^^^^^^^^^^^^^^^^^^^^^^^^^

MATMOD

MATMOD allows the modification of an II x JJ x KK matrix, Z_{old}, by:

$$Z_{new} = C_{slp} * (Z_{old} - Z_{ref})^{Power} + DZ_{ref}$$

where all of the variables can be either a constant or the name of a file with an II x JJ x KK matrix. It is assumed that all matrices start after the first line in the file and that the first KK contiguous blocks of II x JJ numbers is the matrix of interest.

CONTOUR

CONTOUR contours data from irregularly spaced Ii x JJ matrices based on the grid spacings stored in *.GRD. The contouring method is detailed in Hargaugh (1990)*.

*Harbaugh, Arlen W., 1990, A Simple Contouring Program for Gridded Data, U.S. Geological Survey, Open–file Report 90–144, 37 p.

Contouring options for different data types are stored in the control file CONTOUR.DEF. The options for different data types are found by matching the input file suffix to a description heading. For example, [thk] is the description heading that marks where the contouring options for thickness data are. If no suffix matches the data being contoured, the empty brackets, [ ], provide the settings.

Within a description block, an = sign triggers updating a contouring option. All of the options are not required to be in a description block and they do not have to be in any particular order. The options are determined by the first 2 letters of the descriptor and are:

1. LOgarithmic contouring provides opportunity to contour log values of Z matrix. A minimum value of $Z < 0.0$ is provided to trigger option.
2. PLotting area is defined by two coordinates the lower left, SW, and upper right, NE, corners. Four values can be entered directly or as a percentage of the model dimensions.
3. MAximum z; all values greater than this are not contoured.
4. COntour intervals can be specified:

   automatically, by leaving a blank field or test, by entering one value which is repeated, directly, up to 25 values, or as a percentage of DZ = Zmax – Zmin.
5. LAbel interval is the number of feet between contour labels.
6. STarting point is the number of feet travelled along a contour prior to the first label being applied.
7. BAsemap provides the name of a file with the basemap. A blank response generates the default file of ***"reservoir name"***.MAP. This suffix can be changed by entering*. ***"new suffix"***.
8. SHow mesh does or does not draw the contouring mesh based on a Yes/No response.

Percentages are identified by placing a % sign on a line and can define:

Xmax – Xmin, DX, or Ymax – Ymin, DY, for plotting area,

Zmax – Zmin, DZ, for contour intervals,

Base length – DX + DY for label interval,

Label intervals for starting point.

Any or all of these options can be made interactive by putting a ? mark after the = sign.

REQUIRED INPUT FILES:

TEST.GRD:
contains grid spacing, node locations, and a matrix that delineates the reservoir's lateral extent.

Any file with at least an II x JJ size block of numbers.

OPTIONAL INPUT FILES:

CONTOUR.DEF
a control file with pre-defined contouring options.

OUTPUT FILES:

t.cnt:
A list of <x,y> pairs generated from the last matrix contoured.

Sample contour.def file for the *TEST* Reservoir.

[ ] – Default description if the suffix is not recognized.....

logarithmic contouring = no

plotting area = 0 0 100 100 %

maximum z = 1.E20

contour intervals = Auto

label interval = 30 %

starting point = 50 %

basemap = ?

show mesh = n

[thk] – Default description for thickness data.....

logarithmic contouring = no

plotting area = All

contour intervals = 4. Repeat

basemap = *.map

show mesh =

[s?] – Default description for saturation data.....

maximum z = 1.000

show mesh = y

contour intervals = 5% Repeat

logarithmic contouring = no

[k??] – Default description for permeability data.....

logarithmic contouring = .01 minimum z

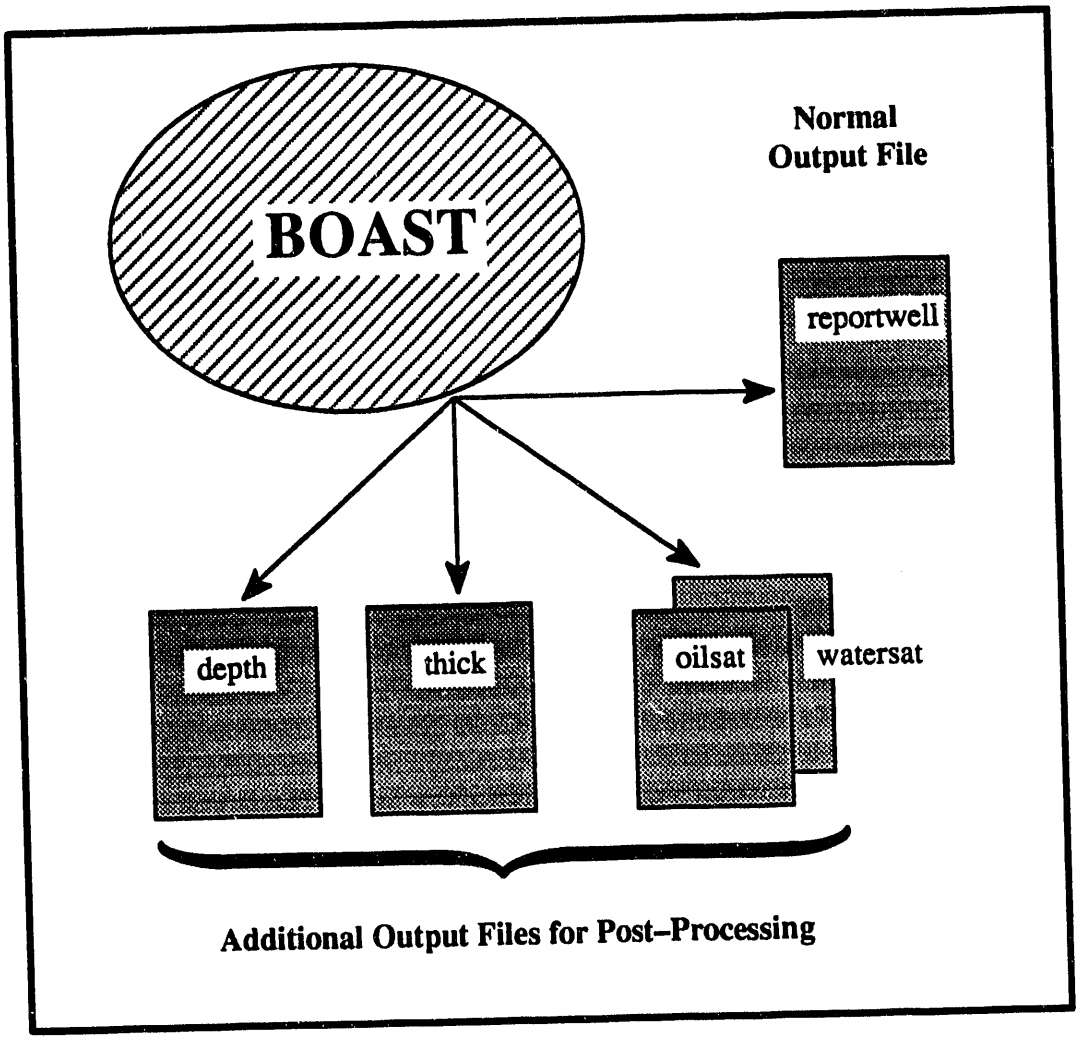
contour intervals = 5% Repeat

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^End Of File^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

Post Processing Programs

Appendix N

Software was developed to present results from BOAST in a more usable and desirable manner. This software is provided on the Post–Processing diskette. The RISC/6000 version of BOAST II was modified to produce additional output files for this purpose.



LSU has a well–equipped Remote Sensing and Image Processing Laboratory (RSIP) that was used to provide this post–processing capability. The LSU RSIP was created to develop spatial information systems and to promote image presentations of data. The RSIP information systems permitted efficient data analysis, interpretation, and modeling of the spatial data that was produced by a single execution of BOAST. With each BOAST execution, input data can be entered that will change the resulting output. The post–processing procedure can present the output as color–coded maps, volume models and cross–sections of such quantities as pressures, depths, saturations, and concentrations. Each of these results can be created as a se-

quence of time–steps, providing the capability to get images for a day–by–day progression, like a movie of progressive images, over a several day period.

Reservoir parameters were displayed as scalar quantities such as pressure, temperature or saturation, and mapped onto a spectrum of colors. The color mapping was straightforward and used a perceptually uniform color space. The intent was to provide a global idea of the operational history of an oil field.

The RSIP laboratory used for this process had a dedicated UNISYS 7000/40 computing system, two model 74 image processing displays, 10 Sun Microsystem Workstations and 3 Integraph workstations. In addition, a Hewlette Packard Paintjet printer was available for color printing. Most of this BOAST post–processing was done on a SPARC station, 2GX. This SPARC station 2GX had the following:

- 28.5 MIPS and 4.2 M flops
- 64 KB Cache
- 96 MB available–16 MB of memory plus swap space
- SUN's graphics libraries Xlib, Pixwin, or Sun Vision
- 2–D and 3–D wireframe graphics

Most of the BOAST post–processing used Sun Vision software. This software had a set of highly integrated visualization tools with components for image processing of 2–D and 3–D data, volume rendering analysis, photorealistic rendering, and interactive display and manipulation of 3–D geometric data. These components were designed to allow sharing of images and data. Sun Vision had the following components:

Sun IP – image processing tool
Sun IP Lib – image processing library
Sun Voxel – volume rendering tool
Sun ART – photorealistic rendering tool
Sun GV – interactive geometry viewer
Sun Movie – movie display tool
Sun VIF – reconfigurable use interface tool

For this BOAST post–processing project, the Sun Voxel was used to generate the oil field models at different stages and the Sun Movie was used to display the resulting movies.

Input Data and Additional Output

Because BOAST produces output results that are not required for this Post–Processing, and because a large volume of output is needed for a movie, the BOAST program was modified to produce a series of additional output files whose explicit use would be for this post–processing. To do this, additional programs were written which used additional input, three dimensional indicies (i,j,k), to create the required output. This input file defined the reservoir area that would be used to make the "BOAST–movies." These indicies were also written out and used in the post–processing programs.

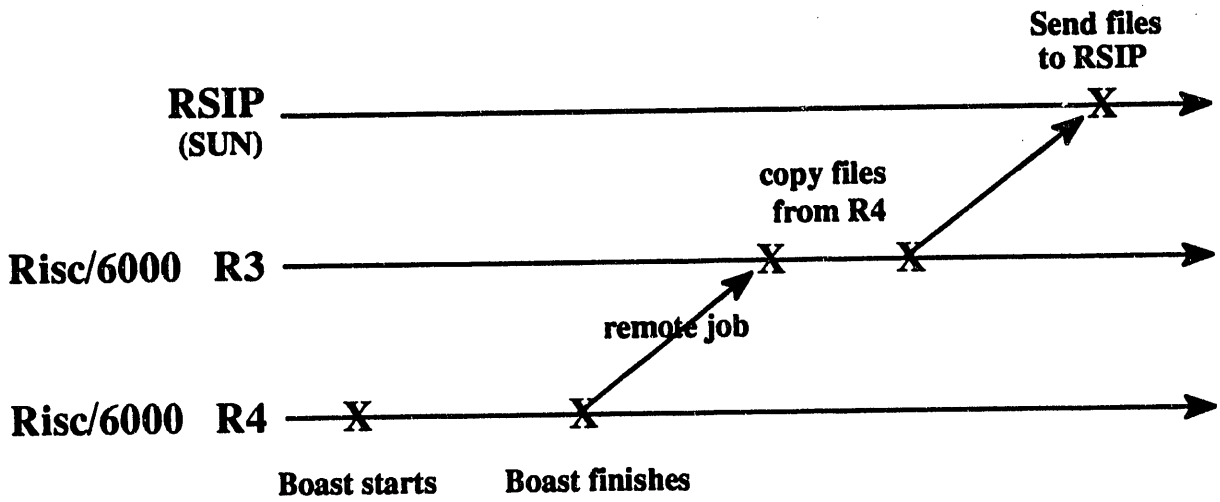
At LSU, the additional BOAST II output files needed for post–processing were transmitted from the RISC/6000 system to the RSIP work stations by means

of network communication. This transmission implementation was written in UNIX shell statements as shown below.

```
 echo "You are logged to : 'hostname' "
if [ 'hostname' = r4.pete.lsu.edu ]
then
        echo " Remote execution on r3 ...."
fi
set 'date'
ladate=$3
A="$HOME/TRANSFERT"
if [ -f $A/oilsat.31 -o -f $A/watersat.32 -o -f $A/fort.31 -o -f $A/fort.32 ]
   then
        echo " Transferring data.... "
        set '1s -1 |tr "-" "a" |grep oilsat.31'
        echo "Here is the date of the file : $7"
        echo "Here is the date of today  : $ladate"
        echo " ">$A/ordres
        if [ $7 -eq $1adate ]
                then
                     echo "$7 : correct date"
                     echo "put oilsat.31" > $A/sousordres
                     cat $A/ordres $A/sousordres> $A/som
                     cp $A/som $A/ordres
    fi
    set '1s -1 |tr "-" "a" |grep "watersat.32"'
    if [ $7 = $1adate ]
                then
                     echo "$ladate : correct date"
                     echo "put water sat.32" > $A/sousordres
                     cat $A/ordres $A/sousordres> $A/som
                     cp $A/som $A/ordres
    fi
    set 'ls -l |tr "-" "a" | grep "boast_watersat.32"'
    if [ $7 -eq $ladate ]
                then
                     echo "$ladate : corrct date"
                     echo "put boast_watersat.32" > $A/sousordres
                     cat $A/ordres $A/sousordres> $A/som
                     cp $A/som $A/ordres
    fi
    set ' ls -l |tr "-" "a" | that was  grep "boast_oilsat.31"'
    if [ $7 -eq $ladate ]
                then
                     echo "$ladate : correct date"
                     echo "put boast_oilsat.31" > $A/sousordres
                     cat $A/ordres $A/sousordres> $A/som
                     cp $A/som $A/ordres
```

```
    fi
    echo "put thickness.29" > $A/som
    echo "put depth.30" > $A/som1
    cat $A/som $A/som1 > $A/sousordres
    cat $A/ordres $A/sousordres > $A/som
    cp $A/som $A/ordres
    cat $A/ordres
    ftp sarah.rsip.lsu.edu < $A/ordres
    rm $A/som
    rm $A/som1
    rm $A/ordres
    rm $A/sousordres
  else echo " None of the expected files is present ."
fi
```

The BOAST "movies" were created by using a series of results, each computed at a constant time increment, i.e. "one per day." Selected output from BOAST was transmitted to a remote sensing and image processing laboratory (RSIP) for presentation. The implementation of this data transmission between departments is illustrated below.



These links were put together with a program written to do this which used the Sun Movie. One problem that occurred was that these movies have large disk space requirements. To generate a movie of 200 days, with one frame per day, this file required 223 MB of disk space. C–language programs were written to do this part of the post–processing. One program produces the movie and the other displays volume models. The period of time between frames is arbitrary and could correspond

to a day, week, month, or any other desired time period. Figures M1 and M2 show the programs, changes, and output files that were created.

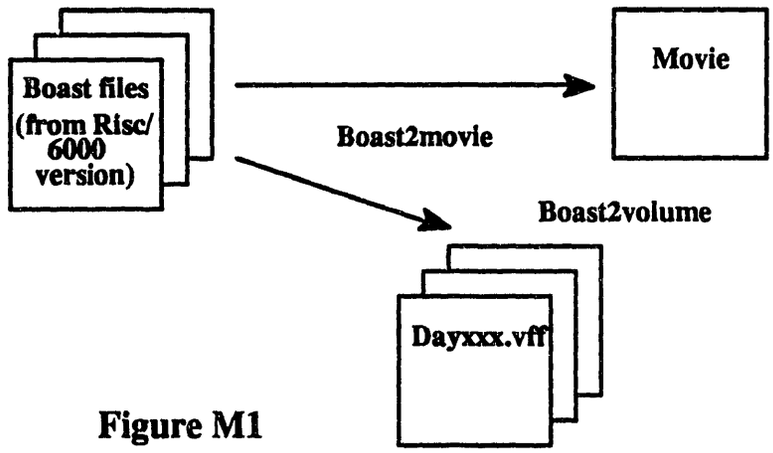


Figure M1

The first step shown in the above schematic is the translation of the output from a BOAST volume model to a Binary volume model. Though they differ by just one word, they have very different meanings.

The **BOAST volume model** is defined from the data files given by **BOAST**. At least three data files are needed: 1) one for the depth of the first layer, whose name is **depth**, 2) one for the thickness of each cube, whose name is **thick**, and 3) one for the definition of the parameter, whose name can be **oilsat.31, watersat.32, pressure.33**, etc.. These data files contain integer values and are stored in two–dimensional (for **depth**) or three dimensional (for **thick** and **oilsat**) matrices, whose name are respectively **De, Th** and **Pa**. The dimensions (**col, row** and **lay**) of these matrices are fixed by **BOAST** and can be different from time to time.

According to the data file **thick**, each cube defining the **BOAST volume model** can have a different thickness. That means that a layer in the **BOAST volume model** can not be represented by a flat slice. A layer doesn't have a uniform thickness.

The **Binary volume model** is defined from these previous matrices with binary values. These binary values are stored in a three–dimensional matrix **array**. The dimensions (**imax, jmax** and **itb**) of this matrix are fixed by the procedure **fort.c** at 32*32*32.

A layer in the **Binary volume model** can be represented by a uniform slice. Each layer defining the **Binary volume model** has the same thickness. The integer defining the thickness between each layer is **it**. Each flat slice in this model may contain information from several **BOAST** layers. The Binary volume model permits manipulation and display with commonly used Sun workstation software.

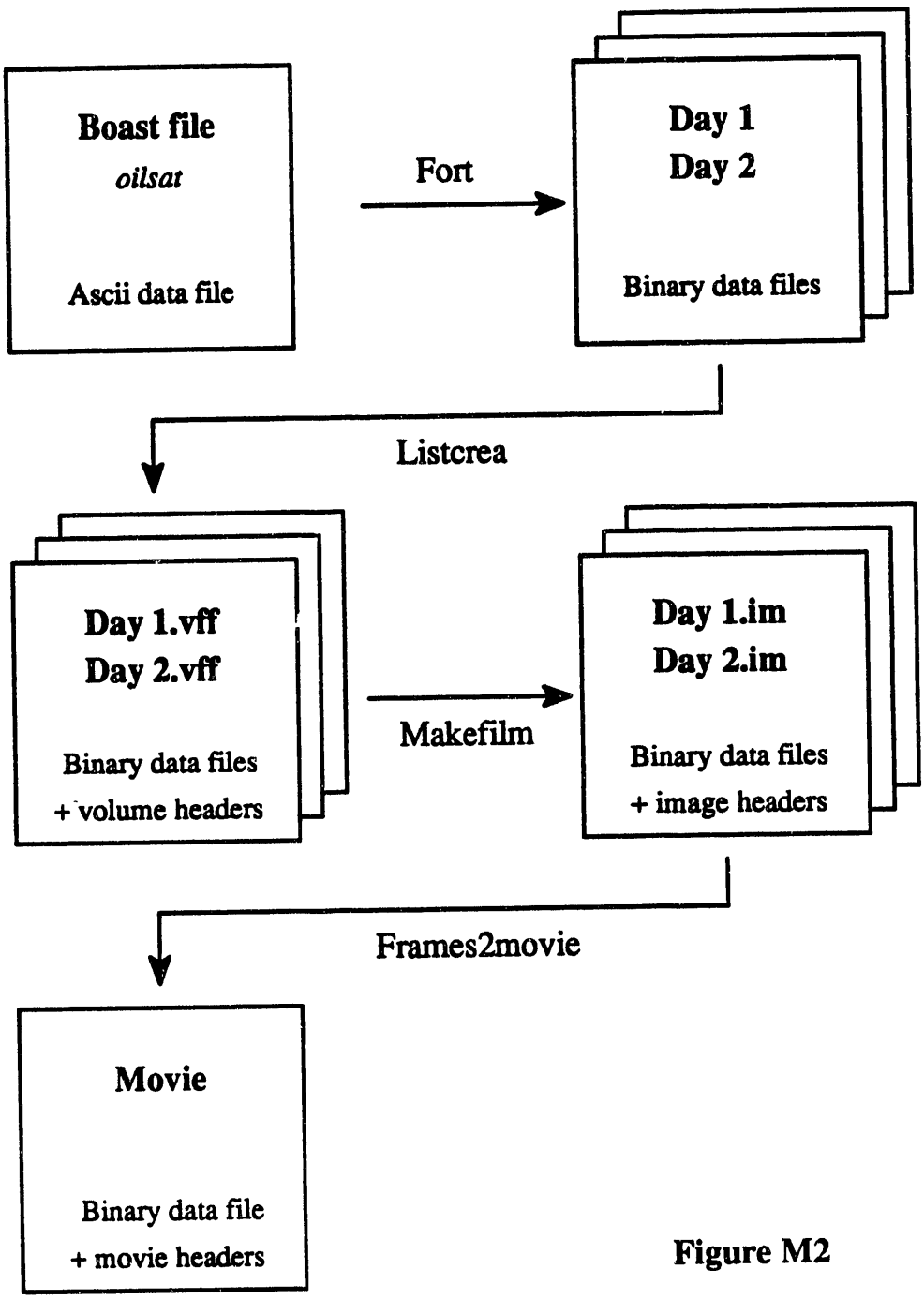


Figure M2

3D Model Visualization System For Open Windows

3D Model Visualization System—3DMVS provides the necessary tools to visualize and analyze, under Open Windows, the models generated by BOAST. It

combines the programs BOAST2Volume, BOAST2Movie and Sun Vision into a single integrated environment. In addition to the existing functions, several useful user–friendly interfaces such as *DataFile Edit, RawData View, and 3D Model View,* are incorporated into the 3DMVS.

3DMVS was developed, under XView ToolKit for OpenWindows, to integrate the individual programs into a menu driven system for visual analysis of field models. Its basic structure can be depicted in Figure M3.

DataFile Edit:

In order to visualize models obtained from the BOAST, files associated with the depth, thickness of the model blocks (known as volumes), oil saturation and report well are required. The interface of *DataFile Edit* is designed to facilitate the use of 3DMVS for different data files (names). That is, data file names need not be the depth, thick, oilsat.31 and reportwell.33, and any file name can be assigned to these four data files.

RawData View:

It will some times be necessary to view or modify the raw data generated by the BOAST before the graphic model is generated. RawData View interface provides such capabilities. The user can use scrollbar to search data (in ASCII format), or open the menu palette to modify the data. The menu palette consists of the basic functions for searching and editing the data. It contains four items, *File, View, Edit,* and *Find*, each of which has its own sub–functions.

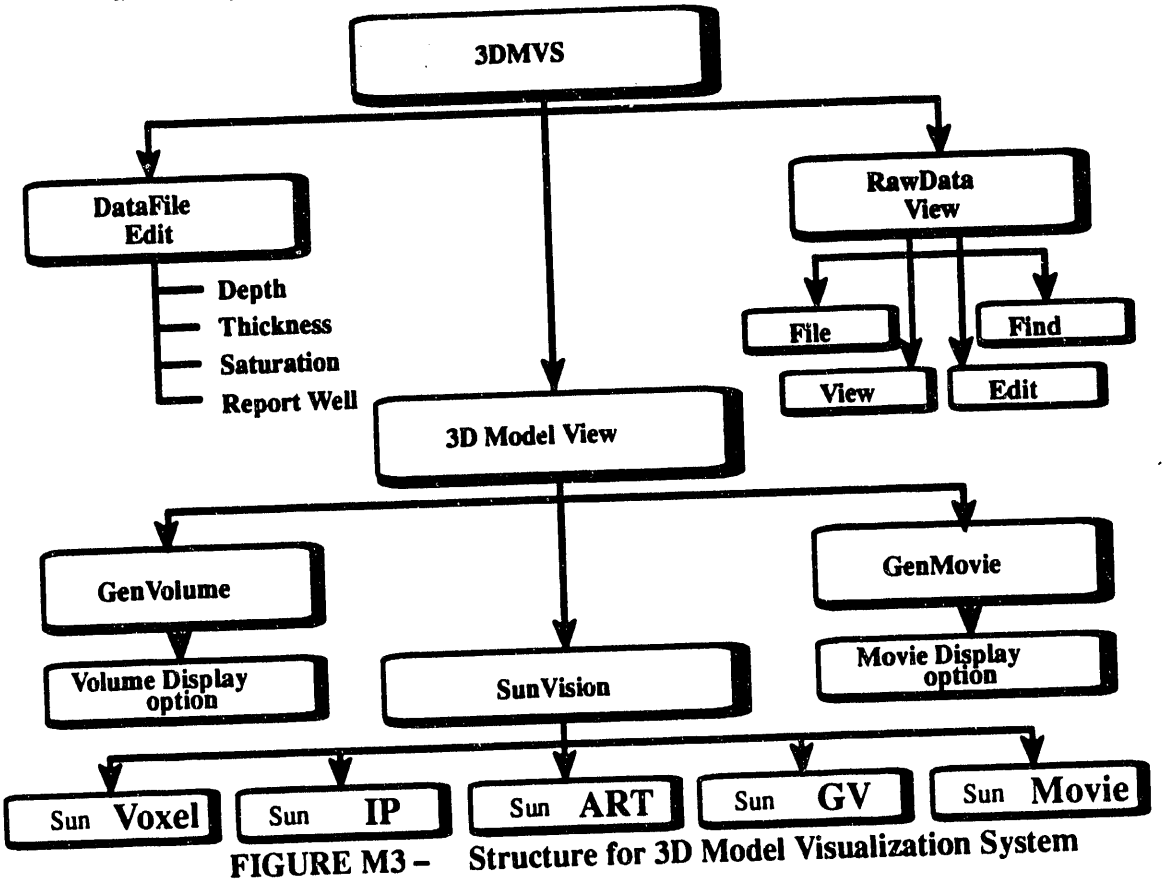


FIGURE M3 – Structure for 3D Model Visualization System

There are five sub–functions under the item *File*: Load File, Save Current File, Store as New File, Include File, and Empty Document. Similarly, Select Line at Number, What Line Number, Show Caret at Top and Change Line Wrap are associated with the *View*; Again, Undo, Copy, Paste and Cut with the *Edit*; Find and Replace, Find Selection, Find Marked Text and Replace 1>Field<1 with the *Find*.

ModelView:

ModelView is a visual display manager which not only has access to the existing external programs such as BOAST2Volume, BOAST2Movie and SUNVision, but also provides necessary options for volume and movie display. For example, a user can specify the volume model for a particular day. In this case, only one volume data can be displayed at a time. Similarly for movie display, the user can specify the volume model for a particular day. In this case, only one volume data can be displayed at a time. Similarly for movie display, the user can select any set of volume data associated with several days among available days. Thus, more than one volume data can be visualized in frame by frame fashion. Moreover, the volume data can be automatically animated by SunMovie. ModelView interface is very useful for analyzing and visualizing the data in the volume model.

2. Getting Started

This section gives the details needed to start 3DMVS.

1) *Start 3DMVS from the OpenWindows' command window:*

   a) Type olst and press the Enter key to start the OpenWindows session.

   b) Press the rightmost button of the mouse and choose the command tool from the Program menu. This step can be skipped if there is a command window on the screen when OpenWindows session starts.

   c) Change the directory to the 3DMVS main directory from the command window.

   d) Once you're in the 3DMVS main directory, type 3DMVS and press the Enter key. This starts a 3DMVS session. The main window of 3DMVS, containing push buttons *DataFile, RawDataView, 3DModelView* and *Quit*, will appear on the screen.

   Note: The files: day.dat containing the number of days available, Boast2Volume, Boast2Movie and the geometric data files associated with the depth, thick, saturation and report well must be kept in the same directory as the ***3DMVS***.

2) *Select the data files to be processed*

Once the 3DMVS session is started, you may select from the DataFile dialog box the file names associated with the depth, thick, saturation and report well. The default file names are depth, thick, oilsat.31 and reportwell.33.

Clicking the *DataFile* push button from the main window of 3DMVS will open a dialog box for data file editing. There are four lines labeled as depth, thickness, saturation and report well in the dialog box. Pointing to the line where you want to change the file name either by using the tab key or mouse, then key in the correct

file name. The changes made to the file names will not be saved and therefore the default file names will be used unless you select the *Save* button in the dialog box.

3) *Generate and display volume model*

Before you can analyze the model obtained from the BOAST in 3D graphics fashion, volume data should be generated from the data associated with the geometric parameters such as the depth, thick, and oil saturation. Because in each data file there are several sets of data, each of which is associated with a particular day, you may select any of the available days. For example, if there are 7 sets of data associated with day 0, day 182, day 547, day 912, day 1277, day 1642 and day 2000 respectively, you can then select any one of them for display.

GenVolume is a push button used to open the volume display option dialog box, in which you can specify a particular day for volume display. In the volume display dialog box, there is a scrolling list of available days.

To open volume display option dialog box:

a) Click the *ModelView* push button in the main window of 3DMVS to open a control panel containing three buttons: GenVolume, GenMovie and SunVision.

b) Click the *GenVolume* button to open the volume display dialog box.

c) Select in the dialog box one of the days either by clicking the item on the list or enter the name of the day in the text input field.

d) Click the *OK* push button to start volume display for the selected day, or the *Cancel* button to quit the volume display.

Remember that the volume display requires the SunVoxel in SunVision. By default, SunVision will be invoked after you push the *OK* button. However, you still need to select the SunVoxel in the SunVision main window. It is suggested that the SunVision and SunVoxel be invoked from the ModelView control panel before using the volume display.

After the volume data is displayed in the SunVoxel window, you can then select any tools from the pull–down menu in the window to analyze the volume model. For example, you may open the cube–model dialog box to change the size and view angle in 3 dimension, conduct profile analysis etc.. You may also open the color palette to change the color mapping for the display.

4) *Generate and display the movie*

SunMovie is a tool box provided in the SunVision. This tool box can be used to visualize the volume model in frame by frame manner; it can display several sets of data associated with different days in sequence. Similar to the volume display, the sets of data should be converted to the one acceptable by SunMovie. Such conversion can be accomplished by the program Boast2Movie. In the 3DMVS, movie display options are provided to facilitate the analysis of volume data. The movie display option dialog box allows the user to select any sets of data among available days other than the whole data sets in the data files associated with the depth, thick and oil satu-

ration. Therefore, the movie display option is very useful if you want to skip some sets of data for movie display.

To open movie display option dialog box:

a) Click the *GenMovie* button in the *ModelView* control panel to open the dialog box.

b) Select the days to be displayed by clicking the items on the scrolling list1 labeled–days available. Once the items are selected, they are displayed on the scrolling list2 labeled–days selected. Items can also be deselected by clicking the selected items again on the scrolling list1.

c) Click the *OK* button to start the movie display for the selected days, or the *Cancel* button to quit the movie display.

5) *Raw Data View*

In the raw data view dialog box, raw data can be searched or modified. There is a scrolling list used for display in the dialog box. This scrolling list can be resized to fit more data in the window.

To open raw data view dialog box:

a) Click the *RawDataView* push button in the main window of 3DMVS to open the dialog box.

b) Push the rightmost button of the mouse to open the pull–down menu in the dialog box.

c) Select the *FileLoad* option to load the data file.

d) Search the data by entering the row number in the *View* dialog box from the pull–down menu.

e) Modify the data by selecting the *Edit* option from the pull–down menu.

f) Save the modified data either in the original data file or in a new data file by choosing the *FileSave* option from the menu.

Note: 3DMVS requires day.dat. This data file contains the names of the available days. Therefore, it is suggested that this file be created with the geometric data files such as depth, thick and oil saturation.

Program **3DMVS** is included on diskette for users having the required SUN workstations and software.

3D Model Display for DOS™ & MicroSoft Windows™

1 Introduction

We have developed 3D model display software ***show*** on IBM compatible PCs. There are two versions of the package, one for DOS and the other for MicroSoft Windows. They were designed particularly for 3D image display of a set of data converted from a raster for-

mat file generated by SunVoxel. This raster format is called visualization file format (.vff) defined by Sun Microsystems, Inc. It contains the header in ASCII format and image data in binary format. The data consists of R.G.B. (red, green and blue) triples ranging from 0 to 255.

Because SunVoxel uses 24–bit colors, each point or pixel on the screen requires 3 bytes to store the color information. Therefore, in order to display an image of the size 800x640, at least 1.53 MB storages are needed. For PC, however, only 256 colors can be displayed simultaneously in super VGA (video graphics array) display mode. This means that each of the R.G.B. triples can only have a color scale ranging from 0 to 63. As a result, PC cannot display as many colors as Sun Sparc Stations, and there is a great amount of redundant color information in the .vff raster file. ***Suntopc*** is another software we have developed to convert the .vff raster file generated by SunVoxel to the .img format acceptable for display on PCs. In fact, ***suntopc*** eliminates the unnecessary color information in the .vff raster file and therefore compresses the data. It can reduce file size by 75%.

2 Getting Started

1) To convert the .vff file into the .img file, type:

   suntopc file1.vff file 2.img

   If the conversion is successful, the image will be displayed on the screen and the data compressed from the file1.vff will be saved in the file2.img.

2) To display the image in file.img (run ***show*** in DOS), type:

   show file.img

3) To run ***show*** in Windows, type:

   win show

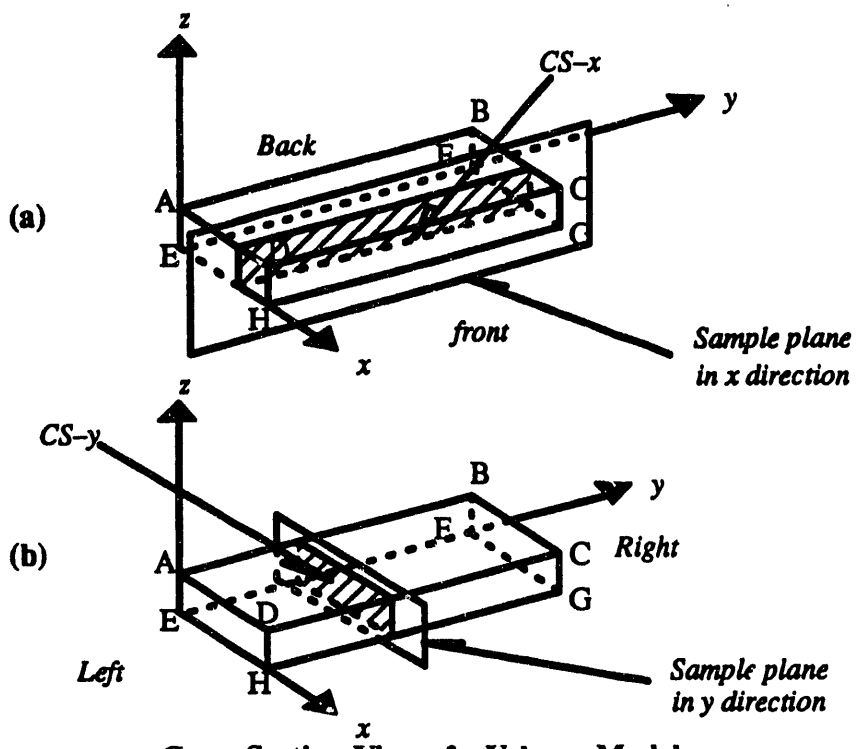
   Then select image file with the suffix .img from the File dialog box provided by ***show***.

Program **show** is included on diskette for an example reservoir to demonstrate the capabilities described above.

Cross Section View for MicroSoft Windows™

1. Introduction

Pfile is a cross–section viewing software we have developed for Microsoft Windows. It has the capability to analyze a volume model generated from the BOAST in 2D graphics fashion. By use of ***pfile***, profiles of the volume model can be visualized in x or y direction as shown in the Figure 2 (a) and (b).



Cross Section View of a Volume Model

In (a) and (b) of Figure 2, ABCDEFGH is an outline of a cube representing a volume model. By moving the sample planes in *x* or *y* direction, we can obtain a set of cross sections, called CX–*x* or CS–*y*. Profiles on the CS–*x* or CS–*y* indicate the oil distribution associated with the model. In ***pfile***, different levels of oil saturation are distinguished by different colors. For example, 90–100% oil saturation is represented by red, 80–89% by pink and so on. There is a color palette provided by ***pfile*** for a user to define the colors of oil saturations. In addition, the profiles displayed on the screen can be resized.

Three menu buttons are available in pfile: File, Display and Color.

File – used to open a File dialog box and load data files

Display– used to open a display dialog box and select display mode. There are two display modes provided for the user:

a) Front to Back–corresponding to the sample planes in *y* direction.

b) Left to Right–corresponding to the sample planes in *x* direction.

Color– used to open the color palette and define colors for oil saturation rates.

2. Getting Started

1) Start ***pfile*** from the DOS prompt, by typing:

win pfile and press the Enter key.

2) Double–click the icon of pfile to start a ***pfile*** session.

3) Click the File menu button to open the file dialog box and then select files associated with the depth, thick and oil saturation by double–clicking the items on the scrolling list.

4) Click Display menu button to open the display dialog box. Select one of the display modes and the section numbers.

5) Click Color menu button to open the color palette. There are ten default colors for the oil saturations ranging from 0–10%, 11–20% and so on. Each color, however, can be changed by clicking the color box, then moving three scrollbars labeled as red, green and blue to define the color.

Program **pfile** is included on diskette for an example reservoir to demonstrate the capabilities described above.

Glossary

Aquifer A water–bearing rock strata. In a water–drive field, the aquifer is the water zone of the reservoir underlying the oil zone.

Array A set of data organized as a grid or a table. The entries can be specified by coordinates. A two–dimensional array can be described with just an I and a J position; three–dimensional arrays are measured as I × J × K entries.

Automatic time–step control A BOAST feature that increases the efficiency of the simulation by increasing the time–step size when changes are occurring slowly in the model and by decreasing the time–step size when model conditions are changing rapidly.

BAND A direct solution method usually applied to problems with a one–dimensional grid.

Bubble point The pressure at which

D4 A direct solution method usually applied to small two– or three–dimensional problems.

Direct solution method An algorithm that solves systems of equations (reservoir processes) through a fixed number of steps. The number od steps is directly related to the size of the grid.

Grid The geometric representation in BOAST of the three–dimensional reservoir space. The grid is a matrix of blocks, each block with its own properties.

IMPES (implicit pressure, explicit saturation) The fundamental approach that BOAST takes to solving pressure and saturation equations. BOAST implicitly solves for pressures throughout the grid, then incorporates these pressure values into equations to explicitly solve for saturation. The new pressure and saturation values are then considered to be the present values. Thus the solutions to the flow equations may be approximated for an arbitrarily long time.

Initialization data In a BOAST input file, the section of data that is read all–at–once at the beginning of the simulation. This section, Chapters 2–12, includes grid geometry, reservoir geology, the solution method, and the aquifer choice.

Iterative solution method An algorithm that computes reservoir processes by starting with an initial guess for the solution vector and iterating upon this initial guess until a convergence criterion is satisfied. BOAST's iterative method is the line successive overrelaxation method (LSOR).

Numerical dispersion

Permeability The ability of a rock to transmit fluid through pore spaces.

Porosity The percentage in a rock that the volume of the pore space bears to the total bulk volume.

PVT Pressure, volume, and temperature. PVT data describes fluid properties.

Recurrent data In a BOAST input file, the section of the data that is read piece–by–piece throughout the simulation. This data controls the simulation as it progresses. This section, Chapters 13 and 14, not only covers time–step and output control but also provides a timetable of well information.

Restart A feature that records in a file the conditions of the simulation at a specified moment so that a user may resume a simulation later without recomputing the initial part of the run. A subsequent run may assume the same history but continue under a new set of conditions.

Restart record A file that holds the information for a restart. The restart record may be written at up to five points during a simulation.

Saturation

SOR The type of iterative method (successive overrelaxation) employed in BOAST.

Throughput

Title A line of arbitrary text that delineates the input, serving to make the input more readable.

Time step A measure of the simulated time (days) between BOAST calculations. The smaller the time step, the more accurate the results and the longer the simulation takes.

Transmissibility A measure of the communication (flux) between grid blocks. The transmissibility between any two grid blocks is a function of the permeabilities and the area in contact between the two blocks.

END

DATE
FILMED
8/5/93