Unstructured Grids for Numerical Reservoir Simulation – Using TOUGH2 for Gas Storage

By W. LITTMANN and K. LITTMANN*

Abstract

For numerical reservoir simulation structured grids are commonly used. These grids are following the geological structure and are bound to the layering of the reservoir. In principal these grids should be regular for the task to solve the differential equations describing the flow of fluids in a reservoir in space and time. For the means of a better adaptation of the grid to the geological structure so called corner point grids have been introduced, which strictly follow the reservoir layering. At faults these corner point grids encounter the problem that grid blocks belonging to different layers should communicate with each other. Formally these non neighbor connections do not comply with the mathematical scheme of the finite difference solution approach.

Another approach for the simulation of fluid and heat flow in reservoirs is the unstructured gridding as it is e. g. used in the TOUGH2 simulator. Grid cells of any geometry can be defined. The location in space of a grid cell is not used in the simulation. Only the volume of a grid cell and the relation to its neighboring cells has to be defined. For each cell the net flow of mass and heat is calculated during the simulation. The flow equations are then represented as Taylor series of the primary variables like temperature or pressure. The Taylor series is truncated after the first term and the resulting system of linear equations for the residuals can be solved using standard lgorithms. Typically the system is solved implicitly in a few iteration steps.

The TOUGH2 simulator has been developed by K. Pruess at the University of California in Berkeley. It is written in Fortran and the source code can be obtained under a license. The way the software is designed makes it very flexible but this concept also makes it difficult to use. The typical reservoir modeling tools applicable in the oil and gas industry cannot be directly used to create a simulation model with TOUGH2.

This paper describes how input data from a simulation input deck such as for Eclipse can be used and how corner point grids can be converted into TOUGH2 input data. The TOUGH2 code can be extended to output data in a format used in oil and gas field simulations. The work flow is discussed by means of examples for gas injection and production.

1 Introduction

The TOUGH (Transport Of Unsaturated Groundwater and Heat) suite of software codes are multi-dimensional numerical models for simulating the coupled transport of water, vapor, non-condensible gas, and heat in porous and fractured media. The software was developed at the Lawrence Berkeley National Laboratory (LBNL) in the early 1980s primarily for geothermal reservoir engineering. The suite of simulators is now widely used at universities, government organizations, and private industries for applications to nuclear waste disposal, environmental remediation problems, energy production from geothermal, oil and gas reservoirs as well as gas hydrate deposits, geological carbon sequestration, vadose zone hydrology, and other uses that involve coupled thermal, hydrological, geochemical, and mechanical processes in permeable media.

Other than the numerical simulation programs used in the oil and gas industry the TOUGH2 simulator uses unstructured grids. The common grids coming from geological modeling tools therefore cannot be used directly in the TOUGH2 simulator.

In the following a procedure is described how to construct TOUGH2-input from a corner point grid as it is used in simulation programs like Eclipse or other simulation programs used for petroleum reservoir engineering and gas storage.

2 Simulation Grids

In numerical reservoir simulation the flow equations for water, oil and gas are solved in time and space. The flow is described by partial differential equations, which usually cannot be solved analytically. Therefore the problem is discretized in time and space and solved by a finite difference approximation. For the finite difference approximation in space in principle a regular 3-dimensional grid is needed as it is shown in Figure 1. The distances are the same for each grid block in each direction.

Such a regular grid, which satisfies the mathematical needs for solving a system of linear equations, does not represent the irregularities of a real reservoir as faults and changing layering.

2.1 Corner point grids

Regular mid-point grids do not adequately represent the sometimes complicated structure and faulting of a reservoir. Therefore corner point grids have been introduced in reservoir simulation to follow the layering of the reservoir and to represent faults. An example is shown in Figure 2.

The size and shape of corner point grids can change throughout the reservoir model. Also like in regular grids the blocks in a corner point grid are numbered in I, J and K. Though the shape of the grid blocks may vary they always have eight corners, belong to a particular layer and share the same area with their top and bottom neighbors. At faults a grid block may have more than one neighbor. These non neighborhood connections can connect to a different layer in the model.

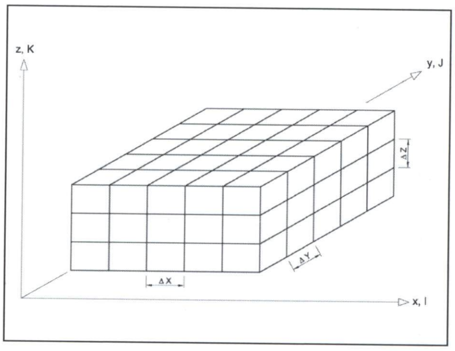


Fig. 1 Regular simulation grid

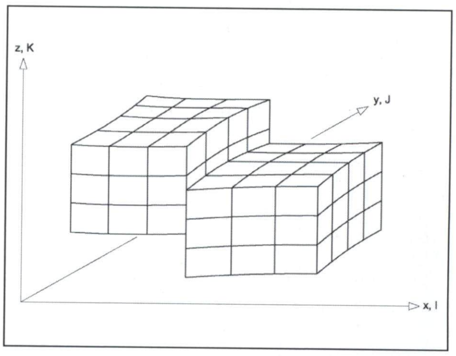


Fig. 2 Example of a corner point grid with a fault

* W. Littmann, K. Littmann, Consulting Reservoir Engineering, Wunstorf (E-mail: W.Littmann@cre-geo.de). Lecture, presented at the DGMK/ÖGEW-Spring Meeting 2012, 19./20. April 2012, Celle, Germany.

0179-3187/12/IV

main focus is on reducing flaring, in more mature areas examples of efficiency increase include the combined generation of heat and power or changing to more efficient engines or cleaner fuels.

There is no immediate solution to efficiency increase. It can only be improved through the sum of smaller steps along the E&P value chain. In RAG ideas and technology were developed to be able to operate successfully in an increasingly mature environment:

- "Fit for purpose" design over field life is key to achieve higher efficiencies
- The modular and mobile design of facilities, multiple generators for drilling rigs, or co-generation motors helps to maintain flexibility in any stage of field life
- Continuous monitoring of fluid level combined with variable speed drives increases lifting efficiency in pump jack systems
- Extending field/well life beyond E&P using for example geothermal applications can lead to additional local energy benefit.

Literature

[1] INTERNATIONAL ENERGY AGENCY, IEA (2011): Key Graphs from World Energy Outlook 2011, November 9th, 2011, Paris.

[2] INTERNATIONAL PETROLEUM INDUSTRY ENVIRONMENTAL CONSERVATION ASSOCIATION, IPIECA, (2012): Energy efficiency: improving energy use from production to consumer, Fact sheet prepared for 2012 United Nations Conference on Sustainable Development (Rio+20), April, 2012, London.

[3] http://www.bp.com/sustainability/environmental performance/energyuse, queried on internet 08/19/2012.

[4] EUROPEAN COMISSION,EC (2011): Proposal for a directive of the European Parliament and of the council on energy efficiency and repealing Directives 2004/8/EC and 2006/32/EC, June 6th, 2011, Brussels.

[5] INTERNATIONAL ASSOCIATION OF OIL AND GAS PRODUCERS, OGP(2011): Environmental performance in the E&P industry 2010 data, Report Nr.466, p.17, October, 2011, London.

[6] http://www.CO2list.org, queried on internet 08/19/2012.

[7] CANADIAN ENVIRONMENTAL TECHNOLOGY ADVANCEMENT CORPORATION WEST, CETAC (2008): Fuel gas best management practices: efficient use of fuel gas in pump jacks, study for the Canadian Association of oil producers & the energy resources conservation board, May 2008, Calgary.

After finalizing his Petroleum Engineering study in Leoben and Colorado School of Mines, ***Henrik Mosser*** *started his career with RAG Rohöl Aufsuchungs AG, Vienna, Austria in November 1995. From 2002 to 2006 he was responsible for Business Planning and Economics. In that period he finalized his Master of Business Administration from IMADEC University Vienna and University of Texas in Austin. After two years as superintendent production operations in Upper Austria he became Manager Asset Gas being in charge of RAG's gas exploration and production in Austria and RAG's E&P joint ventures in Bavaria. From 2009 on he is also member of the supervisory board of RAG Hungary. Since April 2012 his responsibility is now RAG's international E&P joint ventures/operations in West-/Southeast Europe as well as the assessment of new E&P co-operation opportunities. Henrik Mosser is also member of the board of OEGEW "Österreichische Gesellschaft für Erdölwissenschaften".*

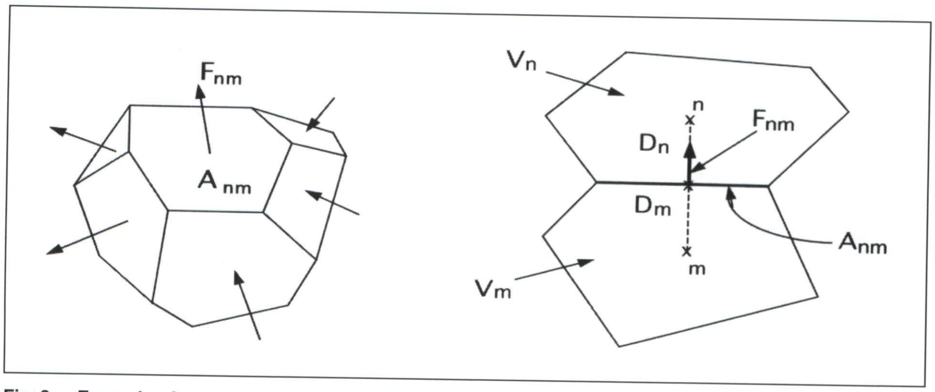


Fig. 3 **Example of an unstructured grid [TOUGH2 manual, p. 148]**

2.2 Unstructured grids

In an unstructured grid the grid blocks do not have a relationship to their neighbor blocks. In an unstructured grid a grid block can have any size and shape. Also the absolute geometric location is not defined. An example of such grid block is shown in Figure 3.
In an unstructured grid as used in TOUGH2 every grid block is only given a name and a volume. If a grid block has communication to other grid blocks the area between both grid blocks for the flow of fluids or heat has to be specified as well as the distance of the midpoints of each block to the common interface area.

3 Flow in an unstructured Grid

Every grid block can be assigned to a rock region which defines its porosity, permeability, relative permeability and capillary pressure. The fluid flow between two grid blocks is described by Equation 1. For each phase and also for heat flow a similar equation can be defined. The equations shown below are taken from the TOUGH2 user manual [1].

$$F_{\beta,m} = -k_{nm}\left[\frac{k_{r\beta}\cdot\rho_\beta}{\mu_\beta}\right]\cdot\left[\frac{p_{\beta,n}-p_{\beta,m}}{D_{nm}}-\rho_{\beta,nm}g_{nm}\right] \quad (1)$$

F is the flow of phase β between grid blocks m and n. k is the distance weighted permeability between the grid blocks n and m, k_r is the relative permeability, ρ the density and μ the viscosity, p is the phase pressure in the respective grid blocks, D the distance between the midpoints of the grid blocks and g is the gravitational acceleration vector.
By observing conservation of mass and adding a source and sink term one obtains a set of differential equations:

$$\frac{dM_n^\kappa}{dt} = \frac{1}{V_n}\sum_m A_{nm}F_{nm}^\kappa + q_n^\kappa \quad (2)$$

M is the mass of the component κ, t is time, V is the block volume, A is the area of the interface and q the mass flow rate.
The time discretization results in a set of coupled non linear algebraic equations:

$$R_n^{\kappa,k+1} = M_n^{\kappa,k+1} - M_n^{\kappa,k} - \frac{\Delta t}{V_n}\left\{\sum_m A_{nm}F_{nm}^{\kappa,k+1} + V_n q_n^{\kappa,k+1}\right\} = 0 \quad (3)$$

where the residuals R were introduced. The equation system is solved by Newton iteration. The iteration index p is introduced and the residuals are expanded in a Taylor series where only terms up to the first order were retained (Eq. 5).

$$R_n^{\kappa,k+1}(x_{i,p+1}) = R_n^{\kappa,k+1}(x_{i,p}) + \sum_i \left.\frac{\partial R_n^{\kappa,k+1}}{\partial x_i}\right|_p (x_{i,p+1}-x_{i,p}) + ... = 0 \quad (4)$$

$$-\sum_i \left.\frac{\partial R_n^{\kappa,k+1}}{\partial x_i}\right|_p (x_{i,p+1}-x_{i,p}) = R_n^{\kappa,k+1}(x_{i,p}) \quad (5)$$

The set of linear equations is then solved in the simulator. For more details see the TOUGH2 manual [1].

4 Converting Corner Point Grids

Corner point grids can be created in many software tools for geological modeling. The grid is output in a format that can be read by the simulation program. Usually information about the location of the corners of every grid block is written to an ASCII file. In addition property information for porosities, permeabilities, saturation etc. is included. This information has to be converted into a format readable by TOUGH2. Grid block information has to be entered in a specific format as shown in Table 1. Below the header in the first five columns the code name of a grid block is given, the next column gives the number of additional grid blocks having the same volume and belonging to the same reservoir region. Further an identifier has to be given for the rock region the block belongs to. Then the volume of the block is read and an interface area for heat exchange with semi-finite confining beds. A permeability modifier is optional. The last three columns contain coordinate information for post processing, i. e. programs to draw regular grid blocks.
As the code name for the grid blocks in the original TOUGH2 source code distribution only allows five characters it is not possible to store big corner point grids directly with the standard alpha-numerical characters, e. g. by assigning I, J and K values to different columns in the name. However if one wants to use large grids one can use the whole set of ASCII characters, ending up with $1.7\cdot10^{178}$ (= 5^{255}) grid blocks.
The following way was chosen to code the block names: In the first character of the code name string the I index of the corner point grid is stored starting from ASCII character 48, which is the character for the number one. In digits 2 and 3 the values for J and K are coded. If a I or J value is larger than 70 the value in the 4th or 5th digit is incremented and counting starts again, so that the value in digits 1 and 2 is kept smaller than the maximum of 128 allowed ASCII characters. This procedure permits to store several hundred blocks in I and J direction and up to 200 layers in K direction.
Further information about the block connections has to be supplied. This is done in a routine that reads the corner point input grid and checks every grid row and column for overlaps of cells that do not belong to the respective layer. Non-neighbor connections are identified and the areas to the connecting blocks are calculated. The results are written into a TOUGH2 input file as shown in Table 2.
The first two columns contain the 5-character code names of connecting grid blocks. The next three columns can contain information about similar connections that need not to be read. As it was chosen to supply all elements and connections this does not apply here and the values are zero. The 6th column can contain information about the direction of the connection, so that different permeabilities can be used, which are defined in the respective ROCKS table. The next two columns contain the distance of the grid block centers to the common interface

Table 1 **TOUGH2 input for grid blocks**

```
ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
11111     0    1ROC01   96436.9       0.0       1.0    -897.2     386.4    1021.6
21111     0    1ROC01   96577.4       0.0       1.0    -714.5     388.0    1019.0
31111     0    1ROC01   96717.9       0.0       1.0    -531.8     389.5    1016.3
41111     0    1ROC01   96858.5       0.0       1.0    -349.1     391.1    1013.6
51111     0    1ROC01   96999.0       0.0       1.0    -166.4     392.6    1011.0
61111     0    1ROC01   97139.5       0.0       1.0      16.4     394.2    1008.5
71111     0    1ROC01   97280.1       0.0       1.0     199.1     395.7    1005.7
81111     0    1ROC01   97423.9       0.0       1.0     381.8     397.3    1002.8
91111     0    1ROC01   97564.4       0.0       1.0     564.5     398.8    1001.3
:1111     0    1ROC01   97701.8       0.0       1.0     747.3     400.4     998.0
```

and the next column holds the size of the interface area. Then the cosine of the angle between the gravitational acceleration vector and the line between the two elements is given. The last column holds a value for a factor to calculate radiative heat transfer, which may be used in some special TOUGH2 modules and is not applied in the cases discussed here.

Table 2 **Input for block connections**

```
CONNE----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
1111121111    0    0    0    0        90.7      90.7     545.0  -0.01424       1.0
1111112111    0    0    0    0        88.8      88.8     548.2  -0.00012       1.0
1111111211    0    0    0    0         1.5       1.5   32145.6   1.00000       1.0
2111131111    0    0    0    0        90.7      90.7     544.1  -0.01435       1.0
2111122111    0    0    0    0        88.9      88.9     548.2   0.00059       1.0
2111121211    0    0    0    0         1.5       1.5   32192.5   1.00000       1.0
2111111111    0    0    0    0        90.7      90.7     544.1   0.01424       1.0
3111141111    0    0    0    0        90.7      90.7     543.4  -0.01444       1.0
3111132111    0    0    0    0        89.0      89.0     548.2   0.00111       1.0
3111131211    0    0    0    0         1.5       1.5   32239.3   1.00000       1.0
3111121111    0    0    0    0        90.7      90.7     543.4   0.01435       1.0
```

5 Properties

The properties needed for a TOUGH2 simulation are porosity, permeability, relative permeability and capillary pressure. These values have to be entered in the ROCKS section of the TOUGH2 input deck. In the original TOUGH2 source code 27 rock regions can be defined. A grid block can only have the properties of one rock region that has to be defined. On one hand, this is a limitation, but eventually it leads to a more consistent data set.

Reservoir properties coming from a geological modeling tool contain usually a wide range of porosity and permeability values. These values have to be reduced to the values defined by rock regions. In the preprocessing program this can be achieved by defining the number of rock regions to be used, and further specifying if either permeability or porosity should be used to define the rock regions. From the minimum and maximum values the spread can be calculated and each grid block can be assigned to an appropriate rock region.

Like all other input data required for a TOUGH2 simulation the rock property data are read in a free format from an input file using keywords (similar to other simulators, e. g. Eclipse). All input data are written from the pre-processing tool to the input file for TOUGH2. After the program has written the TOUGH2 input file it automatically starts the TOUGH2 simulation.

Table 3 **Input of rock properties**

```
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
ROC01    2  2600.000    0.20000 1.00E-013 1.00E-013 1.00E-013      0.10      2.00
     0.000     0.000      0.100     1.000     7.000     8.000     9.000
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
ROC02    2  2650.000    0.20000 1.00E-013 1.00E-013 1.00E-013      0.10      2.00
     0.000     0.000      0.100     1.000     7.000     8.000     9.000
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
ROC03    1  2700.000    0.20000 1.00E-013 1.00E-013 1.00E-013      0.10      2.00
     0.000     0.000      0.100     1.000     0.000     0.000     0.000
ROC04    2  2750.000    0.20000 1.00E-013 1.00E-013 1.00E-013      0.10      2.00
     0.000     0.000      0.100     1.000     0.000     0.000     0.000
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
ROC05    2  2800.000    0.20000 1.00E-013 1.00E-013 1.00E-013      0.10      2.00
     0.000     0.000      0.100     1.000     0.000     0.000     0.000
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
   1           0.000      0.100     0.200     0.300     0.400     0.500     0.600
```

Table 4 **Example for free format (unformatted) input in the pre-processing program**

```
ROCKS
1 5 1

2 2600 0.1 2
1E-7 1E-5 0.1 1 7 8 9
1 0 0.1 0.2 0.3 0.4 0.5 0.6
1 0 0.1 0.2 0.3 0.4 0.5 0.6

/

PARAM
-- NOITE etc. 5
8 1 999  99  1 /
--MOP 1 - 24
--2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
0 0 0 0 0 0 8 0 0 0  1  0  0  0  3  0  0  0  0  0  0  0  0  9 /
-- TEXP BE 2
7 8 /
-- Tstart etc. 8
```

6 Wells

The implementation of wells in TOUGH2 is not straight forward. Wells are defined as sources or sinks. Flow rates have to be provided in kg/s. It is also possible to flow wells at a constant well flowing pressure. To use wells in the usual petroleum reservoir engineering way, the input tables require further preparation. In the pre-processing described here well completions can be defined and the time dependent flow rates are translated into a TOUGH2 "GENER" data set.

Table 5 **Well flow rates as represented in a TOUGH2 "GENER" data set**

```
GENER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
33111B0001    0    0    0    5     COM32   0.00000   0.00000   0.00000
           0.0       345600.0      15811200.0      31708800.0
    94780800.0
         3.889          0.000           0.000           0.000
         0.000
    200000.000     200000.000      200000.000      200000.000
    200000.000
23111B0002    0    0    0    5     COM32   0.00000   0.00000   0.00000
           0.0       345600.0      15811200.0      31708800.0
    94780800.0
```

7 Post Processing

Reservoir simulators used in the oil and gas industry create output that can be used in post processors to generate line graphs and 3D displays of the simulation model. The TOUGH2 output cannot be used directly in post processor packages as they are typically used in petroleum reservoir engineering. It

Table 6 **Extra keyword to supply information about grid size etc. to TOUGH2**

```
PARAM----1-MOP* 123456789012345678901234----*----5----*----6----*----7----*----8
 8 1 999  99   100000080001000300000009                    7.000     8.000
     0.0008640000.000 86400.000 86400.00011111              9.810     0.000     1.000
           100009.0                 10.1                    10.2               25.0

SOLVR----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
3  Z1    001.000E+0001.000E-006

EXTRA----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    4    4    3    1    1 200036526.00
```

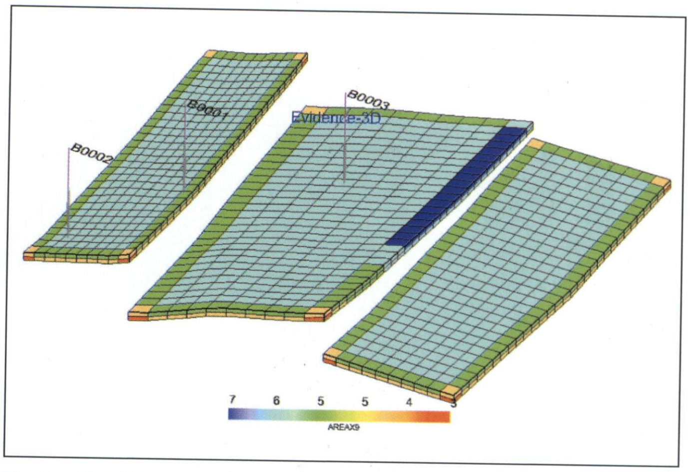


Fig. 4 **Simulation grid showing the number of neighbors for the layers 9 and 10**

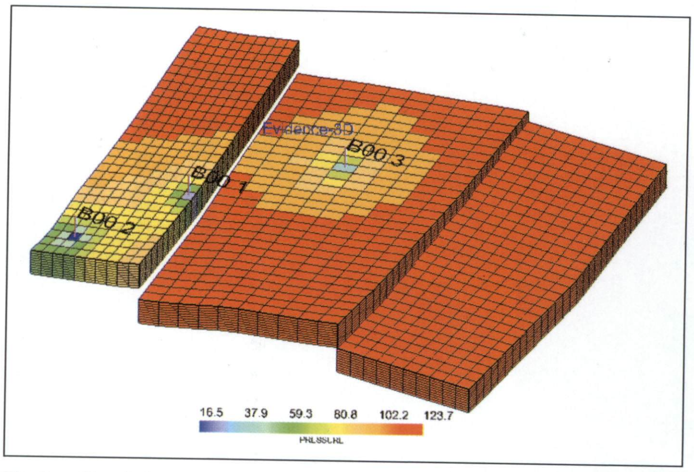


Fig. 5 **Simulation corner point grid with pressure [bar] during gas production**

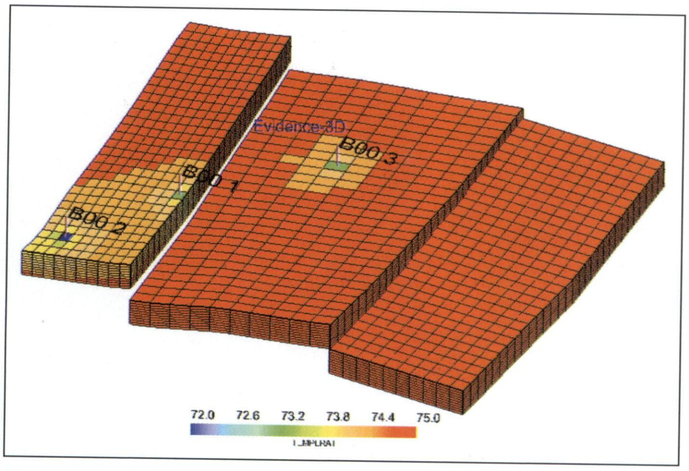


Fig. 6 **Temperature distribution during gas production [°C]**

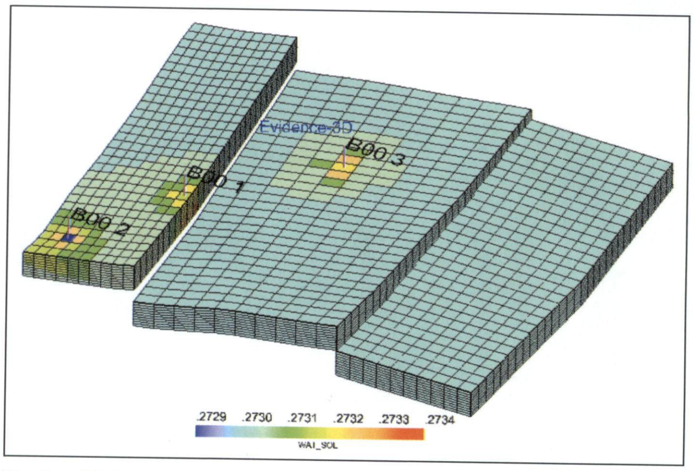


Fig. 7 **Distribution of salt concentration in the water phase [kg/kg]**

seems to be desirable to use the same tools for TOUGH2 output analysis as for the analysis of Eclipse output for example. This requires that the simulation results from TOUGH2 have to be written in a format readable by those tools, also because the output generated by TOUGH2 is rather limited. Eventually two types of data sets have to be written from TOUGH2, one is to give grid block properties at defined report times, the other is for the generation of line graphs of well and field production or pressure development.

This approach requires changes to the original TOUGH2 code. As TOUGH2 is delivered as Fortran source code, changes could be applied to the simulator. At first a new keyword has been introduced called EXTRA. The EXTRA section contains information of the grid dimensions and the start date of the simulation. Such information is needed to convert the coded block names back to block indices. TOUGH2 keeps track only of the past seconds since simulation start. Therefore the start date is required for Eclipse like output files.

The extra code that generates the output files is called in the TOUGH2 output subroutine. It generates four additional output files in BIG_ENDIAN mode, like the respective files in Eclipse. The base name for these files is supplied as a 3rd command line parameter, and the file extensions are ".T.SMSPEC", ".T.UNSMRY", ".T.INIT" and ".T.UNRST". The extra .T-extension was chosen to distinguish TOUGH2 output from Eclipse output. This approach requires only changes to the TOUGH2 code at three locations and does not harm the code execution.

The pre-processor generates grid files in the Eclipse GRID and EGRID format. These grids are used for the 3D display of the simulation results.

8 Example

In order to validate the concept, a simple example was calculated using the TOUGH2 equation of state EWASG. The model calculates the flow of water and NaCl and non-condensible gas (air, CO_2, CH_4, H_2 or N_2) phases and the heat transport. The grid model consists of 30x30x10 blocks with two faults. In Figure 4 the simulation grid is shown with the number of neighbors for each grid cell of the lowest two layers 9 and 10. At the corners a grid cell in the lowest layer has only three neighbors, in other layers grid cells have four neighbors. At a fault, cells may have more than six neighbors, if there is communication across the fault, or less than six neighbors, if there is no communication.

In Figure 5 the pressure distribution in the simulation grid is shown. As TOUGH2 is also simulating the transport of heat the temperature decrease around a production well is calculated. This temperature decrease is caused by the expansion of the gas. The heat transport into the matrix is calculated. The temperature distribution is shown in Figure 6.

Figure 7 shows the salt concentration in the water phase and Figure 8 the concentration of precipitated salt around the wells.

In Figure 9 an example for a line graph is given, showing the well bottom hole pressure, the development of the salt concentration in the water phase of a producing well is shown in Figure 10 and the concentration of water in the gas phase in Figure 11. As the results shown are from a restart run production begins at 250 h.

9 Conclusions

It was possible to convert corner point grids to the format necessary for the simulation in TOUGH2 so that also complex problems with faulted structures and property distributions can easily be simulated in TOUGH2.


WE TALK
RECIPS.
NEA GROUP
Authorized
OEM supplier
for reciprocating
compressor lines:
NEA GROUP
Esslingen
PPC
PENN
Process
TM Compressors
WURZEN
Demag
HALBERG
Linde
E&S
Borsig
(recips built in Berlin
up to the end of 1995)
KSB
COMPRESSORS
CP
Chicago
Pneumatic
NEUMAN & ESSER GROUP
▪ www.neuman-esser.com

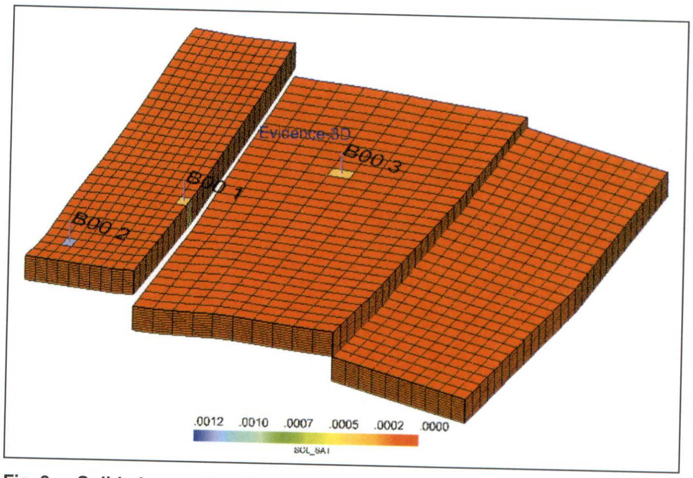


Fig. 8 **Solid phase saturation – Salt precipitation**

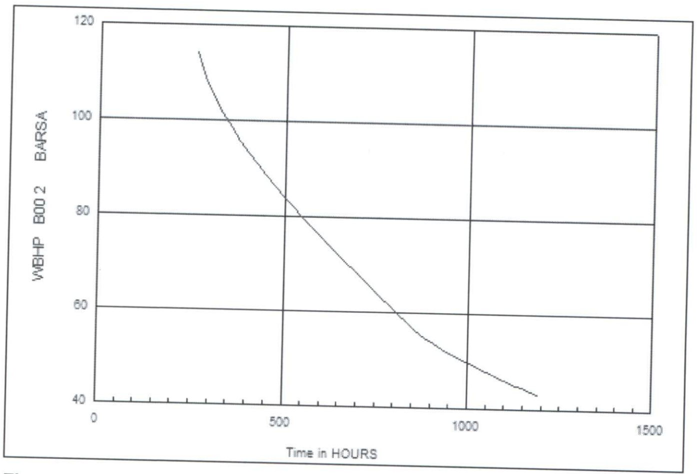


Fig. 9 **Line Graph with well bottom hole pressure**

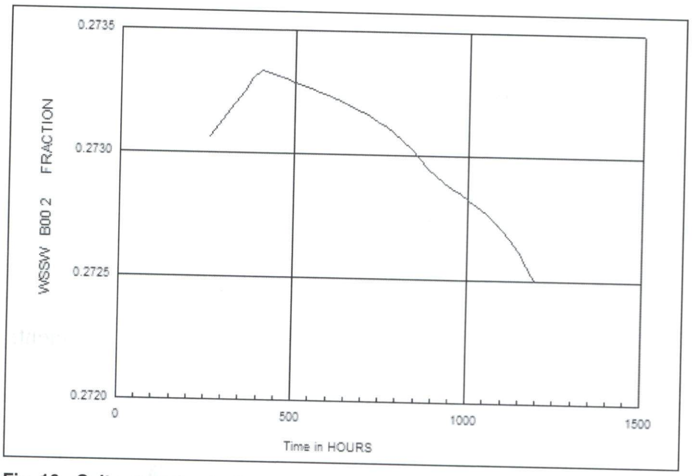


Fig. 10 **Salt saturation in the water phase of a producing well**

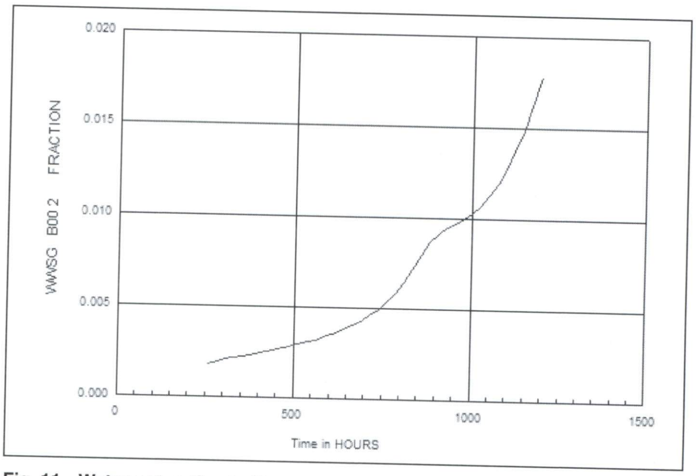


Fig. 11 **Water saturation in the gas phase (at bottom hole conditions) of a producing well**

This approach makes use of the advantage of unstructured grids to hydrocarbon reservoir simulation. Unstructured by definition handle the flow of fluids and heat correctly also across faults and in complicated structures. Also the flow in fractured reservoirs can be simulated reasonably.

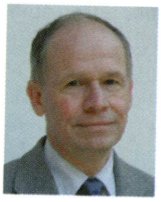
Simulation results from TOUGH2 can be written to binary files in the Eclipse "Standard-Format" so that they can be read by the software packages generally being used in the oil and gas industry. This applies to the 3D representation of the model as well as to line graphs.

Besides the simulation of the flow of water and gas some TOUGH2 models as EWASG (Water, Salt, Gas) handle also the transport of heat and the solution of salt in water. This model, originally designed for geothermal problems, is well suitable for simulating gas reservoirs and gas storage in porous media and salt caverns.

An example shows how the reservoir temperature is dropping around a production well due to the cooling by gas expansion and how this causes changes in salt concentration in the reservoir brine and precipitation of salt close to wells.

Literature

[1] Pruess, K., Oldenburg, C., Moridis, G.: "TOUGH2 User's Guide, Version 2.0, Earth Sciences Division, Lawrence Berkley National Laboratory, University of California, Berkley, California 94720.



Wolfgang Littmann, *independent oil and gas industry consultant: reservoir engineering and production geology, numerical reservoir simulation, logging and well testing, enhanced oil recovery. He holds an M.Sc. degree and Ph.D. in physics from the University of Hannover. Since 1977 in several positions in research and production in the German oil industry, consultant since 1994.*

Kristof Littmann *holds a degree in mechanical engineering from the RWTH Aachen. The master thesis (Dipl. Ing.) was on CFD modelling of convective heat flow. He is now with Benteler Automotive in Paderborn conducting studies on CFD modelling of exhaust systems.*

A 3D Finite Element Model for Simulating Hydraulic Fracturing Processes with Viscoelastic Reservoir Properties

By HAN WANG, HE LIU, XIAOZHOU ZHOU, HENGAN A. WU and XiuXi WANG*

Abstract

A 3D finite element model for simulating hydraulic fracturing processes is proposed with an ABAQUS code. Fluid-solid coupling and viscoelastic reservoir properties are considered in the model. A typical hydraulic fracturing process is simulated with the model and the obtained bottomhole pressure evolution is consistent with the data measured from field. The model is validated. The effect of viscoelasticity on fracture containment is simulated with the model and discussed. The results show that the viscoelastic model predicts smaller fracture height than that predicted by the elastic model, and the fracture height decreases as the viscoelastic parameters of the reservoir decreases.

1 Introduction

Hydraulic fracturing is one of the most important technologies for enhancing the productivity of reservoirs with poor permeability. The mechanism of hydraulic fracturing is complicated. Many numerical models in the literature are 2D or Pesudo-3D and the success of fully 3D hydraulic fracture models is limited [1]. For vertical fracture the fracture height containment is a decisive factor in assessing the success or failure of fracturing, and therefore it is essential in treatment design [2]. Changes in lithological parameters, such as least in-situ stress, elastic modulus etc, could influence the fracture configuration and control the fracture height [3, 4]. It has been pointed out that sliding at the interface between the barrier layer and the pay layer is another likely mechanism of fracture height containment [5]. Laboratory studies have shown that there is a time- and frequency-dependent behavior of reservoir formation, and these phenomena can be described using linear viscoelastic theory [6]. A new viscoelastic mathematical model is derived from the power-law equation and can be used to analyze the stress relaxation curves of different materials [7]. Studies also show that among the various models, the standard linear model is the best one for representing the viscoelasticity of rock [8]. But research on the effect of reservoir viscoelasticity on hydraulic fracturing has to our knowledge not been carried out.

A full 3D finite element model is proposed with an ABAQUS code. Fluid-solid coupling and reservoir viscoelasticity are considered in the model. A typical hydraulic fracturing process of a well in the Daqing Oilfield, China is simulated with the model and the obtained bottomhole pressure evolution is consistent with the field measured data. The verification of the model is approved. The effect of reservoir viscoelasticity on fracture height containment is simulated with a viscoelastic constitutive relation (standard linear model). The numerical results demonstrate that viscoelastic models of formation predict smaller fracture height than the elastic model. The simulation results also show that the fracture height decreases as the viscoelastic parameters of the formation decreases. Thus, the viscoelastic property is another lithological factor which could control fracture height.

2 Mathematical Model

2.1 Coupled fluid-solid equations

The stress equilibrium equation of the solid formation is expressed in the following form [9]

$$\nabla \cdot \sigma + f = 0 \tag{1}$$

where σ is the total stress tensor in the formation, and f is the body force vector.

The mass conservation equation of porous fluid in the reservoir is written as [9]

$$\frac{\partial}{\partial t}(\rho_w \phi) + \nabla \cdot (\rho_w \phi v_w) = 0 \tag{2}$$

where ρ_w, ϕ *and* v_w are the density of porous fluid, the porosity of formation and the seepage flow velocity vector respectively.

According to Darcy's law, the velocity of seepage flow is proportional to the gradient of porous pressure [10, 11], that is

$$v_w = -\frac{1}{\phi g \rho_w} k \cdot (\nabla p_w - \rho_w g) \tag{3}$$

where k, p_w and g represent the hydraulic conductivity tensor [12], the porous pressure and the gravity acceleration vector respectively.

The relationship between formation stress and seepage pressure can be expressed by the effective stress principle [13] as following

$$\bar{\sigma} = \sigma - p_w I \tag{4}$$

where $\bar{\sigma}$ is the effective stress matrix.

2.2 Viscoelastic constitutive model

The standard linear model is commonly adopted in describing the viscoelasticity of rock [8]. It is composed of a Hooke spring and a Kelven body in series as shown in Figure 1. The standard linear model is used to present the relation between deviatoric stress and strain tensors. It is assumed that the volume stress is proportional to the volume strain as in the linear elastic model [14], i. e.

$$s = 3Ke \tag{5}$$

where s, K and e are the volume stress, the bulk modulus and the volume strain respectively.

The relaxation modulus of the standard linear model is [15]

$$G(t) = 2G_1\left(1 - \frac{G_1}{G_1 + G_2}(1 - e^{-\frac{t}{\tau}})\right) \tag{6}$$

where G_1 and G_2 are the instantaneous shear modulus of the material and shear modulus of Kelven body respectively. τ is the relaxation time defined as [15]

$$\tau = \frac{\eta}{G_1 + G_2} \tag{7}$$

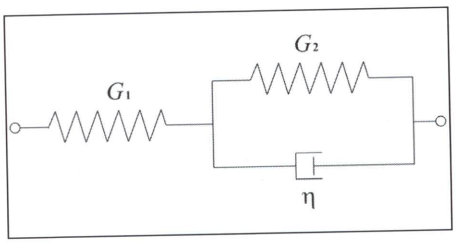


Fig. 1 Standard linear body

* Han Wang, XiaZhou Zhou, Hegan A. Wu, XiuXi Wang, Department of Modern Mechanics, University of Science and Technology of China, Hefei, Anhui, China; He Liu, Department of Modern Mechanics, University of Science and Technology of China, Hefei, Anhui; Research Institute of Petroleum Exploration & Development, PetroChina, China (E-mail: heliu05@ustc.edu.cn; liuhe@petrochina.com.cn).

0179-3187/12/IV