NUMERICAL EXPERIMENTS WITH THE SIMULATION OF NATURAL STATES OF GEOTHERMAL RESERVOIRS

Z. YANG, D.P. BULLIVANT, G.A. ZYVOLOSKI[1] and M.J. O'SULLIVAN

Department of Engineering Science
University of Auckland
and
[1]Los Alamos National Laboratory

SUMMARY

The first stage of a computer modelling study of a geothermal reservoir is usually the simulation of the pre-exploitation or natural state. For large complex models this process may consume a considerable amount of computer time. Numerical experiments aimed at speeding up simulations of natural states are described here. The most successful technique developed is the use of a sequence of graded computational grids, from come to fine, with interpolation of the pressures and temperatures (or saturations) from one grid to the next.

1. INTRODUCTION

The first stage of computer modelling of a geothermal reservoir should be the simulation of the pre-exploitation or natural state (see O'Sullivan, 1985, Pruess, 1990 and Bodvarsson et al., 1986). This enables the testing and refining of conceptual models of the system. In particular theories about the permeability structure of the reservoir can be tested by comparing the temperature distribution and surface discharge of heat given by the model with field measurements.

The usual procedure for simulating the natural state of a geothermal system is to run a transient simulation for a very large (simulated) time until a steady state is obtained. In this process larger and larger time steps are used until finally a very large total simulation time is achieved and pressure and temperature (or saturations) do not change significantly from one time step to the next.

For relatively small and simple models (say, 100 to 200 blocks or elements and a small boiling zone), twenty to thirty time steps are usually sufficient to obtain a steady state, with a very large final time step (say 10^{14} seconds). However for large complex models, with large boiling zones and with gas present, convergence to a steady state may take several hundred time steps. For these models re-running to a new steady state, after minor modifications to the model have been made, requires a large number of time steps. If several modifications of the model structure are investigated and re-runs to a steady state are carried out in each case then a very large amount of computer time will be required.

The purpose of the work described here was to investigate ways of speeding up the computation of natural states of geothermal fields. Two aspects of the problem were considered. Firstly the procedure for time-step control was investigated and various modifications tried. These have not produced very significant improvements so far. Secondly a type of cascade or 'multi-grid' procedure was adopted with a sequence of grids from coarse, with few blocks, up to fine, with many blocks, was used. On a two-dimensional test problem this method reduced the total computation time by up to 60%.

All simulations were carried out using a modified version of the MULKOM code (see Pruess, 1982 and Bullivant, 1990).

2. THEORY

The discussion here is based on the integrated finite difference technique developed at Lawrence Berkeley Laboratory and implemented in the SHAFT79, MULKOM, TOUGH and TOUGH2 codes (see Pruess 1988,1991). It is assumed that the region of interest is divided up into blocks or elements. The ith block has volume V_i and is connected by an interface, with area a_{ij}, to the jth block. This formulation allows for an irregular block structure but includes more regular block structures such as rectangular blocks or polar coordinate systems as special cases. Here p_i^n and T_i^n are used to represent pressures and temperatures in the ith block at the end of the nth time step. The nth time step is of duration Δt_n.

Most geothermal simulation techniques are based on two common ideas:

(i) Difference equations are fully implicit with all mass and energy fluxes evaluated at the new time level.

(ii) Upstream weighting is used to calculate interface quantities.

The procedure discussed here is block-centred for pressures and temperatures while fluxes are calculated at block boundaries.

Using the notation above the discrete mass balance equation can be written:

$$V_i\left(A_{mi}^{n+1} - A_{mi}^{n}\right) = -\sum_j a_{ij} F_{mij}^{n+1} \Delta t_{n+1} + Q_{mi}^{n+1} \Delta t_{n+1} \qquad (1)$$

Here F_{mij}^{n+1} is the mass flux from block i to block j evaluated at the end of the (n+1)th time step. Similarly Q_{mi}^{n+1} is the mass production from block i evaluated at the end of the (n+1)th time step (positive for injection).

Similarly the discrete energy equation is

$$V_i\left(A_{ei}^{n+1} - A_{ei}^{n}\right) = -\sum_j a_{ij} F_{eij}^{n+1} \Delta t_{n+1} + Q_{ei}^{n+1} \Delta t_{n+1} . \qquad (2)$$

Here F_{eij}^{n+1} and Q_{ei}^{n+1} are defined as for the mass equation above. The mass and energy accumulation terms are defined as follows:

$$A_{mi}^{n+1} = \phi_i \left(S_\ell \rho_\ell + S_v \rho_v\right)_i^{n+1} \qquad (3)$$

$$A_{ei}^{n+1} = (1 - \phi_i)\, \rho_{ri}\, C_{ri}\, T_i^{n+1} + \phi_i (\rho_\ell S_\ell u_\ell + \rho_v S_v u_v)_i^{n+1} \qquad (4)$$

Here ϕ is the porosity, ρ_r is the density and C_r is the specific heat of the rock matrix. For the fluid: ρ_ℓ and ρ_v are the densities, S_ℓ and S_v are the saturations and u_ℓ and u_v are the internal energies, with the subscripts ℓ and v referring to liquid and vapour respectively.

In order to calculate the mass flux it is divided into the two separate phases with

$$F_{mij}^{n+1} = F_{m\ell ij}^{n+1} + F_{mvij}^{n+1}$$

Then a two-phase version of Darcy's Law is used in the form

$$F_{m\ell ij}^{n+1} = -\left(\frac{k k_{r\ell}}{\nu_\ell}\right)_{ij}^{n+1} \left[\frac{p_j^{n+1} - p_i^{n+1}}{d_{ij}} - \rho_{\ell ij}^{n+1} g_{ij}\right] \qquad (5)$$

$$F_{mvij}^{n+1} = -\left(\frac{k k_{rv}}{\nu_v}\right)_{ij}^{n+1} \left[\frac{p_j^{n+1} - p_i^{n+1}}{d_{ij}} - \rho_{vij}^{n+1} g_{ij}\right] \qquad (6)$$

Here k is the rock permeability, $k_{r\ell}$ and k_{rv} are the relative permeabilities, ν_ℓ and ν_v are the kinematic viscosities and d_{ij} is the distance between block centres. The energy flux is then calculated using:

$$F_{eij}^{n+1} = h_{\ell ij}^{n+1} F_{m\ell ij}^{n+1} + h_{vij}^{n+1} F_{mvij}^{n+1} - K_{ij}^{n+1} \frac{T_j^{n+1} - T_i^{n+1}}{d_{ij}} \qquad (7)$$

Here h_ℓ and h_v are the enthalpies of the liquid phase and vapour phase, respectively. There are several terms in (5), (6) and (7) whose calculation requires further explanation. The gravity term g_{ij} is the component of gravity acting through the interface. For example, $g_{ij} = 0$ for two blocks horizontally adjacent, and $g_{ij} = g$ for two blocks with block i vertically above block j. The interface densities in the "weight" terms are evaluated using

$$\rho_{\ell ij}^{n+1} = \frac{1}{2}\left(\rho_{\ell i}^{n+1} + \rho_{\ell j}^{n+1}\right) \qquad (8)$$

$$\rho_{vij}^{n+1} = \frac{1}{2}\left(\rho_{vi}^{n+1} + \rho_{vj}^{n+1}\right) \qquad (9)$$

The interblock distance d_{ij} is the sum of the distances d_i and d_j from the centres of the ith and jth block to their connecting interface respectively. The interface permeabilities and conductivities are calculated using harmonic weighting and usually they are assumed to be independent of pressure and temperature and therefore need to be evaluated only once at the beginning of the simulation using

$$\frac{1}{k_{ij}} = \left(\frac{d_i}{k_i} + \frac{d_j}{k_j}\right) / d_{ij} . \qquad (10)$$

The most important aspect of the interface calculations is the upstream weighting of the mobilities and enthalpies.

For the liquid phase, either

$$\left(\frac{k_{r\ell}}{\nu_\ell}\right)_{ij}^{n+1} = \left(\frac{k_{r\ell}}{\nu_\ell}\right)_{i}^{n+1}, \quad \text{for } G_\ell^{n+1} < 0, \tag{11a}$$

or

$$\left(\frac{k_{r\ell}}{\nu_\ell}\right)_{ij}^{n+1} = \left(\frac{k_{r\ell}}{\nu_\ell}\right)_{j}^{n+1}, \quad \text{for } G_\ell^{n+1} < 0, \tag{11b}$$

where

$$G_\ell^{n+1} = \frac{p_j^{n+1} - p_i^{n+1}}{d_{ij}} - \rho_{\ell ij}^{n+1} g_{ij}. \tag{12}$$

This condition ensures that the mobility at the interface is evaluated using the properties of the fluid at the block which the fluid is leaving. Separate upstream weighting is used for the vapour phase with either:

$$\left(\frac{k_{rv}}{\nu_v}\right)_{ij}^{n+1} = \left(\frac{k_{rv}}{\nu_v}\right)_{i}^{n+1}, \quad \text{for } G_v^{n+1} < 0, \tag{13a}$$

or

$$\left(\frac{k_{rv}}{\nu_v}\right)_{ij}^{n+1} = \left(\frac{k_{r\ell}}{\nu_\ell}\right)_{j}^{n+1}, \quad \text{for } G_v^{n+1} < 0, \tag{13b}$$

where

$$G_v^{n+1} = \frac{p_j^{n+1} - p_i^{n+1}}{d_{ij}} - \rho_{vij}^{n+1} g_{ij} \tag{14}$$

Similarly

$$h_{\ell ij}^{n+1} = \begin{cases} h_{\ell i}^{n+1} & \text{for } G_\ell^{n+1} < 0 \\ h_{\ell j}^{n+1} & \text{for } G_\ell^{n+1} > 0 \end{cases} \tag{15}$$

and

$$h_{vij}^{n+1} = \begin{cases} h_{vi}^{n+1} & \text{for } G_v^{n+1} < 0 \\ h_{vj}^{n+1} & \text{for } G_v^{n+1} > 0 \end{cases} \tag{16}$$

The mass and energy balance equations, (1) and (2), together with (3) - (16) above are then solved at each time step. In addition steam table data are required to calculate the secondary parameters ρ_ℓ, ρ_v, u_ℓ, u_v, ν_ℓ and ν_v from the *primary* variables. In the MULKOM code p and T (or S) are used as *primary* variables.

3. NEWTON-RAPHSON METHOD

When (1) and (2) are written out for all blocks in the reservoir model they form a system of nonlinear equations for the unknowns p_i^{n+1}, T_i^{n+1} (or S_{vi}^{n+1}). They *can* be written in the form

$$\mathbf{f(x)} = \mathbf{0} \tag{17}$$

where

$$\mathbf{x} = \begin{bmatrix} p_1^{n+1} \\ T_1^{n+1} \\ \vdots \\ p_M^{n+1} \\ T_M^{n+1} \end{bmatrix} \tag{18}$$

and

$$f_{2i-1} = V_i \left(A_{mi}^{n+1} - A_i^n\right) + \sum_j a_{ij} F_{mij}^{n+1} \Delta t_{n+1} - Q_{mi}^{n+1} \Delta t_{n+1} \tag{19}$$

$$f_{2i} = V_i \left(A_{ei}^{n+1} - A_{ei}^n\right) + \sum_j a_{ij} F_{eij}^{n+1} \Delta t_{n+1} - Q_{ei}^{n+1} \Delta t_{n+1} \tag{20}$$

Equations of the form shown in (17) can be solved using the Newton-Raphson method. This involves an iterative procedure for calculating successive approximations $\mathbf{x}_k$ using the Taylor series approximation:

$$\mathbf{f}(\mathbf{x}_{k+1}) = \mathbf{f}(\mathbf{x}_k) + J_k (\mathbf{x}_{k+1} - \mathbf{x}_k).$$

Here J_k is a Jacobian matrix given by

$$J = \left[\frac{\partial f_i}{\partial x_j}\right]$$

evaluated at $\mathbf{x}_k$. If it is assumed that $\mathbf{x}_{k+1}$ is a solution then $\mathbf{f}(\mathbf{x}_{k+1}) = 0$ and we obtain

$$\mathbf{x}_{k+1} = \mathbf{x}_k - J_k^{-1} \mathbf{f}(\mathbf{x}_k). \tag{21}$$

As the mass and energy balance equations and f_i are very complex the derivatives required to evaluate J are not easy to calculate. In MULKOM difference approximations to the derivatives are used. In this case (21) may not have the quadratic convergence property near the solution provided by exact derivatives.

4. TIME STEP CONTROL

The problem with the iterative procedure given by (21) is that it may converge slowly, or not at all if the initial estimate x_0 is not accurate. In MULKOM the solution at the previous time step is used as the first estimate. If it is found that (21) does not converge then the time step Δt_{n+1} can be made smaller (say divided by 4 or 5) and (21) applied again. In practice this process is found to work well.

At each iteration (21) gives a system of linear equations to be solved. In the standard versions of SHAFT79, MULKOM or TOUGH2 the Harwell sparse solver MA28 is used. In the version of MULKOM used at the University of Auckland a fast solver which uses the conjugate gradient method is used. This solver which was developed at Los Alamos National Laboratory (see Zyvoloski, 1991) speeds up MULKOM considerably (see Bullivant, 1990).

The MULKOM code includes the option of increasing the size of the time step if the Newton-Raphson (N-R) iterations (21) converge very rapidly. In a typical simulation of the natural state of a geothermal field the number of iterations at each time step varies as the simulation proceeds and the size of the time step may oscillate, both decreasing and increasing, before ultimately increasing monotonically up to a very large value (say 10^{14} when a steady state has been reached.

In Fig. 1 a plot of simulation time versus time-step number is shown for one of the test problems discussed below. The variation in time-step size during the calculation and the oscillation in the time-step size as it is cyclically decreased and then increased again is clear. For each time-step a number of N-R iterations may be required, each of which involves the solution of a large linear system of equations. The solution of the linear equations is the most time consuming part of the computation and therefore the total number of N-R iterations is a good measure of the efficiency of the calculation.

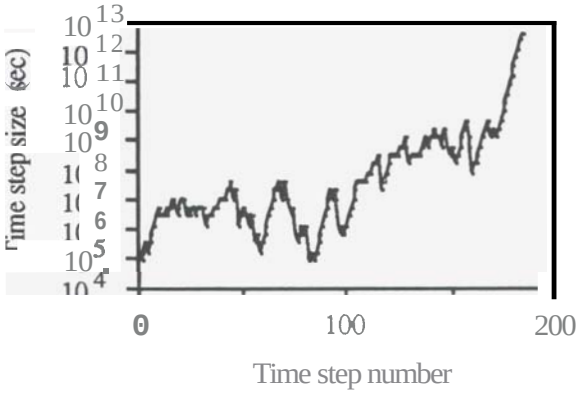

Figure 1. Time-step history for a typical natural state simulation.

The first part of this study was aimed at decreasing the total number of N-R iterations required to obtain a steady state. It is clear from Fig.1 that, if at each time-step exactly the right length of step was used, the time consuming oscillations could be eliminated and many N-R iterations avoided.

The parameters involved in the N-R process and time-step control are as follows:

(i) MAXIT, the maximum number of N-R iterations permitted each time-step before the time-step is decreased (default in MULKOM is 8).

(ii) TOL, the convergence tolerance. That is the maximum size allowed for the residuals. The residuals are the size of the errors in the mass and energy balance equations, ie $|f_i(x_{n+1})|$ (default in MULKOM is 10^{-6}).

(iii) REDLT, the divisor for the time-step when the N-R procedure fails to converge (default in MULKOM is 5).

(iv) MUL, the multiplier for the time-step when the N-R procedure converges quickly (say in four or fewer iterations) (default in MULKOM is 2).

(v) EPS, the small increment added to the primary variables in order to calculate numerical derivatives for the Jacobian matrix, J (default in MULKOM is 10^{-8}).

The basic time-step control strategy used in MULKOM is to allow the N-R procedure to run until the maximum value of the residuals is less than the tolerance (TOL) or until MAXIT iterations have been carried out. If MAXIT iterations do not produce convergence then the time-step is divided by REDLT and the process is repeated. If the convergence is obtained in 4 or less N-R iterations the next time step is increased by the factor MUL (ie $\Delta t_{n+1} = \text{MUL} \times \Delta t_n$).

5. RESULTS FOR ALTERNATIVE TIME-STEP STRATEGIES

Various strategies were tried with the aim of improving on the normal MULKOM procedure. They were all tested on a simple problem used earlier by one of the authors (OSullivan, 1984) to study two-phase convection. The test problem is a two-dimensional vertical slice 100m thick, 1000m long and 1000m deep. It was divided into a 20 x 20 grid of equal sized blocks. The top boundary is held fixed at atmospheric conditions, with a pressure of 1 bar and a temperature of 20°C, while the base is heated over 50% of its length with a 6MW input of heat. Other reservoir parameters are listed in Table 1.

Table 1. Model Parameters for 2-D Test Problem

Porosity	0.20
Specific heat or rock	900 J/kg.K
Conductivity of rock	2.5 W/m.K
Density of rock	2500 kg/m3
Relative permeabilities	Corey (residuals .3, .0)

The heat input is sufficient to cause a significant boiling zone in the natural state.

The natural state simulation was initiated with an approximate steady state obtained from interpolation of results from a 10 x 10 grid. The plot of simulation time versus time-step number has already been shown in Fig. 1 for the standard MULKOM procedure. A total of 162 time-steps were required and there were 37 failures where the time-step was too large to allow convergence of the N-R iterations. An examination of the maximum residuals obtained during these abortive time-steps showed that in many cases it was obvious after 4 or 5 iterations, rather than 8, that convergence would not be obtained. To try and avoid these unnecessary N-R iterations the following strategy was tested.

Strategy I
(a) If the residual *increases* monotonically for the first three N-R iterations then stop.
(b) Or if residual 5 is greater than residual 4 then stop.

This strategy had the desired effect of decreasing the number of time-step failures from 37 to 24 (see Table 2) but unfortunately the number of time-steps increased from 162 to 176 and the total number of N-R iterations increased from 1144 to 1208.

Table 2. Comparison of Time-step Control Strategies on the 2-D Test Problem (20 x 20 grid)

Strategy	No. Time-steps	N-R Iterations	Time-step Failures
Standard	162	1144	37
I	176	1208	24
II	164	1098	16
III	166	1105	16

Further investigation of the detailed behaviour of the pressures, temperature and saturations during each time-step revealed that most time-step failures occurred when phase changes were taking place during the N-R process. Therefore a further strategy was investigated where the number of phase-changes occurring at each N-R iteration were counted and this number was used to decide when to decrease the time-strep as follows:

Strategy II
Weight the number of phase changes at each N-R iteration as follows and if the sum of weights exceeds 10 then stop.

It. No.	1	2	3	4	5	6	7	8
Weight	0	1	2	3	4	11	11	11

As shown in Table 2 this strategy produced an improved performance in both the number of time-step failures and the total number of N-R iterations.

For some of the 16 time-step failures produced by strategy 11 the residual was decreasing monotonically at iteration 8 but was not quite below TOL. An extra test was added to allow for this as follows:

Strategy III
As for II above but in addition if residuals are monotonically decreasing then allow MAXIT to increase from 8 to 9.

As shown in Table 2 this strategy produced slightly worse results than 11.

At this stage it was decided that further more complex strategies aimed at decreasing the number of time-step failures were not worth investigating. The improvement in the total number of N-R iterations from the standard MULKOM strategy to strategy 111 was only 46 out of 1144 even though the number of time-step failures decreased from 37 to 16. It seems that even if a strategy could be found so that the remaining 16 time-step failures could be eliminated it would not produce large savings in the total number of N-R iterations.

Some further parameter variations were considered. First the time-step divisor (REDLT) was changed. As Table 3 shows the efficiency of the calculation is not very sensitive to the choice of this parameter. The value REDLT = 4 gave the best result. Similarly the time-step multiplier (MUL) was varied. The results showed that the speed of calculation is not influenced very strongly by the choice of MUL.

Table 3. Variation of the Time-step Reduction Factor (REDLT)

REDLT	Time-steps	N-R Iterations
2.0	135	1218
3.0	152	1149
4.0	162	1144
5.0	170	1154

Lastly the parameter EPS, used in calculating the numerical derivatives in the Jacobian, J_k, was varied. As Table 4 shows the default value for MULKOM of 10^{-8} produced the best result.

Table 4. Variation in the Increment for Calculating Derivatives

EPS	Time-steps
10^{-6}	178
10^{-7}	180
10^{-8}	170
10^{-9}	183

Some methods for optimising the choice of time-step (see Sammon and Rubin, 1983, for example) were investigated but did not appear relevant to geothermal problems where the speed of approach to steady-state is controlled mainly by nonlinear processes such as the rearrangement of the boiling zone.

It appears that sophisticated time-step control or modification of the N-R parameters have limited potential for speeding up natural state calculations.

6. CASCADE TECHNIQUES

The second stage of this investigation involved a type of cascade or 'multi-grid' approach. For the test problem this involved the following steps

(a) Solve for a steady state using a 5 x 5 grid starting from cold conditions (that is: a temperature of 20°C everywhere and hydrostatic pressures).

(b) Interpolate the 5 x 5 steady-state on to a 10 x 10 grid and use as initial data for a steady-state simulation on the 10 x 10 grid.

(c) Interpolate the 10 x 10 steady-state on to a 20 x 20 grid and use as initial data for a steady-state simulator on the 20 x 20 grid.

The results for the test problem, with various amounts of heat injected at the base, are shown in Table 5. The total computation time for the cascade approach described above is compared with a steady-state calculation on the fine, 20 x 20, grid starting fiom a cold initial state. One case where CO_2 gas is also injected at the base is also included. In all cases the cascade approach produces a speed up factor of around 2.

Table 5. Comparison of Cascade Results for the 2-D Test Problem (CPU seconds given)

Heat	5x5	10x10	20x20	Total	Saving
6MW	-		43059	43059	-
+CO_2	334	3256	19869	23459	45.5%
6MW	-		35912	35912	
	266	1422	15796	17484	51.3%
2MW	-		7722	7722	
	50	235	2627	2912	62.2%
1MW	-		3602	3602	
	28	123	1123	1274	64.6%
.5MW	-		1689	1689	-
	27	103	654	784	53.6%

The most time consuming part of the calculation is the readjustment of the two-phase region. As shown in Fig. 2 the two-phase regions for the three grids are similar but not identical and considerable computational effort, with quite small time-steps, is required to readjust the boiling zone in each successively finer grid.

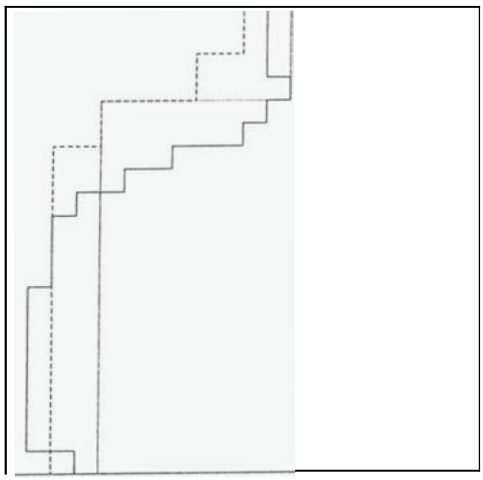

Figure 2. Two-phase region for 2-D test problem with 6 MW heat input. ····· 5x5 grid - - - - 10x10 grid —— 20x20 grid

In order to try to obtain a further speeding up of the calculation intermediate size grids, sub-divided in only one direction, were introduced. The results are given in Table 6 and show that intermediate subdivision in the horizontal direction is worthwhile.

Table 6. Comparison of Various Grid Subdivision Strategies, 6 MW case for 2-D test problem (CPU seconds given)

5 x 5	5 x 10	10 x 5	10x10	10x20	20x10	20x20	Total
						35912	35912
266			1422			15796	17484
266	-	82	1230	-	1522	19204	22124
266	337	-	814	5554	-	9356	16327

The technique was also tested on a three-dimensional problem. A cube 1000m x 1000m x 1000m is heated over a quarter of its base while the top is maintained at atmospheric conditions. The reservoir parameters used are the same as for the 2-D problem (see Table 1). A heat input of 40 MW was used. This value ensured that there was a boiling zone in the natural state. Three grids were used: 4 x 4 x 4, 8 x 8 x 8 and 16 x 16 x 16. Again a multi-grid or cascade calculation was compared with the use of the fine grid alone. The results are shown in Table 7. The number

of time-steps spent on the very fine grid, if the calculation is started from cold on the fine grid, is approximately 3 times that required as part of the cascade calculation.

Table 7. Comparison of Cascade Results for the 3-D Test Problem (number of time-steps given)

4x4x4	8x8x8	16x16x16
-		1251
25	1634	405

The scheme discussed here is not a true multigrid method (see McCormick, 1987, for example). For linear problems, such as isothermal groundwater modelling, multigrid methods involve cycling between coarse and fine grids using relaxation sweeps on the fine grid with an initial estimate obtained by interpolating a coarse grid solution. The coarse grid problem is solved with corrections added which are calculated using the fine grid solution. In the present work only the interpolation step from coarse to fine is used. Even for the nonlinear problem considered here the full multigrid method may be useful and is being investigated further.

7. DISCUSSION

The cascade approach described above was successful in speeding up natural state calculations. A speed-up factor of 2 was obtained for the 2-D test problem and a speed-up factor of 3 was obtained for the 3-D test problem. The implementation of the grid subdivision and interpolation was straightforward for the geometrically simple test problems considered here. For more complex models of real geothermal fields the construction of a sequence of coarse to fine grids will be more difficult.

The speed-up factors obtained here are useful but are not as large as was hoped for. Order of magnitude speed-up factors or better are desirable to cut down the large computational task required to obtain a good model of the natural state of a geothermal field. Research is still proceeding on the problem.

8. REFERENCES

Bodvarsson, G.S., Pruess, K. and Lippmann, M.J. (1986). Modeling of geothermal systems. J. *Petroleum Technology*, 38, p. 1007-1021.

Bullivant, D.P. (1990). Making MULKOM/TOUGH faster and easier to use. Proc. TOUGH workshop, Lawrence Berkeley Laboratory, Berkeley, California.

McCormick, S.F., ed. (1987), *Multigrid Methods*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania.

O'Sullivan, M.J. (1984). Convection with boiling in a porous layer. In: *Convectiveflows* in porous *media*, Proc. of a seminar organised by DSIR and CSIRO Institute of Physical Sciences, Wairakei, New zealand, Wooding, R.A. and White, I. *(eds)*, DSIR Science Information Publishing Centre.

OSullivan, M.J. (1985). Geothermal reservoir simulation. *Energy Research*, 9, p. 319-332.

Pruess, K. (1983). Development of the general purpose simulator MSJLKOM. Annual Report 1982, Earth Sciences Division, Lawrence Berkeley Laboratory, *report* LBL-15500.

Pruess, K. (1988). SHAFT, MULKOM, TOUGH A set of numerical simulators for multiphase fluid and heat flow. *Geotermia, Rev. Mex. Geoenergia*, 4(1), p. 185-202 (also Lawrence Berkeley Laboratory, report LBL-24430).

Pruess, K. (1990). Modeling of geothermal resrvoirs: fundamental processes, computer simulation and field applications. *Geothermics*, 19(1), p. 3-15.

Pruess, K. (1991). TOUGH2 - A general-purpose numerical simulator for multiphase fluid and heat flow. Earth Sciences Division, Lawrence Berkeley Laboratory, report LBL-29400.

Sammon, P.H. and Rubin, B. (1983). Practical control of timestep selection in thermal simulation. SPE 12268, paper presented at Reservoir Simulation Symposium, San Francisco, California.

Zyvoloski, G.A., Dash, Z. and Kelkar, K. (1991). FEHMN 1.0: Finite element heat and mass transfer code. Los *Alamos* National Laboratory, LANL report 12062.