SAND--97-2542C
SAN097-2542C
CONF-980125--

A TWO-PHASE THERMAL MODEL FOR SUBSURFACE TRANSPORT ON MASSIVELY PARALLEL COMPUTERS

M. J. Martinez[1] and P. L. Hopkins[2]

ABSTRACT

This paper presents an unstructured grid numerical algorithm for subsurface transport in heterogeneous porous media implemented for use on massively parallel (MP) computers. The mathematical model considers nonisothermal two-phase (liquid/gas) flow, including capillary pressure effects, binary diffusion in the gas phase, conductive, latent, and sensible heat transport. The Galerkin finite element method is used for spatial discretization, and temporal integration is accomplished via a predictor/corrector scheme. Message-passing and domain decomposition techniques are used for implementing a scalable algorithm for distributed memory parallel computers. Illustrative applications are shown to demonstrate capabilities and performance.

INTRODUCTION

Many research activities in subsurface transport require the numerical simulation of multiphase flow in porous media. This capability is critical to research in environmental remediation (e.g. contaminations with dense, non-aqeous-phase liquids), nuclear waste management, reservoir engineering, and to the assessment of the future availability of groundwater in many parts of the world. Scientific advancements in each of these areas could benefit from a high-performance numerical simulation capability. This paper presents an unstructured grid numerical algorithm for subsurface transport in heterogeneous porous media implemented for use on massively parallel (MP) computers via message-passing and domain decomposition techniques. Among the primary objectives of this research were to investigate the use of MP computing for general multiphase systems (which involve such complications as phase appearance and disappearance), and in particular, to develop scalable algorithms for general unstructured grids. The numerical platform which is presented provides an excellent base for new and continuing research in areas of current interest to the geoscience community.

MATHEMATICAL FORMULATION

The governing equations for nonisothermal multiphase flow in porous media are statements of mass balance of water and air, over both liquid and gas phases, and a statement of energy balance, also over both phases [1,2]. The canonical form of this coupled system of balance equations is given by

$$\frac{\partial}{\partial t}\begin{bmatrix} d_w \\ d_a \\ e \end{bmatrix} + \nabla\bullet\begin{bmatrix} \boldsymbol{F}_w \\ \boldsymbol{F}_a \\ \boldsymbol{q} \end{bmatrix} = \begin{bmatrix} Q_w \\ Q_a \\ Q_e \end{bmatrix}, \tag{1}$$

where subscripts w, a and e denote water, air and energy. The bulk mass and energy densities are given, respectively, by,

$$\begin{aligned} d_\alpha &= \phi \sum_{\beta = l,g} Y_{\alpha\beta}\rho_\beta S_\beta, \qquad \alpha = w, a \\ e &= (1-\phi)\rho_s e_s + \phi(S_l\rho_l e_l + S_g\rho_g e_g) \end{aligned} \tag{2}$$

In these and subsequent equations subscripts l, g, and s refer to the liquid, gaseous, and solid phases, respectively. Also, ϕ denotes porosity, $Y_{\alpha\beta}$ is the mass fraction of component α in phase β, ρ is phase density, e

1. Engineering Sciences Center, Sandia National Laboratories, Albuquerque, NM, USA. Phone:505-844-8729. E-Mail: mjmarti@sandia.gov

2. Phone: 505-844-1743. E-Mail: plhopki@sandia.gov

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

is internal energy, and S_β is phase saturation, and the pore space is assumed fully occupied by the liquid and gas phases, $S_l + S_g = 1$. The net component mass fluxes are defined as

$$F_\alpha = Y_{\alpha l}\rho_l v_l + Y_{\alpha g}\rho_g v_g - \rho_g D_{\alpha g}\nabla Y_{\alpha g}, \tag{3}$$

where v_β denotes the Darcy flux vector and $D_{\alpha g}$ is a pressure- and temperature-dependent effective binary diffusion coefficient. The advective fluxes are described by the extended Darcy law, in which relative permeabilities are introduced to account for the multiphase motion of fluids. Thus the Darcy flux vector of phase β (liquid or gas) is,

$$v_\beta = -\frac{k_{r\beta}}{\mu_\beta}\boldsymbol{k} \bullet (\nabla P_\beta + \rho_\beta g) \tag{4}$$

where P is pressure, g is the gravitational acceleration vector, and μ is dynamic viscosity. The intrinsic permeability tensor of the medium is $\boldsymbol{k}$ and the relative permeabilities are denoted $k_{r\beta}$. The intrinsic permeability tensor is assumed to be a property of the material under consideration, and as such is a spatially heterogeneous quantity. Note that we have assumed that each phase has its own phase pressure. The phase pressures are related via the capillary pressure relation, $P_g - P_l = P_c(S_l)$, which, as indicated, is assumed to be empirically specified as a function of the phase saturation of liquid. The net heat flux vector is defined by

$$q = -\lambda_T \nabla T + \sum_\beta \rho_\beta v_\beta h_\beta + \sum_\alpha h_{\alpha g} J_{\alpha g} \tag{5}$$

where λ_T is a saturation-dependent effective conductivity, h_β is phase enthalpy, $h_{\alpha g}$ is the enthalpy of component α in the gas phase, and $J_{\alpha g}$ is the gas-phase diffusion flux, the last term on the right-hand-side of Eq. (3).

The foregoing describes the major components of the mathematical model. Some additional information concerning thermodynamics and transport models can be found in [3].

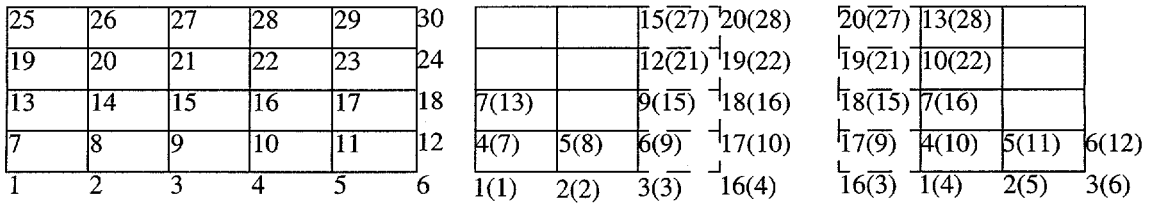
NUMERICAL METHOD

Spatial and Temporal Discretization.

The numerical method applied for solving the initial-boundary value problem formed by the coupled system of equations is a finite element method (FEM), enabling a general representation of complex geologic stratigraphy. The spatial discretization is accomplished by the Galerkin finite element method [4] utilizing fully integrated isoparametric multilinear elements via Gauss-Legendre quadrature. For problems with discontinuous solutions (e.g. phase boundaries), an upwinding scheme similar to Ref. [5] is under development. The resulting system of ordinary differential equations is integrated forward in time by a variable-step backward-difference predictor-corrector scheme first described in [6]. Two time integration methods are implemented. A first order scheme employs a forward Euler (FE) predictor with a backward Euler (BE) corrector. A second order scheme employs an Adams-Bashforth predictor with a trapezoid rule corrector. The predictors are used to obtain an initial estimate of variables at the next time step, t_{n+1}, thereby improving the initial estimate of the solution vector for use in a Newton iteration scheme applied to the nonlinear system of equations. The predictor/corrector scheme provides a method for estimating the local time truncation error, thereby providing a rational scheme for automatic time step control based on a user-specified truncation error tolerance.

Solution Procedures.

The task for the nonlinear solver is to find the solution vector $\boldsymbol{U}$ that minimizes the global residual vector, $\boldsymbol{R} = (\boldsymbol{R}_w, \boldsymbol{R}_a, \boldsymbol{R}_e)^T$, given by the discrete form of Eq. (1). The discretized system of nonlinear equations can be solved for the solution variables by Newton iteration [7]; each step of the iteration requires the solution of the following linear system for the update vector, $\delta\boldsymbol{U}$,



(a) (b) Subdomain for processor P0 (c)Subdomain for processor P1

Figure 1. Schematic of a domain decomposition in which the global mesh (a) is decomposed into two subdomains, (b) and (c). The dashed elements denote the "ghost elements" on each partition. The global mesh displays the global node numbering scheme, whereas the subdomains display the processor-level node numbering, with the corresponding global node numbers displayed in parenthesis.

$$\boldsymbol{J}(\boldsymbol{U}^q)\delta\boldsymbol{U}^{q+1} = -\boldsymbol{R}(\boldsymbol{U}^q), \tag{6}$$

where $\boldsymbol{J}$ is the Jacobian matrix. The solution vector is updated at each iteration (q) via $\boldsymbol{U}^{q+1} = \boldsymbol{U}^q + \delta\boldsymbol{U}^{q+1}$, until convergence is achieved. The Jacobian can be computed efficiently via forward difference approximations, by exploiting the fact that most terms are sums of products of basis functions and grid variables. This "inexact" Newton scheme (the term inexact refers to a numerical approximation of the Jacobian) is a convenient method of determining the Jacobian because any new transport parameter function or equation of state can be implemented without the need for the user to also program the gradient of the functions with respect to the solution vector. This is particularly helpful in the present class of problems where secondary variable calculations depend on which phases are present at a particular node point. However, special care must be taken in computing the forward differences to minimize finite-precision errors [7].

The Newton iteration scheme generates a linear system of equations to be solved for each update vector. The systems are solved using a parallel processing preconditioned Krylov solver library called Aztec [8]. The library includes several parallel iterative solution methods, including the conjugate gradient method for symmetric positive definite systems and a number of related methods for nonsymmetric systems, e.g. the generalized minimum residual method (GMRES). The library includes several preconditioners (e.g., Jacobi, least-squares polynomial, incomplete LU decomposition), which can be "mixed and matched" with the Krylov methods. See reference [7] for additional information.

Parallel Implementation.

The foregoing numerical algorithm is implemented for distributed memory parallel computers, or networked systems, via domain decomposition and message-passing techniques. We used the MPSalsa code [9] as a platform, including the parallel-processing implementation. However, there are several properties of the current problem which necessitated some special developments. A special data structure was necessary for the porous medium calculations. In our formulation it is necessary to have a unique material type specified on a node-point basis. This requirement can be attributed to the capillary pressure vs. saturation constitutive model, which is non-unique at a material boundary, and our choice of primary variables. Our solution to accommodate this non-FEM data structure was to build a processor node-point-to-material mapping. To ensure a consistent mapping across processors, inter-processor communication is required to update the correct mapping for the "ghost nodes" on each processor. The domain decomposition itself is performed with a modified version of the Chaco [10] graph partitioning code. The task for Chaco is to decompose the global node set graph into a user-specified number of partitions (subgraphs) in such a way as to minimize the edge interactions between partitions.

A simplified schematic of a domain decomposition is shown in Figure 1 for a structured grid (the algorithm is implemented for general unstructured grids). The global domain is decomposed into two subdomains, wherein the set of nodes assigned to each processor are numbered sequentially, starting with the set of nodes that "belongs" to the processor, and with the ghost or external nodes numbered last. In the figure, the global node numbers corresponding to the processor-level node numbers on the mesh partitions (Figures

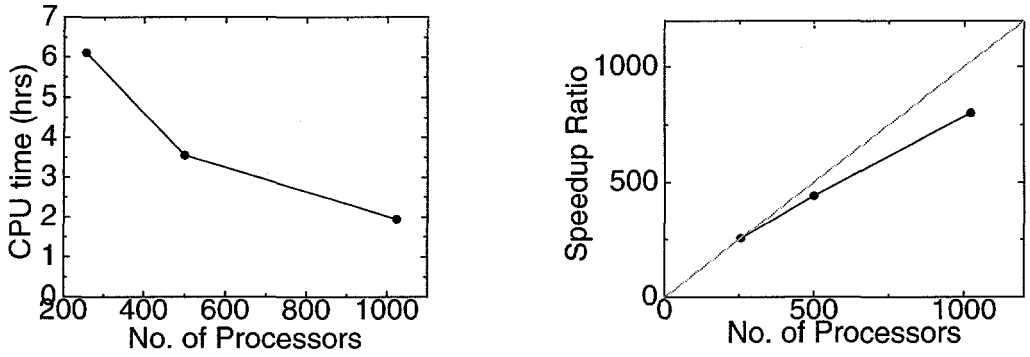


Figure 2. Parallel processing performance data for the 3D hydrothermal simulation.

1b and 1c) are shown in parenthesis. The dashed elements are the so-called "ghost elements" and the associated node points are referred to as "ghost nodes" or external nodes. Each processor is assigned the nodes corresponding to the un-dashed elements in Figure 1. However, on each processor the data for the external nodes is necessary to complete the processing for the border nodes, i.e. the surface nodes on each subdomain. Interprocessor communication is required for exchanging information associated with the ghost nodes, and those "border" nodes associated with the current processor. The interprocessor communication is set up to take advantage of the native Intel Paragon communication structures, enabling efficient communication on the Intel Paragon, and is also set up to use the standard message-passing library definition MPI [11], thus making the algorithm highly portable.

APPLICATIONS

Hydrothermal Simulation.

In this example we examine multiphase hydrothermal transport, due to a heat-generating source distribution, in a large scale 3D region modeled after Yucca Mountain (YM), Nevada, a site being considered for placement of a high-level nuclear waste repository. The computational grid is composed of over 358,000 node points and includes the major hydrostratigraphic units at YM. The entire region modeled measures about 1.6 km east to west, includes about 0.7 km above the water table (the lower boundary) and extends 3 km along its length. The materials are modeled as composite fractured media using the so-called equivalent continuum model (ECM) model [12] which assumes the fracture and matrix systems are locally in pressure equilibrium. The property data for this problem are those specified in [3], which contains additional information about this problem. These materials, and especially with the ECM, display many orders of magnitude variation in permeability from unit to unit, rendering a highly nonlinear problem which requires the Newton iteration scheme for convergence.

Lateral boundaries are specified as no-flow for all mass and energy balance equations. The entire lower boundary is modeled as a water table (moisture saturation unity) at 1 atmosphere pressure and a temperature of 20° C. The upper boundary is also at 1 atmosphere pressure, but at a temperature of 15° C. An infiltration flux of water at 0.1 mm/yr is applied uniformly over the entire upper boundary. These conditions simulate an upper boundary in contact with the atmosphere.

This problem was run on various numbers of processors to assess the parallel performance on the 1800 processor SNL Intel Paragon. This simulation, on a mesh consisting of 358,000+ grid points, requires the solution of a linear system composed of about 1.1 million equations which are solved at each time step. The problem was computed using the GMRES solver with incomplete LU preconditioning, and the forward/backward Euler predictor/corrector time integrator. The total CPU time, excluding loading mesh data and file handling, is shown in Figure 2 as a function of the number of processors. Each simulation, performed with different numbers of processors, required 71 time steps and 125 Newton iterations to integrate the solution out to 3000 yrs. This demonstrates the correct parallel implementation of the solution algorithm. The speedup ratio shown is relative to the CPU time on 256 processors, defined as $256 \bullet T_{256}/T_p$, where T_p is the time on p processors. This formula estimates the CPU time to run the problem on one processor using the CPU

time on 256, the smallest number of processors able to run this size problem on the Intel Paragon. The current problem is far too large to fit on a single processor of the Intel Paragon. On a fixed size grid, such as in the present case, the efficiency will always decrease as the problem is partitioned onto more processors, which increases the relative amount of communication compared to floating point operations.

Infiltration in heterogeneous media on a networked system.

To demonstrate the code's capability to execute in parallel on multiple workstations using the Message Passing Interface (MPI), a 3D problem involving patch infiltration into an initially "dry" layered region, measuring 8x2x6.5 m^3 (see [3] for additional details), was performed on one, two and four Sun SPARCstation20 workstations communicating over ethernet. The problem was computed on a fixed-size grid composed of 21758 node points using the GMRES solver with incomplete LU preconditioning and the forward/backward Euler integrator with automatic time step control, requiring 80 time steps and 164 Newton iterations to integrate to a time of 30 days. The CPU requirements and speedup ratios are shown in Figure 3. Note the high parallel efficiencies achieved, due to the high compute load per processor compared to communication overhead. The 94% efficiency on 4 processors is very good for performing communications over ethernet, demonstrating that existing distributed computing assets can still be used effectively to perform parallel calculations with this simulator.

A recent improvement in the simulator is the development of a general interface for specifying heterogeneous data. It allows the user to specify heterogeneous data on a nodal point basis not only for typical parameters (permeability, porosity), but also for user-specified parameters which may enter into transport models which the user may also define. This capability enables analysis of the impact of spatial distributions of material properties and model parameters, and uncertainty studies. The aforementioned 3D problem involving patch infiltration into a layered region was modified to demonstrate this capability. The modeled domain is shown in Figure 4 and originally involved four homogeneous regions; two thin layers across the full domain extent, and a high-permeability, rectangular inclusion in the remaining region. Different spatial distributions were used to assign permeability values at each computational node in the three "bulk" regions; the inclusion was left unchanged at its constant value. A plot of the resulting permeability field is shown in Figure 4, along with a comparison of saturation values along the same vertical profile $(x,y)=(1,0)$ m for the original problem (solid line) and the heterogeneous problem (dotted line). The heterogeneity results in dispersion of the saturation plume.

SUMMARY AND CONCLUSIONS

The equations describing nonisothermal two-phase flow in heterogeneous porous media have been implemented in a computer code for use on massively parallel computers, or over a network of computers, using unstructured grids. Good performance in terms of speedup and efficiency have been demonstrated through applications of practical interest. The capability described herein may be applied to many problems once considered intractable due to size or property complexity.

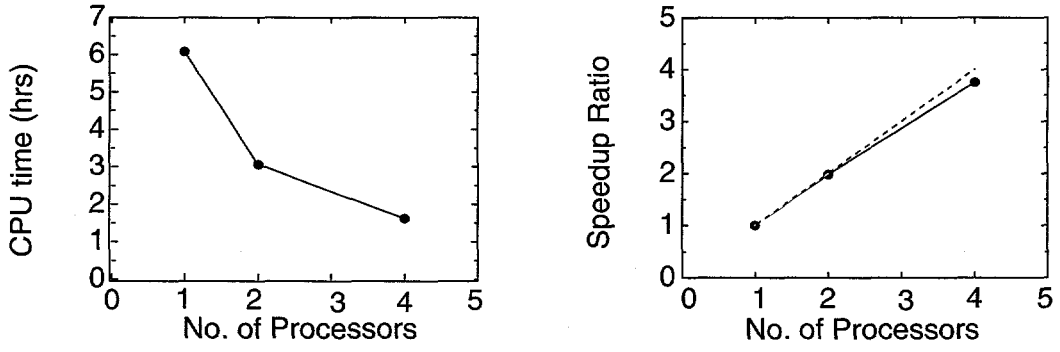


Figure 3. Parallel processing performance data on a network of workstations communicating over ethernet via MPI.

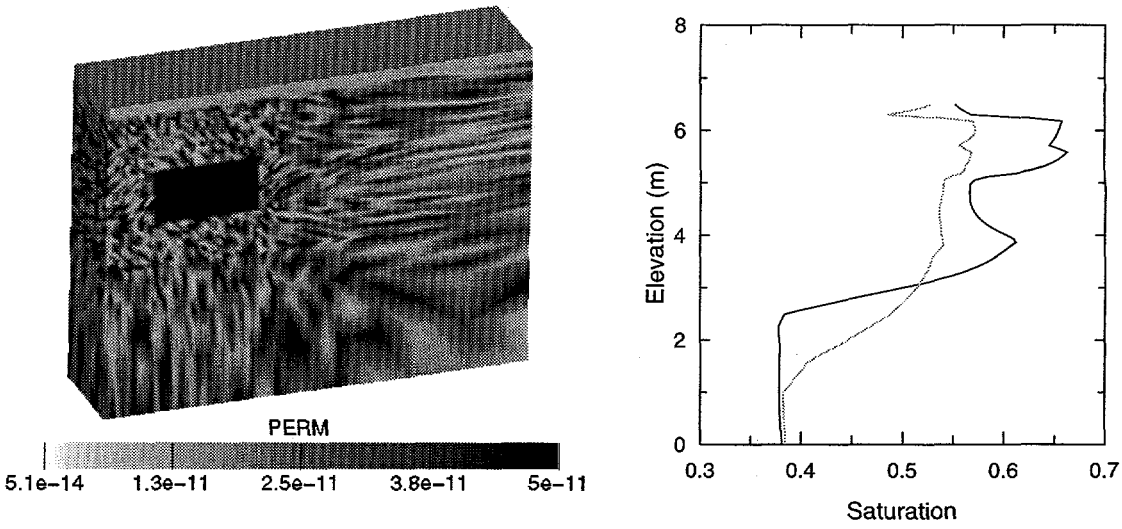


Figure 4. Contour plot of the generated permeability field and comparison of saturation profiles resulting from the original (solid line) and heterogeneous (dotted line) problem definition.

ACKNOWLEDGEMENTS

This work was performed under the LDRD program at Sandia National Laboratories. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-ACO4-94AL85000.

REFERENCES

1. Peaceman, D. W., 1977, *Fundamentals of Numerical Reservoir Simulation*, Elsevier, New York.
2. Martinez, M. J., 1995, Mathematical and numerical formulation of nonisothermal multicomponent three-phase flow in porous media, *SAND95-1247*, Sandia National Laboratories, Albuquerque, NM, 27 pp.
3. Martinez, M. J., P. L. Hopkins, and J. N. Shadid, 1997, LDRD Final Report: Physical simulation of nonisothermal multiphase multicomponent flow in porous media, *SAND97-1766*, Sandia National Laboratories, Albuquerque, NM, 65 pp.
4. Hughes, T. J. R., 1987, *The Finite Element Method*, Prentice-Hall, inc., Englewood Cliffs, NJ.
5. Forsyth, P., 1991, A control volume finite element approach to NAPL groundwater contamination, *SIAM J. Sci. Stat. Comput.*, **12**(5), 1029-1057.
6. Gresho, P. M., R. L., Lee, and R. L. Sani, 1980, On the time-dependent solution of the incompressible Navier-Stokes equations in two and three-dimensions, *Recent Advances in Numerical Methods in Fluids, Volume 1*, Pineridge Press Ltd., Swansea, U. K., 27-81.
7. Gill, P. E., W. Murray, and M. H. Wright, 1981, *Practical Optimization*, Academic Press, New York.
8. Hutchinson, S. A., J. N. Shadid and R. S. Tuminaro, 1995, Aztec user's guide, Version 1.0, *SAND95-1559*, Sandia National Laboratories, Albuquerque, NM, 40 pp.
9. Shadid, J. N., H. K. Moffat, S. A. Hutchinson, G. L. Hennigan, K. D. Devine, and A. G. Salinger, 1996, MPSALSA, A finite element computer program for reacting flow problems, Part 1 - Theoretical development, *SAND95-2752*, Sandia National Laboratories, Albuquerque,NM, 81 pp.
10. Hendrickson, B., and R. Leland, 1993, The Chaco user's guide, *SAND93-2339*, Sandia National Laboratories, Albuquerque, NM, 22 pp.
11. Gropp, W., E. Lusk, and A. Skjellum, 1995, *Using MPI*, MIT Press, Cambridge.
12. Peters, R. R., and E. A. Klavetter, 1988, A continuum model for water movement in an unsaturated fractured rock mass, *Water Resources Research*, **24**, No. 3, pp. 416-430.