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.

CONF-880672--1

DE88 006175

A SUBREGION BLOCK ITERATION TO 3-D FINITE ELEMENT MODELING OF
SUBSURFACE FLOW

G. T. Yeh

Environmental Sciences Division, Oak Ridge National Laboratory,
Oak Ridge, Tennessee 37831

ABSTRACT

A subregion block iteration (SBI) technique has been developed in conjunction with finite element approximations of saturated-unsaturated flow equations. The proposed SBI technique is implemented in a three-dimensional finite element saturated-unsaturated flow model. The model is verified with a nonlinear diffusion equation having an analytical solution. It is then applied to a burial trench problem. It is not possible to solve this field problem using the direct elimination finite element method. The SBI technique provides significant improvement over models based on direct band solution methods in both central processing unit (CPU) storage and CPU time.

INTRODUCTION

While iterative schemes for solving large field problems are routinely used in finite difference simulations, direct band solution methods are generally used in finite element approximations because of simplicity in assembling the coefficient matrix in banded form and easy access to band-matrix solver software. To make the application of finite element models to field problems feasible, a number of attempts have been made to incorporate various iterative schemes into finite element formulations (Yeh¹, Huyakorn et al.², Yeh³). We present here a "subregion block iteration" (SBI) technique for finite element simulations of saturated-unsaturated flow problems using 3DFEMWATER (Yeh³).

The SBI technique consists of three steps. First, the grid system is pre-processed to yield two pointer arrays; one is the global stencil array and the other is the local (sub-region) stencil array. The pre-processing only has to be performed once unless the spatial discretization is changed during a simulation. Second, the global matrix is assembled and

MASTER
Jsu

compressed, using the global stencil array, to contain non-zero elements, thereby minimizing the central processing unit (CPU) storage requirements. Third, for each subregion, the sub-matrix equation is formed from the stored global matrix and global vector by the local stencil array and the node-mapping function; the sub-matrix equation is solved with direct elimination, thus resulting in large savings in CPU time because the band-width of the sub-matrix is much smaller than that of the global matrix. The third step is performed for all subregions for each iteration until a convergent solution is obtained. When every nodal point, line, or vertical plane in the region is treated as a subregion, the SBI becomes the point, line, or sliced successive over relaxation (SSOR) iteration method, respectively.

PRE-PROCESSING OF GEOMETRIC DATA

The pre-processing step analyzes the geometric information of a chosen finite element discretization and prepares two pointer arrays. One of the pointer arrays is used for compressing the global matrix to contain non-zero elements and the other is used to recover the actual global position of the rearranged block position. Let us use a simple example (Fig. 1) to illustrate how two pointer arrays are generated. Having a finite element discretization, we first subdivide the region into an arbitrary number of subregions (Fig. 1a). Second, we renumber each subregion so that the band width for the sub-matrix will be minimum. It is noted that intra-block nodes are numbered after all interior nodes have been numbered (Fig. 1b). Third, we input a node-mapping function $N(K,I)$ between global nodes and local nodes for all subregions. Only block interior nodes (those to the left side of the dotted line) have to be input; intra-block nodes (those to the right side of the dotted line) are instead generated based on the input node-mapping function and element connectivity (Fig. 1c). Fourth, $NGS(N,I)$, a global stencil array, is generated based on the element connectivity (Fig. 1d). Fifth, the node-mapping function $N(K,I)$ is filled up for the intra-block nodes based on element connectivity and the partial input of the node-mapping function (Fig. 1c). Finally, $NLS(L,I,K)$, a local stencil array, is generated based on the global stencil $NGS(N,I)$ and the complete node-mapping function $N(K,I)$ (Fig. 1e).

ASSEMBLING OF COMPRESSED GLOBAL MATRIX AND GLOBAL LOAD VECTOR

Since the Galerkin finite element method results in a 27-point formula for three-dimensional problems, the maximum number of non-zero coefficients for any equation is around 27. The question is how to assemble from element matrices the global matrix in the compressed form. This can be easily accomplished by using the pre-processed global stencil array $NGS(N,I)$. For example, if $NGS(N,I) = K$, then the coefficient corresponding to the K -th unknown in the N -th equation is stored in $C(N,I)$ of the compressed global matrix C . In the

meantime, the right-hand side of the N-th equation is stored in the global vector R(N).

Fig. 1 Pre-processing of global stencil and local stencil arrays, $NGS(N, I)$ and $NLS(L, I, K)$.

SUCCESSIVE BLOCK ITERATION

The major tasks in the block iteration step are: (1) constructing of sub-matrix equations and (2) solving the sub-matrix equations with direct band solvers. The construction of sub-matrix equations out of the compressed global matrix and global load vector can be achieved with the local stencil array $NLS(L,I,K)$ and the node-mapping function $N(K,I)$. For the LI -th equation in the K -th subregion, the LI -th row of the banded sub-matrix $CL(NEQ,IBAND)$ (where NEQ is the number of unknowns and $IBAND$ is the band-width) can be obtained as follows. First, loop over J from 1 to $JBAND$ (where $JBAND$ is the maximum number of non-zero entries in any row of the compressed global matrix C) and second, set $NI = N(K,LI)$ and $LJ = NLS(LI,J,K)$. Third, set $NJ = N(K,LJ)$ and $LJB = LJ - LI + IHBP$ (where $IHBP$ is the half band-width plus 1). Fourth, put $C(NI,J)$ into $CL(LI,LJB)$ if LJ is less than or equal to the total number of interior nodes of the K -th subregion. If LJ is greater than the total number of the interior nodes of the K -th subregion (recall that we number the intra-boundary nodes only after we have numbered all interior nodes of the subregion), then intra-boundary conditions are implemented by setting $RL(LI) = RL(LI) - C(NI,J)*HW(NJ)$ (where $RL(LI)$ is the LI -th component of the local load vector RL and $HW(NJ)$ is the variable at the NJ -th global node from previous iteration). Finally, global boundary conditions are incorporated if the K -th subregion includes any global boundary node. The solution of the sub-matrix equation, $CL(LI,LJB)*H(LI) = RL(LI)$, with a band matrix solver is straightforward (Yeh³). The construction of sub-matrix equations and their solution are performed for all subregions in each iteration until a convergent solution is obtained.

VERIFICATION AND APPLICATION

Verification with a Horizontal Column Diffusion Problem.

This example was selected to represent the simulation of a nonlinear one-dimensional moisture-diffusion problem with the SBI 3DFEMWATER. The total length of the column was 100 cm. The medium in the column was assumed to have a saturated diffusion coefficient of $K_s = 10.0 \text{ cm}^2/\text{day}$, a saturated moisture content of $\theta_s = 0.4$, a field capacity of $\theta_r = 0.1$, and diffusion coefficient function given elsewhere (Philip⁴). The initial conditions were assumed to be $\theta = 0.1$ throughout the entire column. The boundary conditions were: moisture content is held at 0.4 on the left side of the column and is given as a function of time as $\theta = \theta_r + (\theta_s - \theta_r)/(1 + 100/\sqrt{K_s t})^2$ on the right side of the column.

The column was discretized with $1 \times 1 \times 100 = 100$ elements with element size $= 1 \times 1 \times 1 \text{ cm}$, resulting in $2 \times 2 \times 101 = 404$ node points. For model simulation, each of the four horizontal lines was considered a subregion. Thus, a total of four subregions, each with 101 node points, were used for the subregion block iteration simulation. A variable time step

size was used. The initial time step size was 0.01 day and each subsequent time step size was increased by a factor of 0.25 with a maximum time step size of 2.0 days. A 40 day simulation was made with total of 40 time steps. The moisture content tolerance for nonlinear iteration was set to $1.0 \cdot 10^{-5}$, the nonlinear relaxation factor was set equal to 0.5, the tolerance for block iteration was set to $0.5 \cdot 10^{-5}$, and relaxation parameter of block iteration was set 1.0. The simulated moisture content is plotted in Fig. 2. It shows very close agreement between the analytical (Philip⁴) and the numerical solutions. Computer output indicates that the agreement is up to the fourth digit for long-time simulation.

Fig. 2 Comparison of moisture content as simulated by exact and SBI numerical solutions.

Application to a Burial Trench Problem

In order to assess the perching and/or mounding of groundwater in burial trenches, the SBI 3DFEMWATER (Yeh³) was applied to a typical single trench (Solomon and Yeh⁵). A finite element mesh consisting of $25 \times 17 \times 11 = 4,675$ nodes was used for simulating a solution region of 44 m long by 25 m wide and 15 m deep. For SBI simulations, the region is divided into 17 subregions, each subregion has $25 \times 11 = 285$ nodes. For the steady-state simulation, about 15 nonlinear iterations are required to reach a convergent solution because of very high nonlinearity of the problem. Within each nonlinear iteration, an average of 27 block iterations are required to achieve a convergent solution with maximum error tolerance of 0.5 cm. For most of the case studies, approximately 20 minutes of CPU time were required to reach a steady state solution on an IBM 370/3033. Had a direct band matrix solver been used, the band-width of the global coefficient matrix would be 399.

This would have required a CPU storage of approximately 15 mega-bytes to store the global coefficient matrix. Since an IBM 370/3033 has only about 10 mega-bytes CPU storage, we would not be able to use a direct solution method to tackle this problem. On the other hand, the SBI technique requires about 1.0 mega-byte for the compressed global matrix, 1.1 mega-bytes for global stencil array, local stencil array, and node-mapping array, and 0.06 mega-bytes for the sub-matrix. Thus, a saving of about 85% of CPU storage with the SBI technique enable the simulation of this problem. Operational count indicates that the CPU time is proportional to the square of matrix bandwidth. Thus, the CPU time with SBI would have been only approximately equal to $(25/399)^2 \times 27 = 0.11$ times that with direct solution method. Thus, only a fraction of CPU time would be needed with SBI technique in comparison to direct solution method for large three-dimensional problems.

ACKNOWLEDGEMENTS

This research is supported by the Office of Defense Waste and Transportation Management, U. S. Department of Energy under contract No. DE-AC05-84 OR21400, with Martin Marietta Energy Systems, Inc. ESD Publication No. 3064.

REFERENCES

1. Yeh, G. T. (1985), Comparison of successive iteration and direct methods to solve finite element equations of aquifer contaminant transport, Water Resour. Res., Vol. 21, pp. 272-280.
2. Huyakorn, P. S., Springer, E. P., Guvanasen, V., and Wadsworth, D. (1986), A three-dimensional finite element model for simulating water flow in variably saturated porous media, Water Resour. Res., Vol. 22, pp. 1790-1808.
3. Yeh, G. T. (1987), 3DFEMWATER: A Three-Dimensional Finite Element Model of WATER Flow through Saturated-Unsaturated Media, ORNL-6386, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831.
4. Philip, J. R. (1960), General method of exact solution of the concentration-dependent diffusion equation, Aust. J. Phys., Vol. 13, pp. 1-12.
5. Solomon, D. K. and Yeh, G. T. (1987), Application of 3DFEMWATER to the Study of Trench "Bathtubbing", (Personal Communication).