Semianalytical Computation of Path Lines for Finite-Difference Models

by David W. Pollock[a]

ABSTRACT

A semianalytical particle tracking method was developed for use with velocities generated from block-centered finite-difference ground-water flow models. The method is based on the assumption that each directional velocity component varies linearly within a grid cell in its own coordinate directions. This assumption allows an analytical expression to be obtained describing the flow path within an individual grid cell. Given the initial position of a particle anywhere in a cell, the coordinates of any other point along its path line within the cell, and the time of travel between them, can be computed directly. For steady-state systems, the exit point for a particle entering a cell at any arbitrary location can be computed in a single step. By following the particle as it moves from cell to cell, this method can be used to trace the path of a particle through any multidimensional flow field generated from a block-centered finite-difference flow model.

INTRODUCTION

The use of particle tracking techniques to generate path lines and time-of-travel information from the results of numerical models can be extremely helpful in analyzing complex, two- and three-dimensional ground-water flow systems. Particle tracking schemes have been formally incorporated into solute transport models to account for the advective component of transport (Konikow and Bredehoeft, 1978; Prickett and others, 1981). Particle tracking also has been used less formally as a simple means of evaluating the advective transport characteristics of ground-water systems by generating path lines (Mandle and Kontis, 1986; Shafer, 1987) or by tracking a solute "front" with an array of particles (Garabedian and Konikow, 1983).

Particle tracking is extremely simple in concept. Generally, particles are tracked through a flow field explicitly by computing the directional components of velocity at a particle's current position and moving the particle to a new location that is determined by multiplying those velocity components by a finite time step to obtain the incremental changes in the particle's x, y, and z coordinates over that interval of time. As this process is repeated, a series of x-y-z and time coordinates are produced that trace the path of a particle through the flow field as a function of time.

For analytical solutions, the velocity vector field is known directly from the solution at every point in the flow field. For finite-difference models, it is necessary to establish an interpolation scheme so that the ground-water velocity vector can be computed at every point in the flow field, based on the intercell flow rates from the finite-difference model. A variety of interpolation schemes are possible. Perhaps the three simplest approaches are (1) step function interpolation, (2) simple linear interpolation, and (3) multilinear interpolation. The step function interpolation scheme assumes that velocity components are constant between any two nodes, and change value abruptly at the nodes. Simple linear interpolation assumes that the principal velocity components vary linearly

[a]Hydrologist, Water Resources Division, U.S. Geological Survey, 411 National Center, Reston, Virginia 22092.

Received November 1987, revised March 1988, accepted March 1988.

Discussion open until May 1, 1989.

within a grid cell only with respect to their own coordinate direction. That is, the x-velocity component within a cell varies linearly in the x-direction, but is independent of y and z within the cell. Multilinear interpolation is similar to simple linear interpolation, except that each of the three principal components of velocity is assumed to be a linear function of all three coordinate directions. Although these three interpolation schemes all differ slightly from one another, each is capable of producing a reasonable representation of the velocity vector field for well-discretized flow fields.

The particle tracking algorithm described in this paper uses simple linear interpolation to generate the velocity vector field. One advantage of simple linear interpolation is that the three principal velocity component functions can be integrated directly within each individual grid cell to obtain an analytical expression for path line segments within each cell. The ability to generate an analytical expression for a particle path line within a cell eliminates the need for a series of explicit computations at small, discrete time steps, and avoids the additional numerical error and time step size constraints that are characteristic of explicit computation schemes. In the case of steady-state flow, the direct integration approach can be implemented in a very simple algorithm that allows a particle's exit point from any given grid cell to be computed based on any known starting location within the cell. This paper presents a development of this direct integration method and discusses some general aspects of its implementation. Sample problems are presented that demonstrate its use and applicability to ground-water flow problems with complex flow patterns characterized by highly divergent and convergent flow.

THEORY

The partial differential equation describing conservation of mass in a steady-state, three-dimensional ground-water flow system can be expressed as,

$$\frac{\partial(n\,v_x)}{\partial x} + \frac{\partial(n\,v_y)}{\partial y} + \frac{\partial(n\,v_z)}{\partial z} = W \qquad (1)$$

where v_x, v_y, and v_z are the principal components of the average linear ground-water velocity vector; n is porosity; and W is the volume rate of water created or consumed by internal sources and sinks per unit volume of aquifer. Equation (1) expresses conservation of mass for an infinitesimally small volume of aquifer. The finite-difference approximation of equation (1) can be thought of as a mass balance equation for a finite-sized cell of aquifer that accounts for water flowing into and out of the cell and water generated internally within the cell. Figure 1 shows a finite-sized cell of aquifer and the components of inflow and outflow across its six faces. In the discussion that follows, the six cell faces are referred to as x_1, x_2, y_1, y_2, z_1, and z_2. Face x_1 is the face perpendicular to the x direction at $x = x_1$. Similar definitions hold for the other five faces. The average velocity component across each face in cell (i, j, k) is obtained by dividing the volume flow rate across the face by the cross-sectional area of the face and the porosity of the material in the cell,

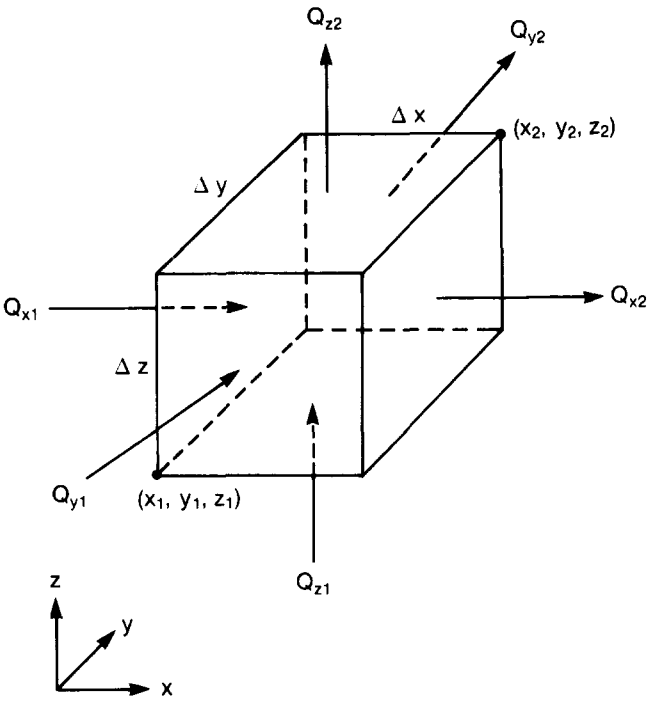

Fig. 1. Orientation of finite-difference cells and definition of cell face flows.

$$v_{x1} = Q_{x1}/(n\,\Delta y\,\Delta z) \qquad (2a)$$

$$v_{x2} = Q_{x2}/(n\,\Delta y\,\Delta z) \qquad (2b)$$

$$v_{y1} = Q_{y1}/(n\,\Delta x\,\Delta z) \qquad (2c)$$

$$v_{y2} = Q_{y2}/(n\,\Delta x\,\Delta z) \qquad (2d)$$

$$v_{z1} = Q_{z1}/(n\,\Delta x\,\Delta y) \qquad (2e)$$

$$v_{z2} = Q_{z2}/(n\,\Delta x\,\Delta y) \qquad (2f)$$

where Q is a volume flow rate across a cell face; and Δx, Δy, and Δz are the dimensions of the cell in the respective coordinate directions. If we allow for possible internal sources or sinks within the cell

(Q_s), the following mass balance equation can be written for the cell,

$$\frac{(nv_{x2}-nv_{x1})}{\Delta x}+\frac{(nv_{y2}-nv_{y1})}{\Delta y}+\frac{(nv_{z2}-nv_{z1})}{\Delta z}=\frac{Q_s}{\Delta x\Delta y\Delta z} \quad \ldots.(3)$$

The left side of equation (3) represents the net volume rate of outflow per unit volume of the cell, and the right side represents the net volume rate of production per unit volume due to internal sources and sinks. Each of the flow rates at the cell's six faces can be expressed in terms of Darcy's law using the head at the cell in question and the heads at the nodes of the six adjacent cells together with estimates of the effective hydraulic conductivity between adjacent cells. Substitution of Darcy's law for each of the flow terms in equation (3) results in a set of algebraic equations expressed in terms of heads at nodes located at the cell centers (McDonald and Harbaugh, 1984, p. 20). The solution of that set of algebraic equations yields the values of head at the node points. Once the head solution has been obtained, the intercell flow rates can be computed from Darcy's law using the values of head at the node points.

In order to compute path lines, a method must be established to compute values of the principal components of the velocity vector at every point in the flow field given the intercell flow rates from the finite-difference model. Both the differential equation of ground-water flow and its finite-difference approximation are commonly derived by developing a mass balance equation for a finite-sized volume (or cell) within an aquifer, as in the preceding paragraph. In such derivation it is customary to interpret the velocity components at the cell faces as representing estimates of the principal velocity components at the locations of the cell faces (Todd, 1980, pp. 99-100; Freeze and Cherry, 1979, pp. 546-548). That interpretation lends support to the use of some sort of continuous, or piece-wise continuous, interpolation function to compute velocity vector components at locations within a cell based on values at the cell faces. The algorithm described in this paper is based on the use of simple linear interpolation of the principal velocity components within a cell. Using simple linear interpolation, the principal velocity components can be expressed in the form,

$$v_x = A_x(x - x_1) + v_{x1} \quad (4a)$$

$$v_y = A_y(y - y_1) + v_{y1} \quad (4b)$$

$$v_z = A_z(z - z_1) + v_{z1} \quad (4c)$$

where A_x, A_y, and A_z are constants that correspond to the components of the velocity gradient within the cell,

$$A_x = (v_{x2} - v_{x1})/\Delta x \quad (5a)$$

$$A_y = (v_{y2} - v_{y1})/\Delta y \quad (5b)$$

$$A_z = (v_{z2} - v_{z1})/\Delta z \quad (5c)$$

Linear interpolation produces a continuous velocity vector field within each individual grid cell that identically satisfies the differential conservation of mass equation [equation (1)] everywhere within the cell. That point can be illustrated by noting that when linear velocity component functions [equations (4a)-(4c)] are substituted into equation (1), the three derivatives on the left side of equation (1) become constants that are identically equal to the three terms on the left side of equation (3) (provided that porosity is considered constant within a cell). Consequently, linear interpolation of the six cell face velocity components results in a velocity vector field within the cell that automatically satisfies equation (1) at every point inside the cell, provided that any internal sources or sinks are thought of as being uniformly distributed within the cell. The fact that the velocity vector field within each cell satisfies the differential mass balance equation is important because it assures that path lines will distribute water throughout the flow field in a way that is consistent with the overall movement of water throughout the system indicated by the solution of the finite-difference flow equations.

Now, let us consider the movement of a particle through a three-dimensional finite-difference cell. The rate of change in the particle's x-component of velocity as it moves through the cell is given by

$$(dv_x/dt)_p = (dv_x/dx)(dx/dt)_p \quad (6)$$

where the subscript, p, refers to the particle. The term $(dx/dt)_p$ is the time rate of change of the x-location of the particle. By definition,

$$v_{xp} = (dx/dt)_p \quad (7)$$

where v_{xp} is the x-component of velocity for the particle. Differentiating equation (4a) with respect to x yields the additional relation,

$$dv_x/dx = A_x \quad (8)$$

Substituting equations (7) and (8) into equation (6) gives,

$$(dv_x/dt)_p = A_x\, v_{xp} \quad (9a)$$

Analogous equations are obtained for the y and z directions,

$$(dv_y/dt)_p = A_y v_{yp} \quad (9b)$$

$$(dv_z/dt)_p = A_z v_{zp} \quad (9c)$$

Equations (9a) through (9c) can be rearranged to the form,

$$(1/v_{xp})\, dv_{xp} = A_x\, dt \quad (10)$$

Equation (10) can be integrated and evaluated between times t_1 and t_2 ($t_2 > t_1$) to give,

$$\ln[v_{xp}(t_2)/v_{xp}(t_1)] = A_x \Delta t \quad (11)$$

where $\Delta t = t_2 - t_1$. By taking the exponential of each side of equation (11), substituting equation (4a), and rearranging, we obtain,

$$x_p(t_2) = x_1 + (1/A_x)[v_{xp}(t_1)\exp(A_x \Delta t) - v_{x1}] \quad (12a)$$

Analogous equations can be developed for the y and z directions,

$$y_p(t_2) = y_1 + (1/A_y)[v_{yp}(t_1)\exp(A_y \Delta t) - v_{y1}] \quad (12b)$$

$$z_p(t_2) = z_1 + (1/A_z)[v_{zp}(t_1)\exp(A_z \Delta t) - v_{z1}] \quad (12c)$$

The velocity components $v_{xp}(t_1)$, $v_{yp}(t_1)$, and $v_{zp}(t_1)$ are known functions of the particle's coordinates at time t_1; consequently, the coordinates of the particle at any future time (t_2) can be computed directly from equations (12a) through (12c).

For steady-state flow, the direct integration method described above can be imbedded in a simple algorithm that allows a particle's exit point from a cell to be determined directly given any known starting location within the cell. To illustrate the method, consider the two-dimensional example shown in Figure 2. Cell (i, j) is in the x-y plane and contains a particle, P, located at (x_p, y_p) at time t_p. For this example, it is assumed that v_{x1} and v_{x2} are greater than zero. That is, water flows into the cell through face x_1 and out of the cell through face x_2. Similarly, it is assumed that v_{y1} and v_{y2} are also greater than zero, so that water flows into the cell through face y_1 and out of the cell through face y_2. Figure 2 shows particle P entering the cell at the boundary face x_1; however, the discussion that follows is valid for any particle location within the cell.

The first step is to determine the face across which particle P passes as it leaves cell (i, j). For the present example, this is accomplished by noting that the velocity components at the four faces require that particle P leave the cell through either face x_2 or face y_2. Consider the x-direction first.

From equation (4a), v_{xp} can be calculated at the point (x_p, y_p). Since we also know v_x equals v_{x2} at face x_2, equation (11) can be used to determine the time that would be required for particle P to reach face x_2,

$$\Delta t_x = (1/A_x) \ln (v_{x2}/v_{xp}) \quad (13a)$$

An analogous calculation can be made to determine the time required for particle P to reach face y_2,

$$\Delta t_y = (1/A_y) \ln (v_{y2}/v_{yp}) \quad (13b)$$

where v_{xp} and v_{yp} are the x and y components of velocity of particle P at (x_p, y_p). If Δt_x is less than Δt_y, particle P will leave the cell across face x_2 and enter cell (i, j + 1). Conversely, if Δt_y is less than Δt_x, particle P will leave the cell across face y_2 and enter cell (i − 1, j). A third possibility is that Δt_x and Δt_y are equal, in which case particle P would leave through the corner of cell (i, j) and enter cell (i − 1, j + 1). The particle trajectory shown in Figure 2 corresponds to a situation where Δt_y is less than Δt_x. The length of time required for particle P to travel from point (x_p, y_p) to a boundary face of cell (i, j) is taken to be the smaller of Δt_x and Δt_y, and is denoted as Δt_e. The value Δt_e is then used in equations (12a) through (12c) to determine the exit coordinates (x_e, y_e) for particle P as it leaves cell (i, j),

$$x_e = x_1 + (1/A_x)[v_{xp}(t_p)\exp(A_x \Delta t_e) - v_{x1}] \quad (14a)$$

and

$$y_e = y_1 + (1/A_y)[v_{yp}(t_p)\exp(A_y \Delta t_e) - v_{y1}] \quad (14b)$$

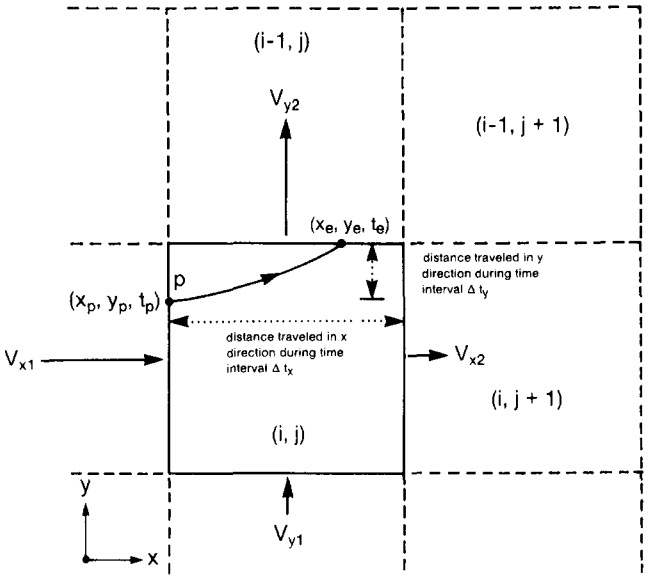

Fig. 2. Schematic representation of a particle path through a two-dimensional cell.

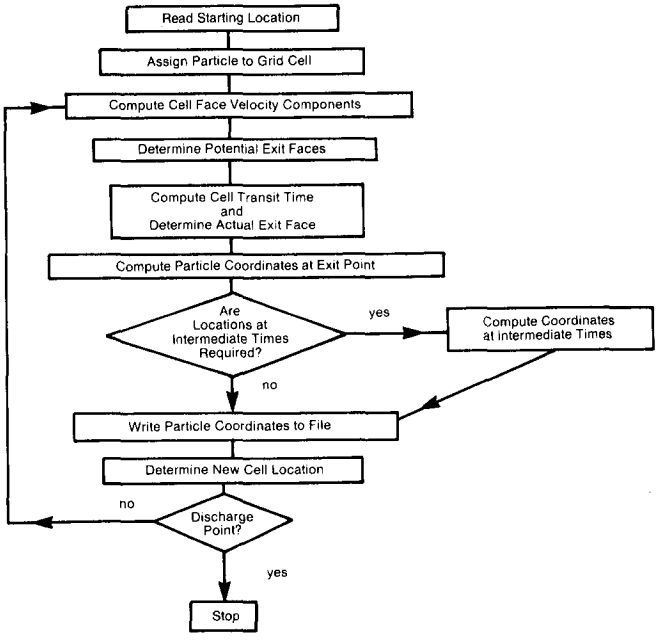

Fig. 3. Flow chart for particle tracking algorithm for steady-state systems.

The time at which particle P leaves the cell is given by: $t_e = t_p + \Delta t_e$. This sequence of calculations is repeated, cell by cell, until the particle reaches a discharge point. The approach can be generalized to three dimensions in a straightforward way by performing all of the calculations for the z-direction in addition to the x- and y-directions. A flow chart outlining the algorithm for a steady-state flow system is presented in Figure 3.

It is sometimes desirable, or even necessary, to calculate the position of a particle at a specific point in time. In the case of transient problems, the coordinates of the particle must be computed at the end of each finite-difference time step because the velocity components at the cell face change from one time step to the next. Even for steady-state systems, it is often desirable to calculate the location of a particle at specific points in time that will not generally correspond to those points in time at which the particle passes from one cell to another. For these cases, the coordinates of a particle at any intermediate time can be computed directly from equations (12a) through (12c) using an appropriate value for Δt within the range 0 to Δt_e.

For the purposes of illustration, the preceding example considered a specific case where all of the velocity components at the cell faces were nonzero and in the positive x or positive y directions. Of course, those conditions will not always exist. Figure 4 illustrates the other possible situations that can occur in any of the three coordinate directions. Figure 4(a) shows the case where v_{x1} and v_{x2} are in opposite directions and flow is into the cell through both faces x_1 and x_2. For this case, it is obvious that once a particle enters the cell, it cannot leave the cell in the x-direction. When implementing this algorithm, a check is made to determine if this condition exists for given coordinate direction. If so, a flag is set to indicate that the particle cannot leave the cell across either of the faces in that direction. When this situation prevails in all three coordinate directions, it indicates that a strong sink is present within the cell and no outflow can occur. Figure 4(b) shows a second alternative in which v_{x1} and v_{x2} are in opposite directions and flow is out of the cell through faces x_1 and x_2. This condition implies that a local flow divide exists for the x-direction somewhere within the cell. For this situation, the potential exit face in the x-direction is determined

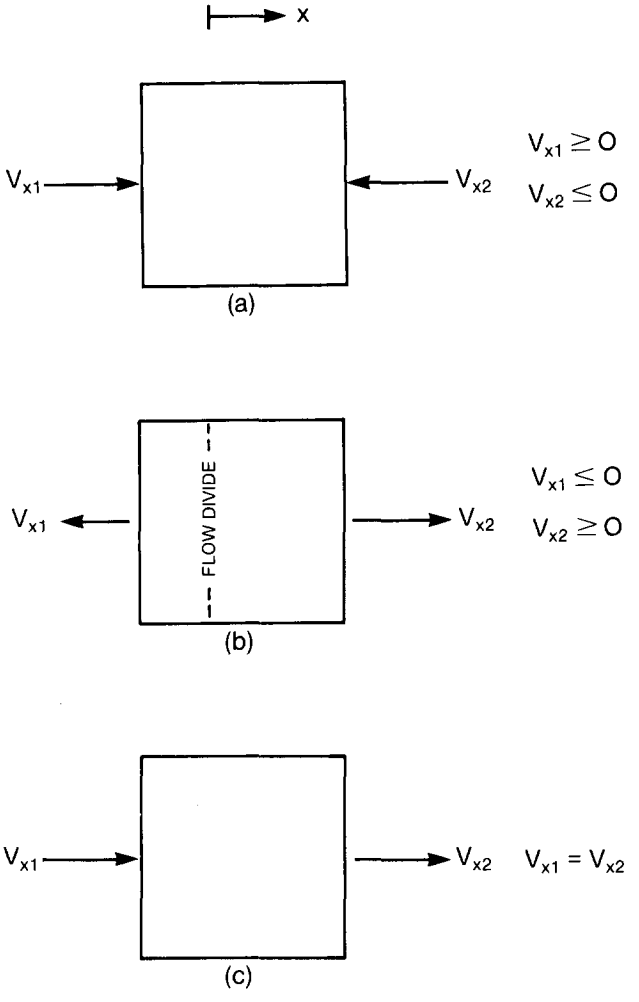

Fig. 4. Possible orientations of cell face velocity components that must be considered in determining potential exit faces.

by checking the sign of v_{xp}. If v_{xp} is less than zero, the particle has the potential to leave the cell across face x_1. On the other hand, if v_{xp} is greater than zero, the particle has the potential to leave the cell in the x-direction only through face x_2. Once the appropriate potential exit face has been determined for a coordinate direction, the transit time for that direction can be computed as outlined in the preceding discussion. Finally, the case where v_{x1} is non-zero and equal to v_{x2} [Figure 4(c)] also must be considered a special case because the quotient ln(1)/0 that results from the formulation of equation (11) is indeterminant and cannot be computed. When that is the case, equation (11) is bypassed, and the transit time in the x-direction is computed from the simple relations,

$$\Delta t_x = (x_2 - x_p)/v_{x1} \qquad (v_{x1} > 0) \qquad (15a)$$

or

$$\Delta t_x = (x_1 - x_p)/v_{x1} \qquad (v_{x1} < 0) \qquad (15b)$$

EXAMPLES AND DISCUSSION

Two examples are considered to illustrate the application of the technique. The first example is shown schematically in Figure 5. The steady-state two-dimensional ground-water flow equation is solved for a situation where water is injected at a constant rate of 160,000 cubic ft/day into a well located in a confined aquifer with a thickness of

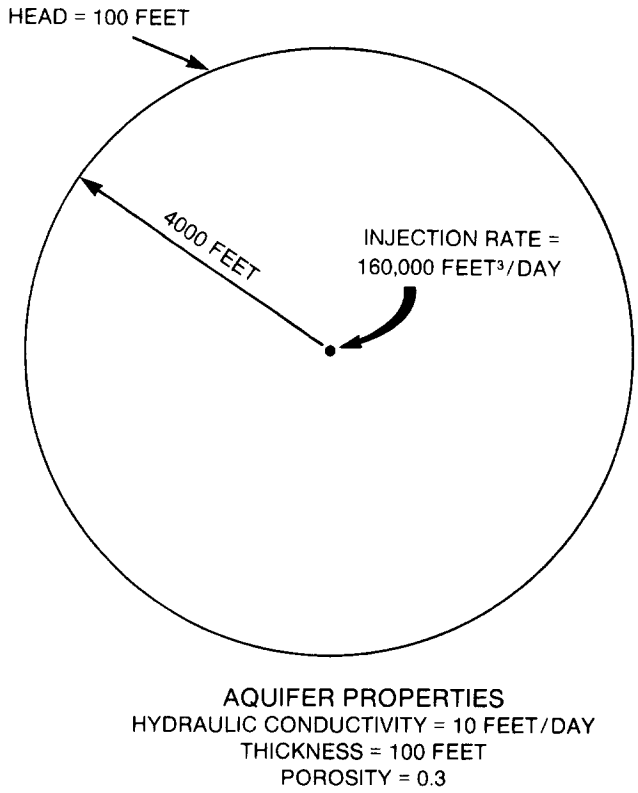

Fig. 5. Schematic description of radial flow example.

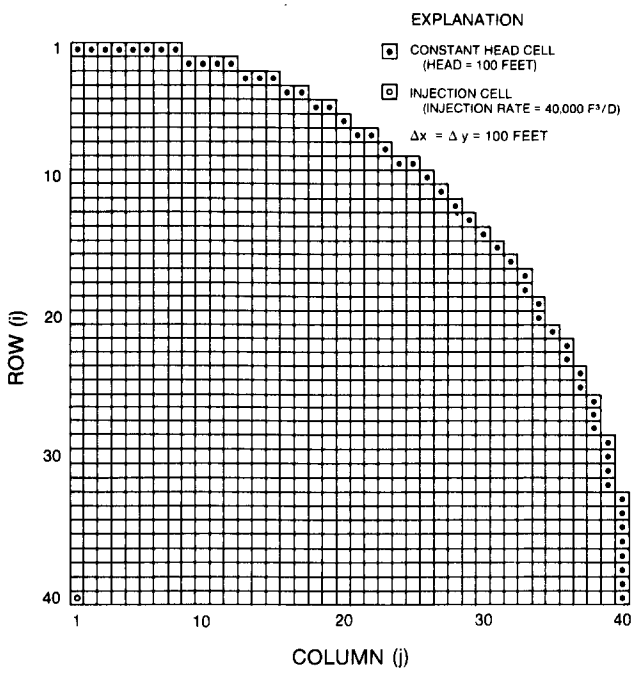

Fig. 6. Finite-difference grid and boundary conditions for radial flow problem.

100 ft, an hydraulic conductivity of 10 ft/day, and a porosity of 0.3. The hydraulic head at a radial distance of 4000 ft from the well is held constant at 100 ft. For this problem the velocity at any point in the flow field is one-dimensional in the radial direction, and an exact, analytical solution can be obtained describing the position of a particle as a function of time. The symmetry of the problem requires that only one-fourth of the circular flow field be considered. Figure 6 shows the finite-difference grid and boundary conditions used to approximate flow through one-fourth of the system shown in Figure 5. For this approximation, the center of the well is located in the lower left corner of Figure 6, and an injection rate of 40,000 cu ft/day is applied to cell (40, 1). The finite-difference solution was obtained using the USGS modular finite-difference flow model (McDonald and Harbaugh, 1984). Ten particles were placed at a radial distance of 150 ft from the center of the well. The points in Figure 7 show the positions of the particles after 2500 days, 5000 days, and 7500 days. The dashed curves represent the predicted position of the particles at these three points in time based on the analytical solution. The numerically generated particle path lines are shown as solid lines. The numerical results agree well with the analytically determined particle positions.

The second example is illustrated schematically in Figure 8. To the left of an impermeable retaining

wall is a 50-ft thickness of fine sand overlain by 5 ft of water in a lake, or reservoir. A seepage face at an elevation of 25 ft above the base of the aquifer is present to the right of the retaining wall. The difference in head between the reservoir and the seepage face causes water to flow downward from the reservoir, laterally under the wall, and discharge upward to the seepage face. Figure 9 shows the finite-difference grid and boundary conditions used to simulate constant heads of 55 ft at the base of the reservoir and 25 ft at the top of cells (7,12) through (7,21) in row 7. Figure 10 shows path lines for five particles originating along the base of the reservoir. The path lines are superimposed on a plot of equipotential lines contoured from the results of the numerical simulation. A good orthogonal relation exists between the path lines and the equipotential lines.

Particle 1 starts very near the far left boundary of the system. It moves down along the left boundary, turns right at the base of the aquifer, moves to the right along the base until it reaches the far right boundary, where it then turns upward and eventually discharges along the seepage face within cell (6,21). In Figure 10, the path line for particle 1 coincides with the boundary and is only visible as two diagonal line segments in the lower left and lower right corners of the aquifer. These line segments connect the entry and exit points of particle 1 in cells (11,1) and (11,21). In fact, the method actually implicitly accounts for a highly curved flow path in these two cells, but that curvature is not expressed in Figure 10 because the path lines were constructed simply by connecting the entry and exit points in each of the cells with straight lines. However, the influence of the highly curved path is evident in the transit time calculated for cells (11,1) and (11,21). This point can be illustrated by taking a closer look at cell (11,1). Figure 11 shows an expanded plot of cell (11,1). The entry and exit points are shown as solid circles, and their respective x-z coordinates are noted in parentheses. The velocity components at the four faces are,

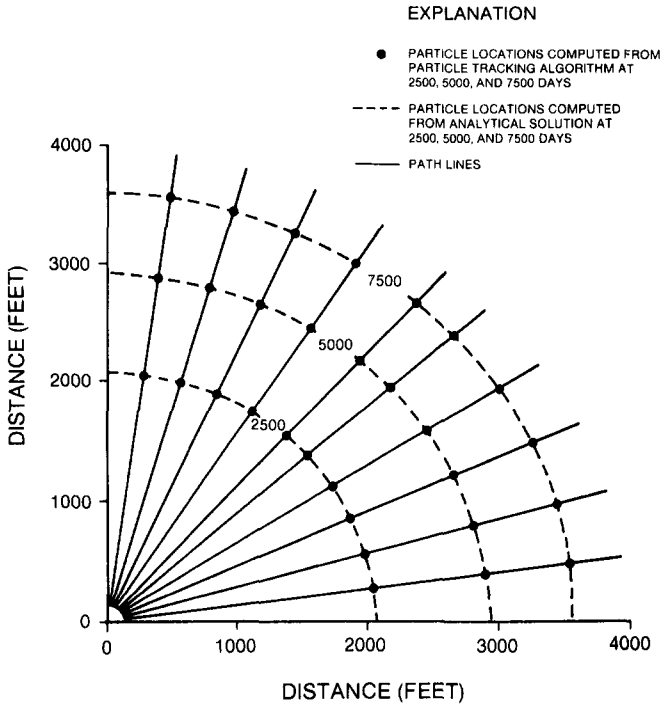

Fig. 7. Comparison of numerically and analytically determined particle locations for radial flow problems.

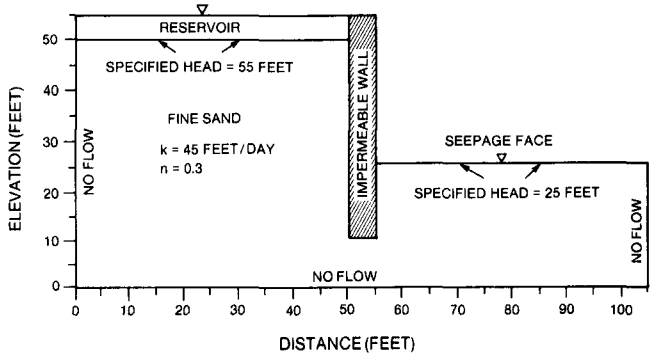

Fig. 8. Schematic description of the impermeable wall problem.

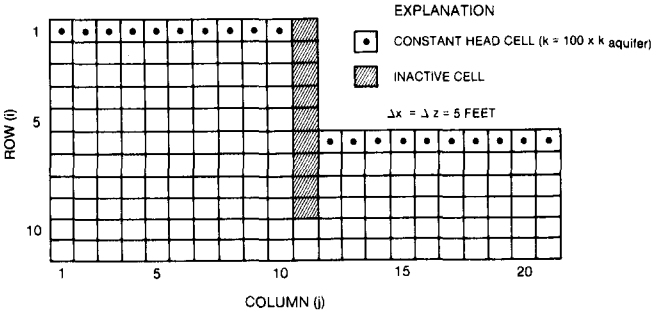

Fig. 9. Finite-difference grid and boundary conditions for the impermeable wall problem.

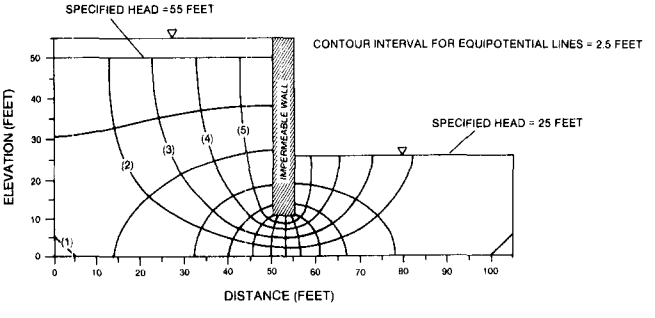

Fig. 10. Path lines and equipotential lines for impermeable wall problem.

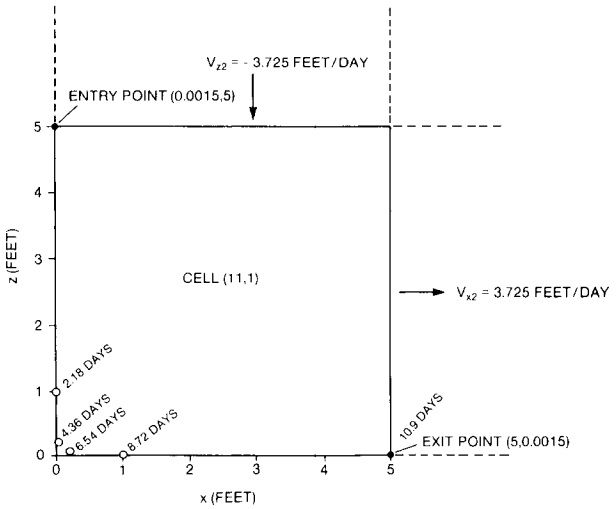

Fig. 11. Details of the path of particle 1 through cell (11,1) for the impermeable wall problem.

$$v_{x1} = 0 \tag{16a}$$

$$v_{x2} = 3.725 \text{ ft/day} \tag{16b}$$

$$v_{z1} = 0 \tag{16c}$$

$$v_{z2} = -3.725 \text{ ft/day} \tag{16d}$$

From inspection of Figure 11, it is apparent that particle 1 will leave the cell across face x_2. Using the x-face velocity components noted above and a cell width of 5 ft in the x-direction, A_x is found to be 0.745 ft/day/ft. Therefore, the x-component of velocity within the cell is given by,

$$v_x = (0.745)\, x \tag{17}$$

The x-component of velocity for particle 1 at the entry point is found to be 0.00112 ft/day. The transit time for the cell can now be computed from equation (13a), and is found to be 10.9 days. A transit time of 10.9 days seems long considering that the magnitudes of the velocity components at faces z_2 and x_2 are approximately 3.7 ft/day. It is apparent that the transit time of nearly 11 days reflects the effects of a curved flow path and the "dead end" corner that are implicitly imbedded within the calculations as a result of the direct integration scheme. This point is further illustrated by dividing the computed cell transit time by 5 and calculating the path line traced by particle 1 for five time steps of 2.18 days in length. The end points for each of these time steps are shown by the open circles on Figure 11. The point at which particle 1 leaves the cell is the same regardless of the number or length of time steps considered. Finally, this example also illustrates how no-flow boundaries are accommodated naturally with no special consideration.

SUMMARY AND CONCLUSIONS

The particle tracking technique described in this report produces an analytical expression for curved particle path lines within individual grid cells. The method can be thought of as "semi-analytical" since the overall path of a particle is the summation of a series of analytically determined path lines for the individual grid cells. An extremely simple and efficient algorithm can be developed for steady-state flow systems whereby a path line is constructed by computing only the exit points for grid cells traversed by a particle. The fact that the method accounts for no-flow boundaries implicitly also contributes to the simple structure of the algorithm. Although the method is completely general and can be applied to transient flow systems, the need to compute particle locations at the end of each finite-difference time step of a transient simulation offsets much of the computational efficiency that can be obtained for steady-state flow systems.

REFERENCES

Freeze, R. A. and J. A. Cherry. 1979. Groundwater. Prentice-Hall, Englewood Cliffs, NJ. 604 pp.

Garabedian, S. P. and L. F. Konikow. 1983. Front-tracking model for convective transport in flowing ground water. USGS Water-Resources Investigations Report 83-4034. 55 pp.

Konikow, L. F. and J. D. Bredehoeft. 1978. Computer model of two-dimensional solute transport and dispersion in ground water. Techniques of Water-Resources Investigations of the U.S. Geological Survey. Book 7, 90 pp.

Mandle, R. J. and A. L. Kontis. 1986. Directions and rates of ground-water movement in the vicinity of Kesterson Reservoir, San Joaquin Valley, California. USGS Water-Resources Investigations Report 86-4196. 57 pp.

McDonald, M. G. and A. W. Harbaugh. 1984. A modular three-dimensional finite-difference ground-water flow model. U.S. Geological Survey Open-File Report 83-875. 528 pp.

Prickett, T. A., T. G. Naymik, and C. G. Lonnquist. 1981. A "random-walk" solute transport model for selected ground-water quality evaluations. Illinois State Water Survey, Champaign, IL. Bulletin 65, 103 pp.

Shafer, J. M. 1987. Reverse pathline calculation of time-related capture zones in nonuniform flow. Ground Water. v. 25, no. 3, pp. 283-289.

Todd, D. K. 1980. Groundwater Hydrology. John Wiley and Sons, New York. 535 pp.

* * * * *

David W. Pollock is a Hydrologist with the Office of Ground Water of the U.S. Geological Survey in Reston, Virginia. He holds a B.S. degree in Geology from the University of Illinois and graduate degrees in Hydrogeology from the University of Minnesota (M.S.) and the University of Illinois (Ph.D.).