A Finite-Volume ELLAM for Three-Dimensional Solute-Transport Modeling

Thomas F. Russell and Caroline I. Heberton
University of Colorado at Denver
Department of Mathematics
P.O. Box 173364, Campus Box 170
Denver, CO 80217-3364
trussell@carbon.cudenver.edu
cheber@math.cudenver.edu

Leonard F. Konikow and George Z. Hornberger
U.S. Geological Survey
Water Resources Division
431 National Center
Reston, VA 20192
lkonikow@usgs.gov
gzhornbe@usgs.gov

ABSTRACT

A three-dimensional finite-volume *ELLAM* method has been developed, tested, and successfully implemented as part of the U.S. Geological Survey (USGS) *MODFLOW*-2000 ground-water modeling package. It is included as a solver option for the Ground-Water Transport process. The *FVELLAM* uses space-time finite volumes oriented along the streamlines of the flow field to solve an integral form of the solute-transport equation, thus combining local and global mass conservation with the advantages of Eulerian-Lagrangian characteristic methods. The USGS *FVELLAM* code simulates solute transport in flowing ground water for a single dissolved solute constituent and represents the processes of advective transport, hydrodynamic dispersion, mixing from fluid sources, and simple chemical reactions. Implicit time discretization of the dispersive and source/sink terms is combined with a Lagrangian treatment of advection, in which forward tracking moves mass to the new time level, distributing mass among destination cells using approximate indicator functions. This allows the use of large transport time increments (large Courant numbers) with accurate results, even for advection-dominated systems (large Peclet numbers). Numerous test cases, including comparisons with analytical solutions and benchmarking against other numerical codes, indicate that the *FVELLAM* can usually yield excellent results, even if relatively few transport time steps are used, although the quality of the results is problem-dependent. The code and documentation (Heberton et al., 2000) can be downloaded from the web site http://water.usgs.gov/nrp/gwsoftware/ as Version 3.5 of *MOC3D* or as *MF2K_GWT*.

INTRODUCTION

The modular finite-difference ground-water flow model (*MODFLOW*) developed by the U.S. Geological Survey (USGS) is a widely used and flexible computer program for simulating three-dimensional ground-water systems (McDonald & Harbaugh, 1988, Harbaugh & McDonald, 1996). *MOC3D* is a solute-transport program that is integrated with *MODFLOW* and has the capability to calculate changes in concentration of a single solute subject to advection, dispersion, diffusion, fluid sources, decay, and retardation (Konikow et al. 1996, Kipp et al. 1998). *MOC3D* solves the solute-transport equation in three dimensions using the method of characteristics, with forward particle tracking to represent advection, coupled with either an explicit or implicit finite-difference method to calculate dispersive flux. This approach is optimal for advection-dominated systems, which are typical of many field problems involving ground-water contamination, as it minimizes numerical dispersion. The model assumes that fluid properties are homogeneous and independent of concentration. The solution techniques, however, do not guarantee a mass balance and also require the use of an areally uniform grid.

A Finite-Volume Eulerian-Lagrangian Localized Adjoint Method (*FVELLAM*) (Healy & Russell, 1993) was developed as an alternative numerical solution algorithm for the *MOC3D* transport model. *ELLAM* (Celia et al. 1990) solves a mass-conservative integral form of the solute-transport equation. The *ELLAM* algorithm uses an implicit time method for dispersion calculations, which allows for large time steps without stability constraints. *ELLAM* uses an Eulerian-Lagrangian advection approach, tracking mass through time and then solving a dispersion equation on a fixed-in-space grid. For advection-dominated problems, this has the advantage of generating less numerical dispersion than standard Eulerian approaches using finite-difference and finite-element methods. *ELLAM* solves integral equations and thus tracks mass associated with fluid volumes, so that it conserves mass locally and globally.

THEORETICAL BACKGROUND AND GOVERNING EQUATIONS

The ground-water flow and interstitial velocity equations used in *MOC3D* are given by Konikow et al., 1996, and will not be repeated here. Solution to the flow equation provides the interstitial velocity field, which couples the solute-transport equation to the ground-water flow equation.

Governing Equation for Solute Transport

The solute-transport equation is that presented in Konikow et al. (1996, eq. 6):

$$\frac{\partial(\varepsilon C)}{\partial t}+\frac{\partial(\rho_b \overline{C})}{\partial t}+\frac{\partial}{\partial x_i}(\varepsilon C V_i)-\frac{\partial}{\partial x_i}\left(\varepsilon D_{ij}\frac{\partial C}{\partial x_j}\right)-\sum C'W+\lambda(\varepsilon C+\rho_b \overline{C})=0 \qquad (1)$$

(summation over repeated indices is understood), where C is volumetric concentration (mass of solute per unit volume of fluid, ML^{-3}), ρ_b is the bulk density of the aquifer material (mass of solids per unit volume of aquifer, ML^{-3}), $\overline{C}$ is the mass concentration of solute sorbed on or contained within the solid aquifer material (mass of solute per unit mass of aquifer material, MM^{-1}), ε is the effective porosity (dimensionless), **V** is a vector of interstitial fluid velocity components (LT^{-1}), **D** is a second-rank tensor of dispersion coefficients (L^2T^{-1}), W is a volumetric fluid sink ($W<0$) or fluid source ($W>0$) rate per unit volume of aquifer (T^{-1}), C' is the volumetric concentration in the sink/source fluid (ML^{-3}), λ is the decay rate (T^{-1}), t is time (T), and x_i are the Cartesian coordinates (L).

The terms controlling sorption are combined into a single parameter--the retardation factor (R_f)--assumed to be constant in time because on a linear isotherm, $C/\overline{C}$ is constant. The retardation factor is defined as:

$$R_f = 1+\frac{\rho_b \overline{C}}{\varepsilon C}. \qquad (2)$$

An integral form of the solute-transport equation, which is a statement of conservation of mass over the domain of integration, is the governing equation for this finite-volume *ELLAM* approach. Integration against a "test function" is used to provide the formulation of conservation of mass, including treatment of cell or subdomain boundary conditions and solute decay.

The test function effectively specifies the domain of integration for the transport equation by the portion of the space-time domain where its value is nonzero. On a subdomain of integration, the test function can be seen as an integration weight at each point. Varying the weight along streamlines of the flow is a convenient mechanism to provide solute growth or decay.

If we divide eq. 1 by R_f, multiply by a test function u, integrate over time and space, and assume R_f is constant in time, we have:

$$\int_{\Omega}\int_{0}^{T}\left(u\frac{\partial(\varepsilon C)}{\partial t}+\frac{u}{R_f}\nabla\cdot(\varepsilon C\mathbf{V}-\varepsilon\mathbf{D}\nabla C)-\frac{u}{R_f}\sum C'W+u\lambda\varepsilon C\right)\mathrm{d}t\mathrm{d}x=0 \tag{3}$$

where Ω is the entire spatial domain, and T is the end of the simulation time period starting at time zero.

Equation 3 is integrated by parts using

$$u\frac{\partial(\varepsilon C)}{\partial t}=\frac{\partial(u\varepsilon C)}{\partial t}-\frac{\partial u}{\partial t}\varepsilon C \tag{4}$$

and

$$\frac{u}{R_f}\nabla\cdot(\varepsilon C\mathbf{V}-\varepsilon\mathbf{D}\nabla C)=\frac{1}{R_f}\nabla\cdot(u\varepsilon C\mathbf{V}-u\varepsilon\mathbf{D}\nabla C)-\frac{1}{R_f}\nabla u\cdot(\varepsilon C\mathbf{V}-\varepsilon\mathbf{D}\nabla C) \tag{5}$$

to yield the global equation,

$$\int_\Omega \int_0^T \left(\frac{\partial (u\varepsilon C)}{\partial t} + \frac{1}{R_f} \nabla \cdot (u\varepsilon C \mathbf{V} - u\varepsilon \mathbf{D} \nabla C) + \frac{1}{R_f} \nabla u \cdot \varepsilon \mathbf{D} \nabla C - \frac{u}{R_f} \sum C' W \right.$$

$$\left. - \varepsilon C \left(\frac{\partial u}{\partial t} + \frac{\mathbf{V}}{R_f} \cdot \nabla u - \lambda u \right) \right) dt\, d\mathbf{x} = \mathbf{0} \tag{6}$$

The Eulerian-Lagrangian aspects of the method derive from the requirement that the test function satisfy the adjoint equation, $\frac{\partial u}{\partial t} + \frac{\mathbf{V}}{R_f} \cdot \nabla u - \lambda u = 0$. Thus, for the time step from t^n to t^{n+1}, use u of the form $u(\mathbf{x},t) = f(\mathbf{x},t) e^{-\lambda (t^{n+1} - t)}$, where $\frac{\partial f}{\partial t} + \frac{\mathbf{V}}{R_f} \cdot \nabla f = 0$, so that f is constant along characteristics of the retarded interstitial velocity field. Note that with $u = e^{-\lambda (t^{n+1} - t)}$ (that is, $f = 1$) in the following for $t^n \le t \le t^{n+1}$, and $u = 0$ otherwise, we arrive at a statement of global conservation of mass for a time step:

$$\int_\Omega (u\varepsilon C)^{n+1} d\mathbf{x} - \int_\Omega (u\varepsilon C)^n d\mathbf{x} + \int_\Omega \int_{t^n}^{t^{n+1}} \left(\frac{1}{R_f} \nabla \cdot (u\varepsilon C \mathbf{V} - u\varepsilon \mathbf{D} \nabla C) \right.$$

$$\left. + \frac{1}{R_f} \nabla u \cdot \varepsilon \mathbf{D} \nabla C - \frac{u}{R_f} \sum C' W \right) dt\, d\mathbf{x} = 0 \tag{7}$$

For a local conservation equation on each finite-difference cell Ω_l in the domain, let

$$u_l(\mathbf{x},t) = f_l(\mathbf{x},t) e^{-\lambda (t^{n+1} - t)} \tag{8}$$

where $f_l(\mathrm{x},t^{n+1}) = 1$ on Ω_l and $f_l(\mathrm{x},t^{n+1}) = 0$ elsewhere.

We thus arrive at a system of local *ELLAM* equations,

$$\int_{\Omega_l} (\varepsilon C)^{n+1}\,\mathrm{d}\mathbf{x} \;+\; \iint_{\mathrm{supp}\,u_l \cap \Gamma^{n+1}} \frac{e^{-\lambda\left(t^{n+1}-t\right)}}{R_f}\left(\varepsilon C\mathbf{V} - \varepsilon\mathbf{D}\nabla C\right)\cdot\mathbf{n}\,\mathrm{d}t\,\mathrm{d}\mathbf{s}$$

$$-\;\int_{t^n}^{t^{n+1}}\int_{\partial\,\mathrm{supp}\,u_l} \frac{e^{-\lambda\left(t^{n+1}-t\right)}}{R_f}\left(\varepsilon\mathbf{D}\nabla C\right)\cdot\mathbf{n}\,\mathrm{d}t\,\mathrm{d}\mathbf{s} \;-\; \iint_{\mathrm{supp}\,u_l \cap \mathrm{supp}\,W} \frac{e^{-\lambda\left(t^{n+1}-t\right)}}{R_f}\sum C'W\,\mathrm{d}t\,\mathrm{d}\mathbf{x} \qquad (9)$$

$$=\; e^{-\lambda\Delta t}\int_{\Omega_l^*} (\varepsilon C)^n\,\mathrm{d}\mathbf{x}$$

The notations introduced in eq. 9 are explained as follows. First, *supp f* denotes the support of a function *f*, that is, the part of its domain on which it is nonzero; for example, *supp W* is the union of all finite-difference cells containing a source or sink having a nonzero flow rate. If S is a subset of the spatial domain Ω, then ∂S denotes its boundary, and S^* represents its pre-image at time t^n under advection along characteristics; that is, $\mathbf{x}\in S^*$ whenever the characteristic that starts at x at time t^n and tracks forward in time lands in S at time t^{n+1}. Next, $\Gamma^{n+1} \equiv \partial\Omega\times\left(t^n, t^{n+1}\right)$ denotes the space-time boundary over the current time step. If B is a subset of Γ^{n+1}, the pre-image B^* can again be defined as a subset of Ω; $\mathbf{x}\in B^*$ whenever the forward characteristic from $(\mathbf{x},t^n)$ meets the outflow boundary somewhere in B. Finally, **n** is the outward unit normal vector on the specified boundary, and $d\mathbf{x}$ and $d\mathbf{s}$ signify differential volume and boundary area, respectively.

Note that eq. 9 has the form of space-time integrals of dispersion equations. *ELLAM* can be viewed as a method of characteristics, tracking mass along streamlines of the flow to accumulate data to the right-hand side of the system of equations.

Review of Assumptions

As described by Konikow et al. (1996), a number of assumptions have been made in the development of the governing equations. Following is a list of the main assumptions for review:

1. Darcy's law is valid and hydraulic-head gradients are the only significant driving mechanism for fluid flow.
2. The hydraulic conductivity of the aquifer system is constant with time. Also, if the

system is anisotropic, it is assumed that the principal axes of the hydraulic-conductivity tensor are aligned with the coordinate system of the grid, so that the cross-derivative terms of the hydraulic-conductivity tensor are eliminated.

3. Gradients of fluid density, viscosity, and temperature do not affect the velocity distribution.
4. Chemical reactions do not affect the fluid or aquifer properties.
5. The dispersivity coefficients are constant over a flow time step, and the aquifer is isotropic with respect to longitudinal dispersivity.

As noted by Konikow and Bredehoeft (1978), the nature of a specific field problem may be such that not all of these underlying assumptions are valid. The degree to which field conditions deviate from these assumptions will affect the applicability and reliability of the model for that problem. If the deviation from a particular assumption is significant, the governing equations and the numerical simulator may have to be modified to account for the appropriate processes or factors.

NUMERICAL METHODS

Gridding of the spatial domain Ω into three-dimensional Cartesian cells with possibly variable thickness is described in detail by Heberton et al. (2000), and will not be repeated here. The conventions conform to those of *MODFLOW*.

Ground-Water Flow Equation

A numerical solution of the three-dimensional ground-water flow equation is obtained by the *MODFLOW* code using implicit (backward-in-time) finite-difference methods. Successful use of *MOC3D* requires a thorough familiarity with the use of *MODFLOW*. Comprehensive

documentation of *MODFLOW* is presented by McDonald and Harbaugh (1988), Harbaugh and McDonald (1996a and 1996b), and the various reports for additional implemented packages and modules.

Average Interstitial Velocity

The solution of the transport equation requires knowledge of the velocity (or specific discharge) field. Therefore, after the head distribution has been calculated for a given time step or steady-state flow condition, the specific discharge across every face of each finite-difference cell within the grid is calculated using a finite-difference approximation (see Konikow et al., 1996).

The mass-tracking algorithm requires that the seepage velocity at any point within a cell be defined to compute advective transport. It is calculated at points within a finite-difference cell based on linearly interpolated estimates of specific discharge at those points divided by the effective porosity of the cell (see Konikow et al., 1996).

Solute-Transport Equation

The mathematical properties of the transport equation vary depending upon which terms in the equation are dominant in a particular system. Where solute transport is dominated by advection, as is common in many field problems, the transport equation resembles a hyperbolic type of equation (similar to equations that describe the propagation of a wave or of a shock front). In contrast, where a system is dominated by dispersive and diffusive fluxes, such as might occur where fluid velocities are relatively low and aquifer dispersivities are relatively high, the transport equation becomes more parabolic in nature (similar to the transient ground-water flow equation). Because system properties and fluid velocity may vary significantly, the dominant process (and the mathematical properties of the governing equation) may vary from point to point and over time within the domain of simulation. The challenge for a numerical

solution method is to represent solute fronts without introducing either erroneous oscillations or nonphysical dispersion, and to do so without requiring a cost-prohibitive use of computer resources.

ELLAM

The *ELLAM* method advances in time by considering the space-time domain $\Omega \times (t^n, t^{n+1})$. Part of this domain flows out of the spatial domain during the time step, and separate outflow boundary equations below address this part. The remaining part is modeled using cell integral equations that do not contain outflow terms. The next two subsections present these equations in turn. Overall, they can be coupled in such a way that the outflow concentrations contribute to the right-hand side of the system of cell integral equations.

Like the algorithms in the previous versions of *MOC3D, ELLAM* approximates total solute flux across the domain boundary by the advective flux; that is, the dispersive flux is neglected. This is not required by *ELLAM* methods in general, but rather is a feature of this particular implementation. As a result, boundary-face concentrations are not coupled to cell-center concentrations through a dispersive concentration gradient, simplifying the separation of outflow boundary equations from cell integral equations. All mass moving across the domain boundary can be tracked by the advective algorithm, and at outflow boundaries, this mass provides data for the outflow boundary equations. Solutions of these outflow equations, together with user-input inflow concentrations, appear on the right-hand side of the cell integral equations, which represent local mass conservation on finite-difference cells of the domain.

ELLAM approaches the hyperbolic-parabolic nature of the solute-transport equation by combining a method of characteristics technique for advection with a backward Euler in time and centered differences in space solution to a diffusion equation. The following sections describe key elements of the numerical methods in more detail.

Cell Integral Equations

Taking the conservation of mass equation for each cell (eq. 9), we approximate the total boundary flux with advective flux (that is, the dispersive flux is taken to be zero) and the dispersion time integral with a backward Euler formulation, and then rearrange terms. The system of equations to be solved is then:

$$\begin{aligned}
&\int_{\Omega_l} (\varepsilon C)^{n+1} d\mathbf{x} - \Delta t \int_{\partial\Omega_l} \frac{1}{R_f} (\varepsilon \mathbf{D} \nabla C)^{n+1} \cdot \mathbf{n} d\mathbf{s} = \\
&\quad e^{-\lambda \Delta t} \int_{\Omega_l^*} (\varepsilon C)^n d\mathbf{x} \\
&- \iint\limits_{\text{supp}\, u_l \cap \Gamma^{n+1}} e^{-\lambda (t^{n+1} - t)} \varepsilon C_{\text{inflow}} \frac{\mathbf{V}}{R_f} \cdot \mathbf{n} dt d\mathbf{s} \\
&+ \iint\limits_{\text{supp}\, u_l \cap \text{supp}\, W} e^{-\lambda (t^{n+1} - t)} \sum C' \frac{W}{R_f} dt d\mathbf{x}
\end{aligned} \tag{10}$$

where Ω_l^* means the pre-image in the spatial domain at t^n of Ω_l at t^{n+1}; $\Gamma^{n+1} \equiv \partial\Omega \times (t^n, t^{n+1})$ is the space-time boundary at time step n+1; and $supp\, f \equiv \{x |\ f(x) \neq 0\}$. These integral equations are solved for C^{n+1}, the concentration at the new (n+1) time level at each cell center. Note that the right-hand side of eq. 10 consists of advective mass contributions from storage (that is, advection of mass in the domain at the start of the time increment), inflow boundaries, and sources. This is illustrated schematically for a simple case having constant velocity in figure 1, which shows how mass is advected into cell Ω_l at time level t^{n+1} from an inflow boundary, from a fluid source in a nearby cell, and from the mass present at time level t^n in nearby cells.

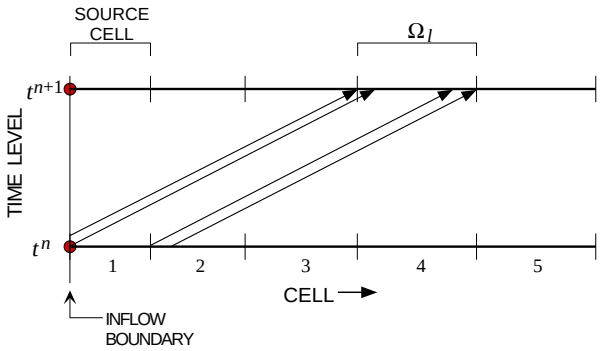

Figure 1. Schematic representation (for a simple case having constant velocity) showing how mass is advected into cell Ω_l at time level t^{n+1} from the inflow boundary, from a fluid source in cell 1, and from storage at time level t^n in cells 1 and 2.

Outflow Boundary Equations

The integral in eq. 7 expressing mass crossing the domain boundary during a time step is:

$$\int_{\Omega}\int_{t^n}^{t^{n+1}} \frac{1}{R_f}\nabla\cdot\left(u\varepsilon C\mathbf{V} - u\varepsilon\mathbf{D}\nabla C\right)dt\,d\mathbf{x} = \int_{t^n}^{t^{n+1}}\int_{\partial\Omega}\frac{1}{R_f}\left(u\varepsilon C\mathbf{V} - u\varepsilon\mathbf{D}\nabla C\right)\cdot\mathbf{n}\,d\mathbf{s}\,dt \tag{11}$$

Considering just the outflow portion of the boundary, this becomes

$$\begin{aligned}&\int_{t^n}^{t^{n+1}}\int_{(\partial\Omega)_{\text{outflow}}}\frac{1}{R_f}\left(u\varepsilon C\mathbf{V} - u\varepsilon\mathbf{D}\nabla C\right)\cdot\mathbf{n}\,d\mathbf{s}\,dt \\ &\approx \int_{(\partial\Omega)_{\text{outflow}}}\int_{t^n}^{t^{n+1}}\left(u\varepsilon\, C_{\text{outflow}}\frac{\mathbf{V}}{R_f}\right)\cdot\mathbf{n}\,dt\,d\mathbf{s}\end{aligned} \tag{12}$$

where total flux across the boundary is now approximated by advective flux.

We index the outflow boundary faces with *ll* and define the following test functions:

$$u_{ll} = \begin{cases} e^{-\lambda\left(t^{n+1}-t\right)} & \text{on characteristics from } \Omega \\ & \text{at time level } n \text{ into boundary} \\ & \text{area } (\partial\Omega)_{ll} \text{ at any time} \\ & \text{during time step} \\ \\ \\ \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

The mass across outflow boundary face *ll* is the mass stored at the previous time level that flows across the face, together with any inflow and source mass that both enters the domain and leaves through face *ll* during the time step.

Taking $u = u_{ll}$ on the right-hand side of eq. 12, and including those terms from eq. 7 that are appropriate in the context of the outflow boundary, we can write three terms representing mass contributions from storage, inflow, and sources, as follows:

$$\begin{aligned} \int_{(\partial\Omega)_{ll}} \int_{t^n}^{t^{n+1}} \left(e^{-\lambda\left(t^{n+1}-t\right)} \varepsilon C_{\text{outflow}} \frac{\mathbf{V}}{R_f} \right) \cdot \mathbf{n}\, dt\, ds &= e^{-\lambda \Delta t} \int_{(\partial\Omega)_{ll}^*} (\varepsilon C)^n d\mathbf{x} \\ &- \iint\limits_{\text{supp } u_{ll} \cap \partial\Omega_{\text{inflow}} \times \left(t^n, t^{n+1}\right)} e^{-\lambda\left(t^{n+1}-t\right)} \varepsilon C_{\text{inflow}} \frac{\mathbf{V}}{R_f} \cdot \mathbf{n}\, dt\, d\mathbf{s} \\ &+ \iint\limits_{\text{supp } u_{ll} \cap \text{supp W} \times \left(t^n, t^{n+1}\right)} e^{-\lambda\left(t^{n+1}-t\right)} \sum C' \frac{W}{R_f}\, dt\, d\mathbf{x} \end{aligned} \tag{14}$$

where $(\partial\Omega)_{ll}$ is a discretized portion of the boundary face, and $(\partial\Omega)_{ll}^*$ is its pre-image, in the manner described following eq. 9. The advection of mass to an outflow boundary is illustrated schematically for a simple case having constant velocity in figure 2, where mass is advected to

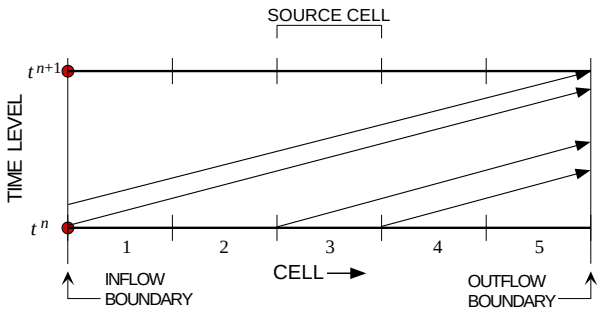

Figure 2. Schematic representation showing how mass is advected to an outflow boundary at time level t^{n+1} from the inflow boundary, from a fluid source in cell 3, and from storage at time level t^n in all five cells.

an outflow boundary at time level t^{n+1} from an inflow boundary, from a fluid source in a nearby cell, and from the mass present at time level t^n in nearby cells.

ELLAM equations are a formulation of mass conservation on each cell. Therefore, approximations to concentrations result that conserve mass locally (on each finite-difference cell) and globally (on the entire domain).

Mass Tracking

For each cell in the fixed finite-difference grid, the integrals on the right-hand side of eq. 10 represent solute mass advected into the cell during the time step from storage (that is, advection of mass in the domain at the start of the time increment), the domain boundary, or a fluid source, respectively.

Advection in flowing ground water is simulated by mass tracking along the characteristic curves determined by the seepage velocity. Calculation of advective movement during a flow time step is based on the specific discharges computed at the end of the step.

As in *MOC3D*, tracking is performed using linear interpolation of velocity in the direction of the component of interest and piecewise-constant interpolation in the other two

directions. The approach is to solve a system of three ordinary differential equations to find the characteristic curves [$x = x(t)$, $y = y(t)$, and $z = z(t)$] along which fluid is advected:

$$\frac{dx}{dt} = \frac{V_x}{R_f} \tag{15}$$

$$\frac{dy}{dt} = \frac{V_y}{R_f} \tag{16}$$

$$\frac{dz}{dt} = \frac{V_z}{R_f} \tag{17}$$

This is accomplished by introducing a set of moving points that can be traced within the stationary coordinates of a finite-difference grid. Each point corresponds to one characteristic curve, and values of x, y, and z are obtained as functions of t for each characteristic (Garder et al., 1964). Each point moves through the flow field by the flow velocity acting along its trajectory.

The *ELLAM* equations, eqs. 10 and 14, suggest that mass is tracked backwards along characteristics to the pre-image of each cell or boundary face. It is not possible, however, to exactly locate all of the mass at the previous time level by backtracking a finite number of points (see figure 3). In order to achieve mass balance, this implementation of the *ELLAM* algorithm tracks the known mass distribution forward from the old time level to the new time level (see figure 4). The accuracy of point tracking can be related to the Courant number, which is the ratio of (1) the distance a point will move in one time increment (velocity times Δt) to (2) the grid spacing (Δx).

ELLAM tracks points that are the centers of volumes of fluid. Thus, mass in a fluid volume is tracked under advection during a time step, distributed among destination cells, and accumulated to the right hand side storage, inflow, or source integral for each cell.

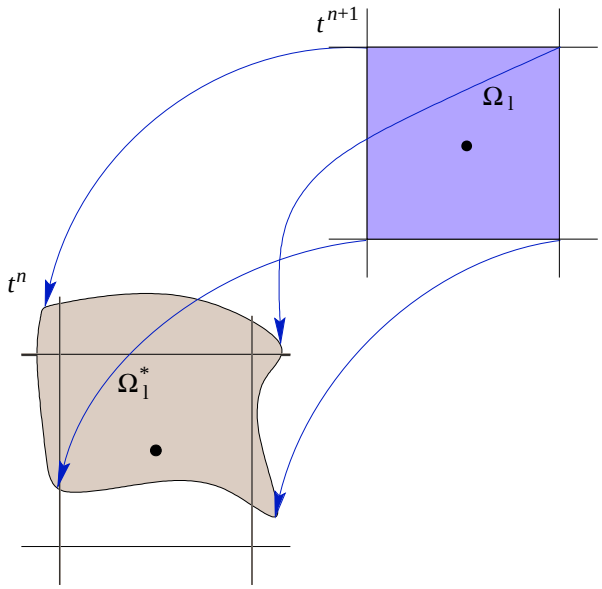

Figure 3. Two-dimensional example illustrating that pre-image of a cell may be irregularly shaped and not easily defined by backtracking from t^{n+1} to t^n.

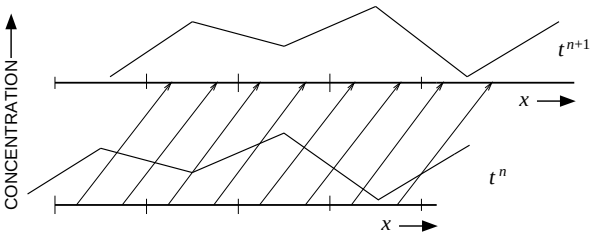

Figure 4. Plots of solute concentration versus distance that illustrate one-dimensional advection of known mass distribution from old time level to new time level (Courant number = 1).

Decay

When simulating linear decay, all mass in the system at the beginning of each transport time step is decayed over the entire time step by a factor $e^{-\lambda \Delta t}$, where λ is the decay rate. Inflow

and source mass are decayed in the same way, where the time interval is now not the entire time step, but the part of it during which new mass is in the domain.

This decay algorithm has no numerical stability restrictions associated with it. If the half-life is on the order of, or smaller than, the transport time step, however, some accuracy will be lost.

When a solute subject to decay enters the aquifer through a fluid source, it is assumed that the fluid source contains the solute at the concentration specified by C'. *MOC3D* allows decay to occur only within the ground-water system, and not within the source reservoir. In other words, for a given stress period, C' remains constant in time. If the problem being simulated requires that the solute in the source fluid itself undergo decay, then the source code will have to be modified.

Numerical Integration

The numerical treatment of each term in eqs. 10 and 14 is discussed in detail by Heberton et al. (2000) and will be summarized briefly here. The equations are first divided through by porosity, which is represented by piecewise constants in space and time. This is valid because there are no spatial derivatives of porosity in the local *ELLAM* equations. The first term in eq. 10 is evaluated with the trapezoidal rule on each cell octant. Concentrations at octant corners are weighted averages of neighboring node concentrations, determined by trilinear interpolation. The second (dispersion) term involves centered differences as in *MOC3D* (Konikow et al., 1996), modified for varying grid dimensions.

For the third term, numerical forward tracking (see the section "Mass Tracking") is used, in which each cell is divided into subcells, where the parameters *NSC, NSR, NSL* specify the number of subdivisions in the column, row, and layer direction, respectively. A moving point starts at the center of each subcell at time level n. Depending on where the point lands at the new time, the subcell mass may be distributed among cells neighboring the destination cell using

"approximate test functions," w_l, that add up to 1. This distribution prevents small velocity perturbations from causing large changes in concentration. This yields the formulation of the third term in eq. 10,

$$e^{-\lambda\Delta t}\int_{\Omega_l^*} C^n \mathrm{d}\mathbf{x} = e^{-\lambda\Delta t}\sum_{j,i,k}\sum_{\substack{\text{p=subcell}\\ \text{center}}} \frac{\Delta x_j \Delta y_i b_{j,i,k}}{(NSC)(NSR)(NSL)} w_l\left(p^f\right) C^n(p) \tag{18}$$

where summation runs through all subcells of each cell in the domain, b is the cell thickness, and p^f is the image of p under forward tracking to the new time level. Examples of approximate test functions are illustrated in figure 5 for one direction. An approximate test function is determined

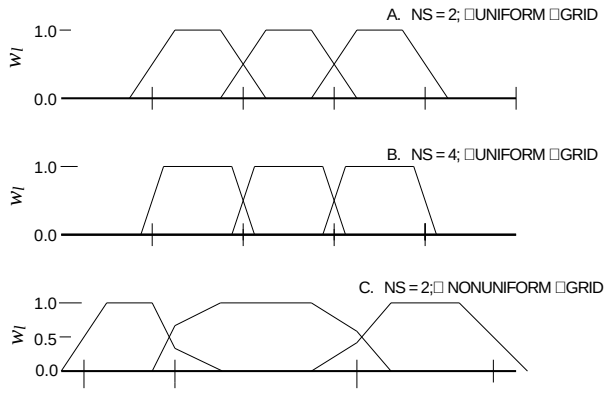

Figure 5. Examples of spatial distribution of approximate test functions (w_l) for selected one-dimensional cases. Vertical ticks represent cell boundaries.

by *NSC*, *NSR*, and *NSL*, the proximity of the domain boundary, and the active status of neighboring cells. Mass is not split across the boundary or into inactive cells.

In the fifth term, a source or sink is distributed uniformly over the cell containing it. For a source cell, a time step is discretized into a number *NT* of sub-time steps, integrating in time with the composite trapezoidal rule. Source mass is tracked forward in the grid, just like mass that began the time step in the system, but starting at the sub-step when it enters the system. Sink concentration is assumed to be the average nodal concentration for the transport time step,

except in the case of a sink due to evapotranspiration, where it is taken to be zero. Integration rules are midpoint in space and a one point backward Euler in time. The fourth (inflow boundary) term is treated in a manner similar to a source, with sub-time steps.

At the boundary, like previous versions of *MOC3D*, this *ELLAM* implementation approximates total solute flux by the advective flux (not required by *ELLAM* in general). Hence there is no distinction between total-flux and specified-concentration inflow boundary conditions, and the outflow boundary condition is zero dispersive flux, which causes the analogue in eq. 14 of the second term in eq. 10 to vanish. The first term in eq. 14 is approximated with the midpoint rule in space and backward Euler in time, and the other terms are treated like their analogues in eq. 10. Outflow-boundary-face concentrations are unknowns, but are not coupled to cell-center concentrations through the numerical derivative because of the absence of dispersive flux. Mass tracked across outflow boundaries provides data for a system of associated outflow boundary equations decoupled from the cell equations. User-input inflow concentrations together with the outflow solutions then accumulate on the right-hand side of the system of cell equations.

Accuracy Criteria

An accuracy criterion incorporated in *MOC3D* constrains the distance that solute mass is advected during each transport time step. A restriction can be placed on the size of the time step to ensure that the number of grid cells a point moves in the *x*-, *y*-, or *z*-directions does not exceed some maximum. The simulator allows the user to specify this maximum (named *CELDIS* in the code and input instructions). This translates into a limitation on the transport time-step length. If the time step used to solve the flow equation exceeds the time limit, the flow time step will be subdivided into an appropriate number of equal-sized smaller time increments for solving transport.

For advective transport, at least four grid nodes across a solute front (or zone of relatively steep concentration gradient) are needed for good accuracy. Hence, the profile of a concentration peak should be represented across at least eight nodes of the grid for good accuracy. In such test results, a peak concentration can be advected with very small dissipation per time step of the maximum for a variety of Courant numbers. With insufficient mesh, a peak will decay rapidly for an initial period of time as it spreads and oscillates; thereafter, the numerical decay is very slow and the oscillations do not worsen. A fine discretization of tracked mass (large *NSC*, *NSR*, *NSL*) reduces the rate of peak decay when modeling with many transport time steps. Regardless of the solution accuracy, global mass is conserved. Mass balance errors of less than 10^{-4} percent can generally be expected.

Dispersion accuracy is governed in part by the central-difference approximations to the space derivatives; a finer mesh is more accurate. The implicit time-stepping is unconditionally stable. This allows for large time steps during the simulation. Because *ELLAM* solves the dispersion equation along characteristics, thus avoiding large values of time derivatives of the solution at passage of a steep front, temporal errors are small compared to a standard finite-difference solution to an advection-diffusion equation, and large, accurate, stable time steps are possible, though the accuracy of the solution to the dispersion equation still decreases as the time step size increases. However, smaller time steps in order to reduce this error could result in a loss of peak to interpolations performed at each advective step, an effect that can be reduced by increasing *NSC*, *NSR*, and *NSL*.

One additional difficulty encountered with implicit temporal differencing results from the use of a symmetric spatial differencing for the cross-derivative terms of the dispersion tensor. This creates a potential for overshoot and undershoot in the calculated concentration solution, particularly when the velocity field is oblique to the axes of the grid. A remedy for excessive overshoot and undershoot is to refine the finite-difference mesh. This may, however, increase simulation times.

ELLAM can produce qualitatively good results in a small number of time steps, provided the *NT* value is sufficient to yield smooth distribution of mass along the inflow path. (See sections "Special Problems" and "Guidance on Input Parameter Values".)

Special Problems

Fronts too sharp for the given mesh density (grid spacing) may produce negative concentration values and/or numerical dispersion. For the mass in one cell to be positive, the mass in the adjacent cell to be zero, and concentration to vary linearly, the cell with zero mass may show a negative concentration. For non-integer Courant numbers, numerical dispersion results from the solution algorithm being insensitive to the exact location of advected mass in a destination cell. On a well-discretized front, these effects are minimal due to error cancellation. Thus, solving on a fine grid with few time steps may mitigate these difficulties.

Numerical dispersion may also result from tracking subdivisions of mass that are too coarse. The level of discretization of mass tracked and accumulated to the right-hand side vector is determined by parameters, *NSC*, *NSR*, and *NSL*. These parameters define the number of subcells in the column, row, and layer direction, respectively. To increase the resolution of mass tracking under advection, it may be desirable to increase the values of these parameters.

Parameter *NT* defines the number of sub-time steps per transport time increment. *NT* should be large enough so that all cells in the path of flow from the inflow boundary or source to the location of the front at the end of the time step receive incoming mass. This is to avoid artificial mass lumping. See section "Guidance on Input Parameter Values."

To avoid non-physical accumulation of mass at an outflow boundary, the spatial *NS* parameter in the direction normal to the boundary must be such that $1/(2NS)<$Courant. This is to ensure that at least some mass is calculated by the algorithm as reaching the boundary during a time step.

Extreme variation in cell thickness among neighboring cells in a layer may adversely affect model results. This is caused by the inherent geometric inconsistency in the vertical direction between adjacent cells that have different thicknesses. (Also see McDonald and Harbaugh, 1988, figure 9 and related discussion.)

Review of *ELLAM* Assumptions

The assumptions that have been incorporated into the *ELLAM* simulator are very similar to those for *MOC3D* Version 1 and Version 2. They are relevant to both grid design and model application. Efficient and accurate application of *ELLAM* requires the user to be aware of these assumptions. Therefore, the user should review the description of these items as presented by Konikow et al. (1996).

Domain boundaries are assumed to be far enough from the plume that any errors in the treatment of the boundaries will not have a significant effect on the solution. The boundary condition is that the normal component of the concentration gradient on the boundary is zero, meaning there is no dispersive flux across the domain boundary.

Unlike the previous *MOC3D* explicit and implicit difference approximations, *ELLAM* does <u>not</u> require a uniform grid spacing within the domain. Likewise, there is no longer a formal restriction on variations in the product of porosity and thickness within the domain. *ELLAM* <u>does</u> assume:

- Concentration at an outflow boundary face at the new time level is well approximated by the mass crossing the face during the time step divided by the fluid volume across the face.
- Mass in or out of the domain during a time step via a source or sink cell is well approximated by the average nodal concentration during the time step times the fluid volume through the source or sink. Mass loss in flow through upstream sinks is negligible.

- Cell thicknesses are smoothly varying within a horizontal layer.

Guidance on Input Parameter Values

Discretization parameters *NSC*, *NSR*, *NSL*, and *NT* must all be powers of two. In each case, the input parameter specified by the user is the exponent: *NSCEXP*, *NSREXP*, *NSLEXP*, and *NTEXP*, respectively.

In general, use *NSC* = *NSR* = *NSL* = 4 (and therefore *NSCEXP* = *NSREXP* = *NSLEXP* = 2) except if modeling a one- or two-dimensional problem. Here, a value of 4 is only needed in the dimension(s) of the problem, with *NS* values of 2 adequate in the missing direction(s). *NS* values greater than four may be useful when modeling with a complicated velocity field, or with numerous transport time steps, or to improve accuracy near a boundary. Computational efficiency is strongly related to the values of the *NS* parameters, but the impact of changing *NS* values is highly problem dependent.

The number of discrete mass-bearing volumes entering the domain during a time step is *NT*+1. This number must be large enough so that each grid cell, from the one at the first boundary to the one where mass entering at the beginning of that time step is advected, can receive a portion of the discretized inflow mass. If the solution shows mass becoming distributed down the flow path in clumps (this is illustrated schematically in figure 6 for a case in

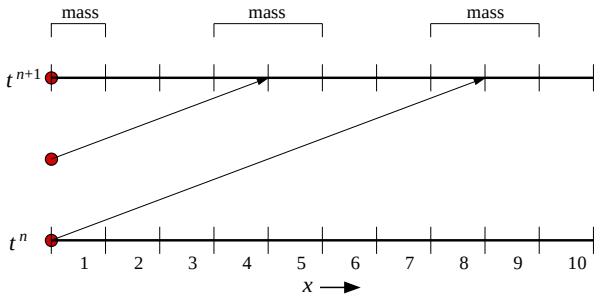

Figure 6. Schematic representation of tracking mass from an inflow boundary. For a Courant number of 8 and NT = 2, not every cell along the inflow path receives mass.

which two sub-time steps are used), increase *NTEXP* (thus increasing *NT*). Increasing *NT* will smooth the front to the point of ensuring an even distribution of mass among destination cells. *NT* only comes into effect for problems with concentrations entering via fluid sources or inflow across boundaries of the grid.

MODEL TESTING AND EVALUATION

The *ELLAM* simulator was tested and evaluated by running the same suite of test cases as was applied to *MOC3D* Version 1 by Konikow et al. (1996) and *MOC3D* Version 2 by Kipp et al. (1998). This suite includes results generated by analytical solutions and by other numerical models. It spans a range of conditions and problem types so that the user will gain an appreciation for both the strengths and weaknesses of this particular code. It should be noted that all test cases involve steady flow conditions. One test case evaluated *ELLAM* for a relatively simple system involving one-dimensional solute transport in a finite-length aquifer having a third-type source boundary condition, as described by Konikow et al. (1996). The numerical results were compared to an analytical solution by Wexler (1992, p. 17). Details of those results were reported by Heberton et al. (2000). The multi-dimensional cases will be included here.

Uniform Flow, Three-Dimensional Transport

To evaluate and test *ELLAM* for three-dimensional cases, we compared numerical results with those of the analytical solution developed by Wexler (1992) for the case of three-dimensional solute transport from a continuous point source in a steady, uniform flow field in a homogeneous aquifer of infinite extent. Konikow et al. (1996) note that this evaluation primarily is a test of the accuracy of the calculated dispersive flux in three directions because the flow field is aligned with the grid. The problem and analytical solution are described in detail by Konikow et al. (1996, p. 45-48); the parameters and boundary conditions for this test case are summarized in table 1. This case

Table 1. Base-case parameters used in *ELLAM* simulation of transport from a continuous point source in a three-dimensional, uniform, steady-state flow system

Parameter	Value
$T_{xx} = T_{yy}$	0.0125 m^2/day
ε	0.25
α_L	0.6 m
α_{TH}	0.03 m
α_{TV}	0.006 m
PERLEN (length of stress period)	400 days
V_y	0.1 m/day
$V_x = V_z$	0.0 m/day
Initial concentration (C_0)	0.0
Source concentration (C')	2.5×10^6 g/m^3
Q (at well)	1.0×10^{-6} m^3/d
Source location	Column = 1, Row = 8, Layer = 1
Number of rows	30
Number of columns	12
Number of layers	40
DELR (Δx)	0.5 m
DELC (Δy)	3.0 m
Layer thickness (Δz)	0.05 m
CELDIS	1.0
NSCEXP	2
NSREXP	2
NSLEXP	2
NTEXP	2

also represents a test of the ability of the algorithm to represent the effects of a solute source at a specified flux boundary condition.

The results of *ELLAM* are compared graphically in figure 7 with those of the analytical solution for the *x*-*y* plane passing through the point source. Figure 7a shows the concentrations in this plane at t = 400 days as calculated using the analytical solution. Also shown, in figures 7b-d, are the *ELLAM* solutions using *CELDIS* = 7 (two transport time increments), *NSC* = *NSR*

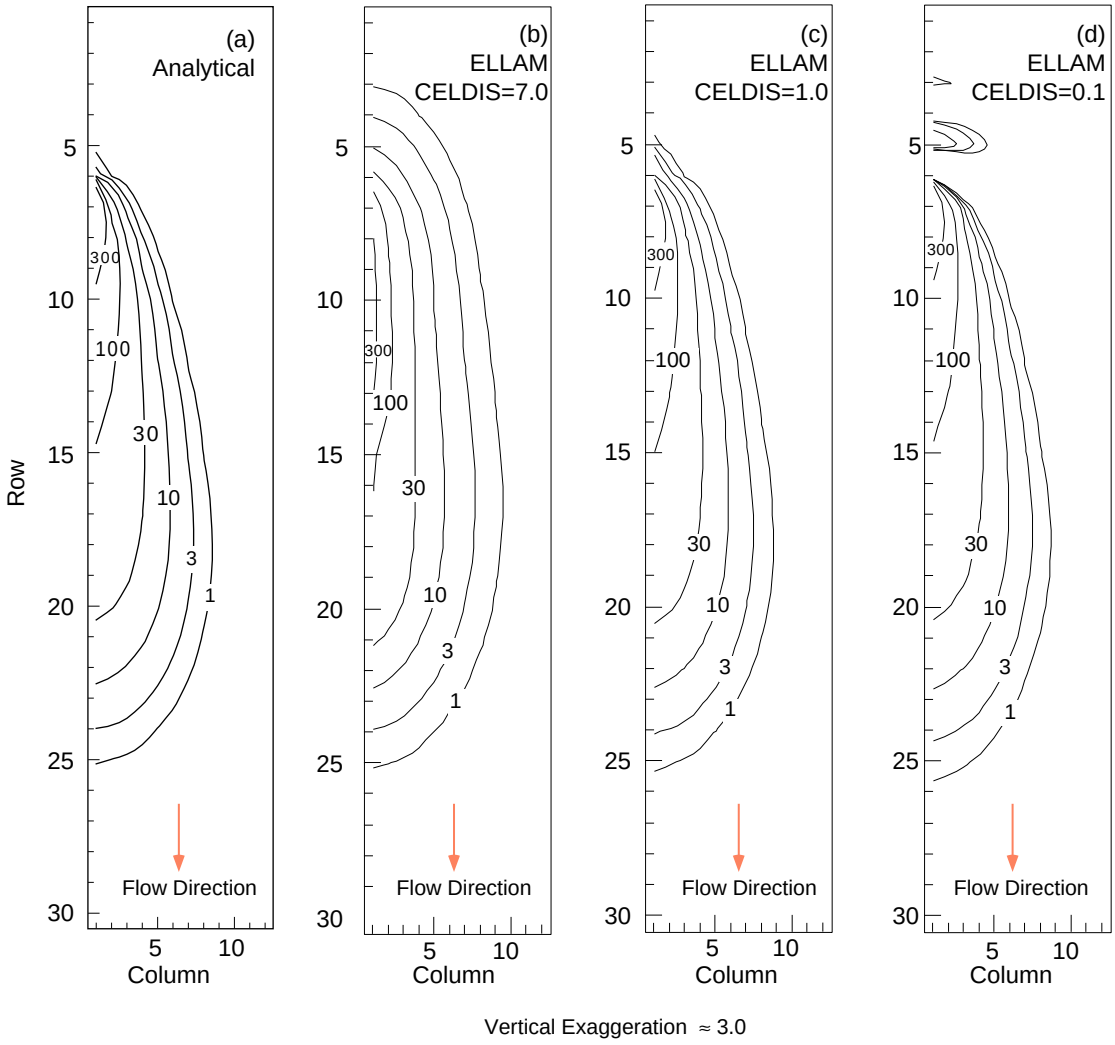

Figure 7. Concentration contours for (a) analytical and (b-d) *ELLAM* numerical solutions in the horizontal plane containing the solute source (layer 1) for three-dimensional solute transport in a uniform steady flow field at t = 400 days. Parameters are defined in table 1.

= *NSL* = 4, and *NT* = 16 (figure 7b); *CELDIS* = 1 (14 time increments), *NSC* = *NSR* = *NSL* = 4, and *NT* = 4 (figure 7c); and *CELDIS* = 0.1 (134 time increments), *NSC* = *NSL* = 4, *NSR* = 8, and *NT* = 16 (figure 7d).

As noted in previous *MOC3D* reports, a slightly greater spreading is evident in the numerical model results than in the analytical solution, both upstream as well as downstream of the source. Part of this difference, however, is explained by the fact that the numerical source is applied over a finite area in the horizontal plane of the model, in which the length of the source

cell is 3 m in the direction parallel to flow, whereas the source is represented as a true point in the analytical solution.

The *ELLAM* results using two transport time increments (figure 7b) indicate that more time steps are needed in order to accurately simulate dispersion. The *ELLAM* results for 14 time steps (figure 7c) accurately characterize the dispersive flux without the spreading upstream from the source that is produced by *MOC3D*. The *ELLAM* results for 134 time steps (figure 7d) yield even less spreading upstream of the source, but do exhibit numerical oscillations produced because the concentration gradient is too steep relative to the grid spacing.

Konikow et al. (1996) also present comparisons for this case for vertical planes parallel and perpendicular to the flow direction. These same comparisons between the analytical and *ELLAM* results are as close as between figures 7a and 7c, and are not reproduced here.

Two-Dimensional Radial Flow

A radial-flow transport problem was used to compare the *ELLAM* solution to the analytical solution given by Hsieh (1986) for a finite-radius injection well in an infinite aquifer. The problem involves flow from a single injection well; the velocities vary in space and vary inversely with the distance from the injection well.

The parameters for the problem are summarized in table 2 and the analytical solution and other details about this test case are presented by Konikow et al. (1996, p. 49-50) and Kipp et al. (1998, p. 21-22). The problem was modeled using a grid having 30 cells in the *x*-direction and 30 cells in the *y*-direction, representing one quadrant of the radial flow field (90 of 360 degrees).

In figure 8, *ELLAM* solutions using 2, 29, and 563 transport time increments (corresponding to *CELDIS* values of 75, 5, and 0.25, respectively) are shown along with the analytical solution. Each run represents the qualitative features of the analytical solution (figure 8a), with the exception of the high concentration contour of the two-time-step (*CELDIS* = 75) run, which manifests an articulated rather than smooth shape. The high concentration contour is often the most difficult for *ELLAM* to portray accurately.

Table 2. Parameters used in *ELLAM* simulation of two-dimensional, steady-state, radial flow case, showing range of values tested for selected numerical parameters

Parameter	Value
$T_{xx} = T_{yy}$	3.6 m^2/hour
ε	0.2
α_L	10.0 m
$\alpha_{TH} = \alpha_{TV}$	10.0 m
PERLEN (length of stress period)	1000 hours
Q (at well)	56.25 m^3/hour
Source concentration (C')	1.0
Number of rows	30
Number of columns	30
Number of layers	1
DELR (Δx) = *DELC* (Δy)	10.0 m
Thickness (*b*)	10.0 m
CELDIS	0.25-75.0
NSCEXP	2-3
NSREXP	2-3
NSLEXP	1
NTEXP	2-4

The two-time-step run shown in figure 8b uses *CELDIS* = 75, *NSC* = *NSR* = 4, *NSL* = 2, *NT* = 16. The high concentration contour has nonphysical oscillations. Increasing *NS* and *NT* values (not shown) smooth that one contour noticeably, but inadequately.

The 29-time-step run shown in figure 8c uses *CELDIS* = 5, *NSC* = *NSR* = 4, *NSL* = 2, *NT* = 4. It is a very close approximation to the analytical solution, although the high concentration contour does not have a constant radius.

Figures 8d and 8e show the results of two runs using 563 time steps. Parameter values for these two cases are *CELDIS* = 0.25, *NSC* = *NSR* = 4, *NSL* = 2, *NT* = 4, and *CELDIS* = 0.25, *NSC* = *NSR* = 8, *NSL* = 2, *NT* = 4. The improvement obtained using higher *NS* values when

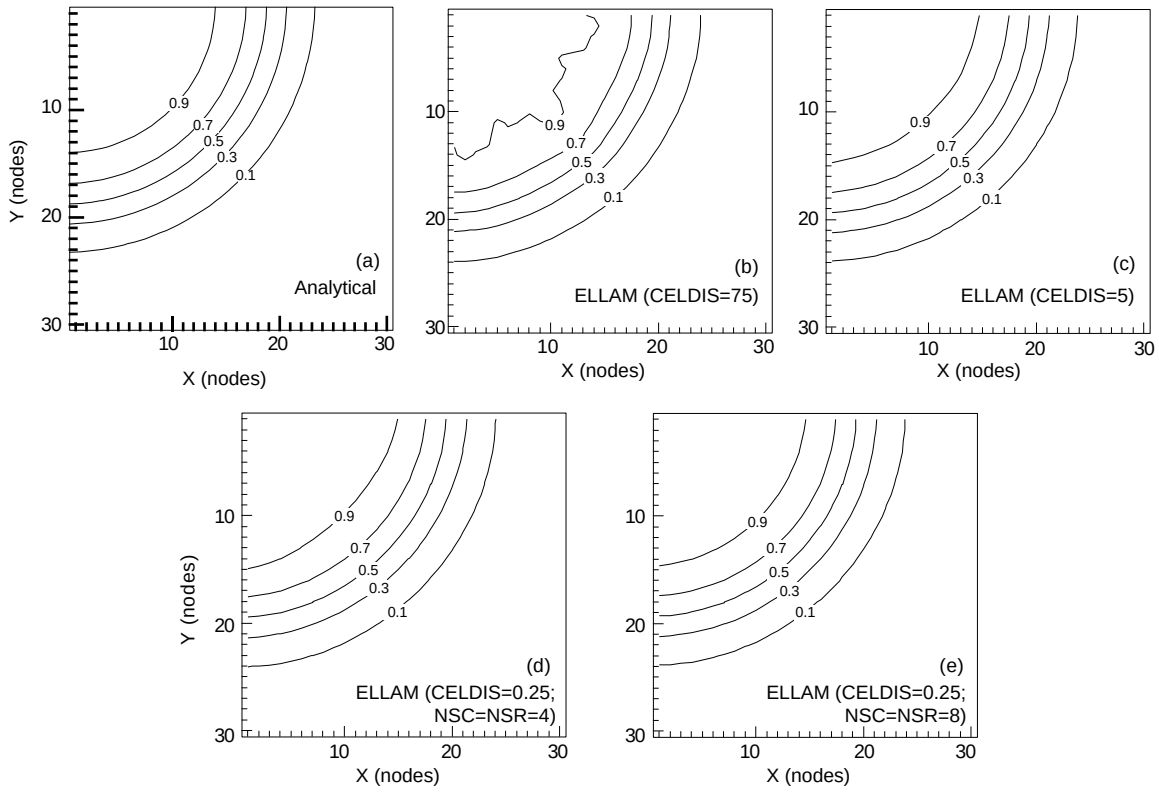

Figure 8. Contours of relative concentrations calculated using (a) analytical and (b-e) numerical *ELLAM* models for solute transport in a steady radial flow field. Source concentration is 1.0 and source is located in cell (1,1). Grid spacing is 10.0 m. Parameters are defined in table 2.

modeling with numerous time steps is illustrated here. The former values yield a set of contours not at constant radii, whereas the latter, using $NSC = NSR = 8$, produce a very close match to the analytical solution.

For comparison, the implicit solutions of *MOC3D* match the analytical solution almost exactly and agree very closely with the explicit dispersive transport solution of *MOC3D* (see Konikow et al., 1996, figure 29). The explicit solution required 596 time increments and used 750 s of cpu time, and the implicit solutions required 282 time increments and used 445 s of cpu time. The *ELLAM* solutions represented in figures 8c-e required 138 s, 1,990 s, and 5,780 s, respectively, of cpu time (all times represent runs on a Unix workstation).

Point Initial Condition in Uniform Flow

A problem involving three-dimensional solute transport from an instantaneous point source, or Dirac initial condition, in a uniform flow field was used as another test problem for *MOC3D*. An analytical solution for an instantaneous point source in a homogeneous infinite aquifer is given by Wexler (1992, p. 42), who presents the *POINT3* code for a related case of a continuous point source. The *POINT3* code was modified to solve for the desired case of an instantaneous point source.

Test problems were designed to evaluate the numerical solution for two cases—one in which flow is parallel to the grid (in the *x*-direction) and one in which flow occurs at 45 degrees to the *x*- and *y*-axes. This allows us to evaluate the accuracy and sensitivity of the numerical solution to the orientation of the grid relative to the flow. The assumptions and parameters for this test case are summarized in table 3 and are described in more detail by Konikow et al. (1996).

We specified boundary conditions for the test case of flow in the *x*-direction such that V_x = 1.0275 m/d, and $V_y = V_z$ = 0.0 m/d. For flow at 45 degrees to *x* and *y*, we specified boundary conditions such that $V_x = V_y$ = 1.0275 m/d, and V_z = 0.0 m/d. For both cases, the distance the center of mass of the plume travels in the *x*-direction is the same for equal simulation times. Note, however, that the magnitude of velocity is higher in the latter case; therefore, there will be more dispersion in that problem during an equal time interval.

The results for both the analytical and numerical solutions for the case in which flow occurs in the *x*-direction are shown in figure 9, where values of *CELDIS* = 5 (yielding six transport time increments), *NSC* = *NSR* = *NSL* = 4, and *NT* = 2 were used. These results represent the concentrations in the plane of the initial source of solute. The *ELLAM* transport algorithm gives results (figure 9b) for a 72 by 72 grid that are close to those of the analytical solution (figure 9a). The numerical results, however, do show some slight spreading (or numerical dispersion) relative to the analytical solution in both the transverse and longitudinal

Table 3. Parameters used in *ELLAM* simulation of three -dimensional transport from a point source with flow in the *x*-direction and flow at 45 degrees to *x*- and *y*-axes

Parameter	Value
$T_{xx} = T_{yy}$	10.0 m^2/day
ε	0.1
α_L	1.0 m
$\alpha_{TH} = \alpha_{TV}$	0.1 m
PERLEN (length of stress period)	90 days
V_x	1.0275 m/day
$V_y = V_z$	0.0 m/day*
Initial concentration at source	1×10^6
Source location in transport grid	*Column* = 11, *Row* = 36, *Layer* = 4
Number of rows	72
Number of columns	72
Number of layers	24
DELR (Δx)	3.33 m
DELC (Δy)	3.33 m
Layer thickness ($b=\Delta z$)	10.0 m
CELDIS	5.0
NSCEXP	2
NSREXP	2
NSLEXP	2
NTEXP	1

* For flow at 45 degrees to *x*- and *y*-axes, V_y = 1.0275 m/day

directions. Increasing the number of time increments does not completely eliminate the spreading and causes some loss of peak concentrations, even with increased *NS* values. In contrast to the previous *MOC3D* solutions, *ELLAM* results retain the symmetry of the analytical solution. Part of the discrepancy is attributable to the need in *ELLAM* to use four grid points to discretize a front. This precludes the possibility of modeling with high accuracy the migration of an instantaneous point source placed in a single grid cell. Therefore, we modified this test problem for *ELLAM* by using a dispersed solute mass as an initial condition. The initial condition for this *ELLAM* test is the analytical solution to the original point source problem at *t* =

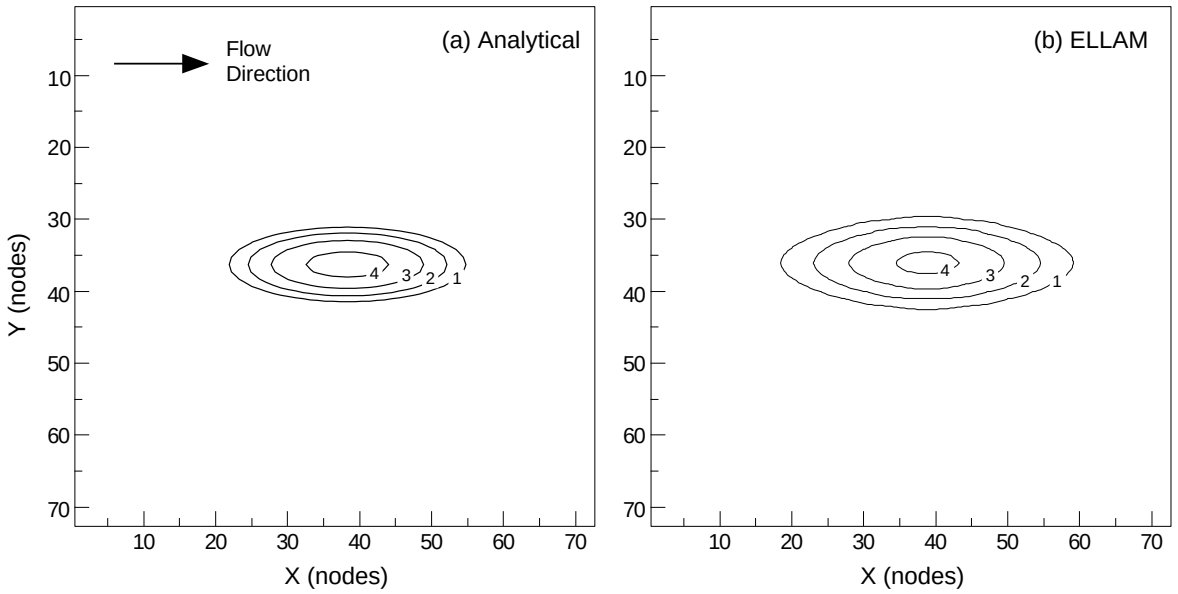

Figure 9. Concentration contours for (a) analytical and (b) numerical solutions for transport of a point initial condition in uniform flow in the *x*-direction at *t* = 90 days. The *z*-component of flow is zero, but there is dispersion in all three directions. Contour values are the log of the concentrations.

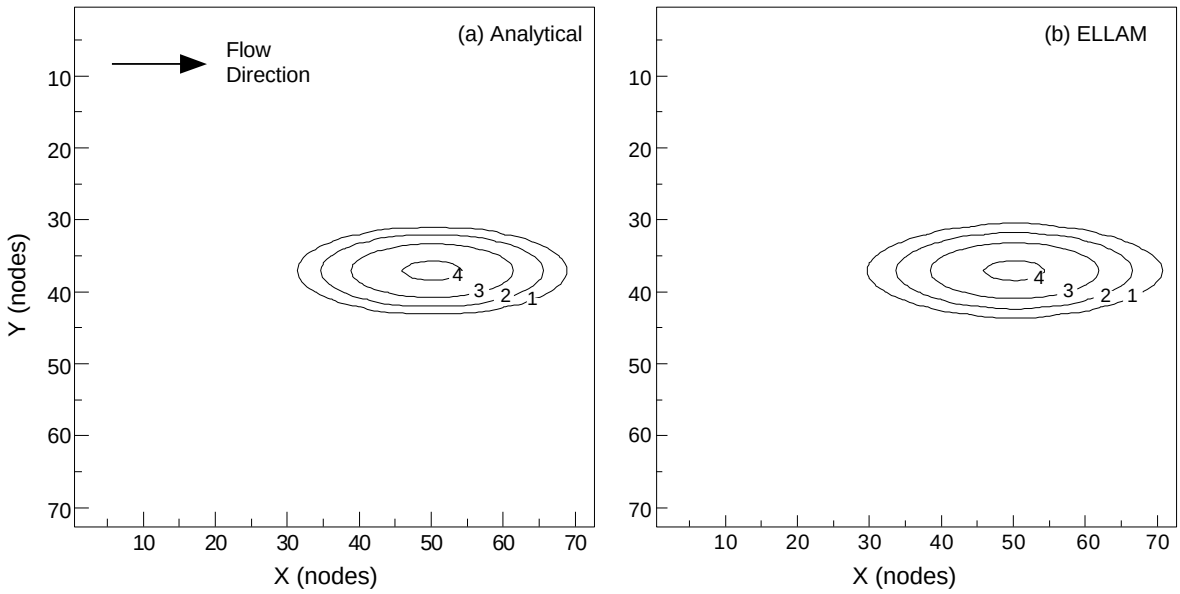

Figure 10. Concentration contours for (a) analytical and (b) *ELLAM* numerical solutions for transport of a dispersed-point initial condition in uniform flow in the *x*-direction at *t* = 130 days. The *y*- and *z*-components of flow are zero, but dispersion occurs in all three directions. Contour values are the log of the concentrations.

90 days, and the *ELLAM* solution is evaluated against the Wexler analytical solution later in time at *t* = 130 days. These results are presented in figure 10, where it can be seen that the analytical

solution (figure 10a) and the numerical solution (figure 10b) are very similar. The *ELLAM* solution, however, still clearly exhibits some numerical dispersion, which is most evident at the lower concentrations.

The results of the test problem for flow at 45 degrees to the grid are shown in figure 11, again using a 72×72×24 grid. The analytical solution for t = 130 days, which provides the basis for the evaluation, is shown in figure 11a. As was done for the previous analysis shown in figure 10, the *ELLAM* solution in this case also used the analytical solution at t = 90 days as the initial

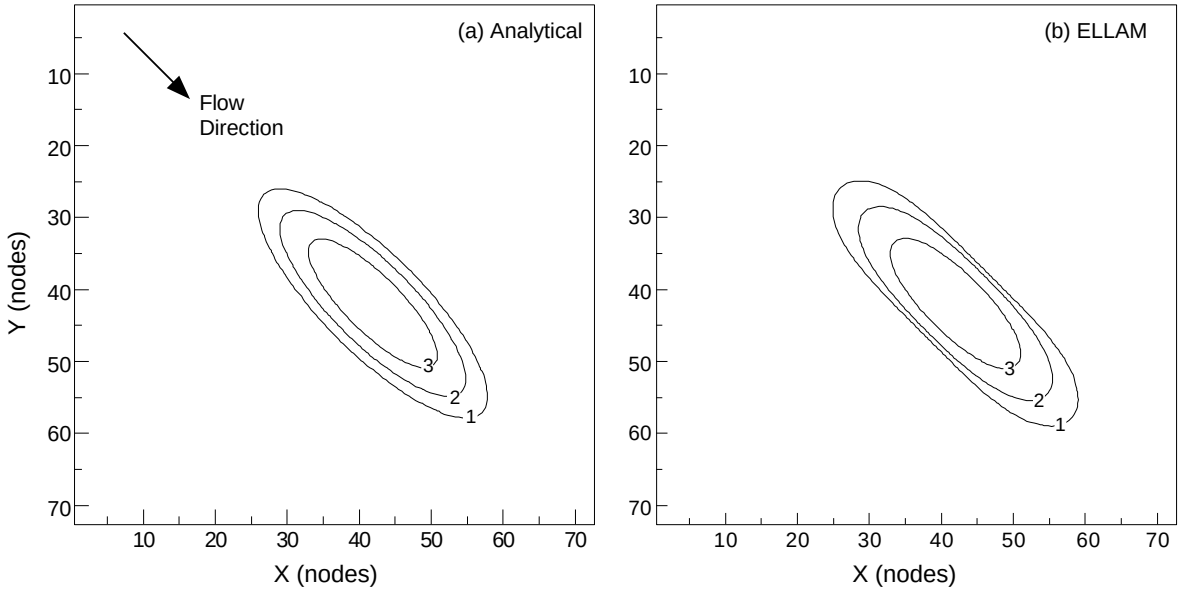

Figure 11. Concentration contours for (a) analytical and (b) *ELLAM* numerical solutions for transport of a point initial condition in uniform flow at 45 degrees to the *x*-direction at t = 130 days. Contour values are the log of the concentrations.

conditions. The results using *CELDIS* = 5 (three time increments), *NSC* = *NSR* = *NSL* = 4, and *NT* = 2 are shown in figure 11b for the plane of the initial source. As in the previous case (where flow is aligned with the grid), *ELLAM* produces the symmetry characteristic of the analytical solution. There is also slight longitudinal spreading (numerical dispersion) that is not alleviated by increasing the number of time steps.

Unlike the previous case, the numerical results in figure 11b do show some distortion of the shape of the plume relative to the analytical solution. It is not as pronounced, however, as the "hourglass" shape yielded by *MOC3D* for the Dirac problem (see Kipp et al., 1998, figure 14). There is a narrowing of the plume calculated with the numerical model, which is characteristic of a grid-orientation effect and is caused primarily by the off-diagonal (cross-derivative) terms of the dispersion tensor. When flow is oriented parallel to the grid, or when longitudinal and transverse dispersivities are equal, the cross-derivative terms of the dispersion tensor are zero. Because flow is at 45 degrees to the grid in this test problem, the cross-derivative dispersive flux terms are of maximum size and negative concentrations are most likely to occur. The calculated concentration field is less accurate in this case largely because the standard differencing scheme for the cross-derivative dispersive flux terms can cause overshoot and undershoot of concentrations. If the base (or background) is zero concentration, then undershoot will cause negative concentrations. The magnitude of this overshoot and undershoot effect can be reduced by using a finer grid.

Some small areas of negative concentrations were calculated, but they do not appear in figure 11b using logarithmic-scale contouring. To show the extent of the areas of negative concentration, we have replotted the results illustrated in figure 11b in figure 12, using two types of shading for areas where the relative concentration is less than -0.05 and less than -10.0. We tested the sensitivity of the extent of negative concentrations to the size of the transport time increment by reducing the value of *CELDIS* to 0.25. The area over which negative concentrations occurred was only slightly smaller. The increase in execution time, however, was significant, so the very small improvement does not appear to justify the extra computational costs.

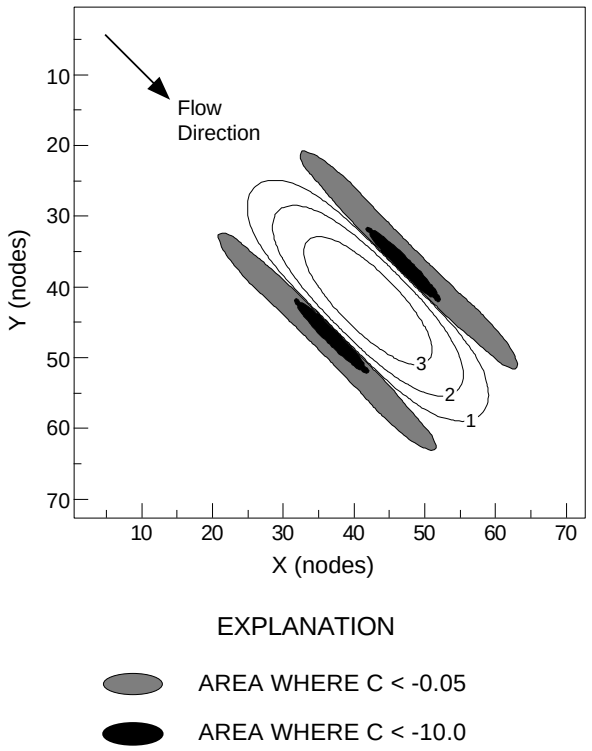

Figure 12. Concentration contours for *ELLAM* numerical solution showing areas of calculated negative concentrations for problem represented in figure 11b.

Constant Source in Nonuniform Flow

Burnett and Frind (1987) used a numerical model to simulate a hypothetical problem having a constant source of solute over a finite area at the surface of an aquifer having homogeneous properties, but nonuniform boundary conditions, which result in nonuniform flow. Because an analytical solution is not available for such a complex system, we use their results for this test case as a benchmark for comparison with the results of applying the *ELLAM* algorithm in *MOC3D*, as was also done by Konikow et al. (1996) and Kipp et al. (1998). Burnett and Frind (1987) used an alternating-direction Galerkin finite-element technique to solve the flow and solute-transport equations in both two and three dimensions. Their model also includes the capability to vary α_T as a function of coordinate direction, thereby allowing this feature of

MOC3D to be evaluated. A detailed description of the problem geometry and of the parameters for the numerical simulation are presented by Konikow et al. (1996, p. 55-60).

Cases of both two- and three-dimensional transport were examined for this problem. The grids used in the *ELLAM* simulations were designed to match as closely as possible the finite-element mesh used by Burnett and Frind (1987). Some differences in discretization, however, could not be avoided because the finite-element method uses a point-centered grid whereas *ELLAM* uses a block-centered (or cell-centered) grid. The former allows specifications of values at nodes, which can be placed directly on boundaries of the model domain. Nodes in *ELLAM* are located at the centers of cells, and block-centered nodes are always one-half of the grid spacing away from the edge of the model domain. Among the small differences arising from the alternative discretization schemes are that, in the *ELLAM* grid, (1) the modeled location of the 14.25 m long source area is offset by 0.225 m towards the right, and (2) the total length of the domain is 199.5 m.

The first simulation of this test problem was for the case of a two-dimensional model. The input data values for this analysis are listed in table 4. The top discretization layer consisted of constant-head nodes and the solute source.

Results for the two-dimensional case from the *ELLAM* simulation closely match those of Burnett and Frind (1987) (see figure 13). The results using *CELDIS* = 30 (seven time increments), *NSC* = *NSR* = *NSL* = 4, and *NT* = 32 are shown. The shape of the plume is almost exactly the same for both models. In the *ELLAM* results, however, the highest concentration contour (0.9) does not extend as far downgradient as that of Burnett and Frind (1987), while the low concentration contour (0.3) from *ELLAM* extends slightly farther downgradient. Overall, the *ELLAM* results provide a closer match to the contours of Burnett and Frind than do the *MOC3D* contours using 381, 1901, or 4218 time increments (see Kipp et al., 1998). The *ELLAM* contours (for all *NS* values tested) are free of "wiggles" in the *MOC3D* solution discussed by Kipp et al.

Table 4. Parameters used for *ELLAM* simulation of transport in a vertical plane from a continuous point source in a nonuniform, steady-state, two-dimensional flow system (described by Burnett and Frind, 1987)

Parameter	Value
K	1.0 m/day
ε	0.35
α_L	3.0 m
α_{TH}	0.10 m
α_{TV}	0.01 m
PERLEN (length of stress period)	12,000 days
Source concentration (C')	1.0
Number of rows	1
Number of columns[1]	141
Number of layers[1]	91
DELR (Δx)	1.425 m
DELC (Δy)	1.0 m
Layer thickness ($b=\Delta z$)	0.2222-0.2333 m
CELDIS	30.0
NSCEXP	2
NSREXP	2
NSLEXP	2
NTEXP	5

[1] One row and layer were allocated to defining boundary conditions, so concentrations calculated in only 140 columns and 90 layers were used for comparison.

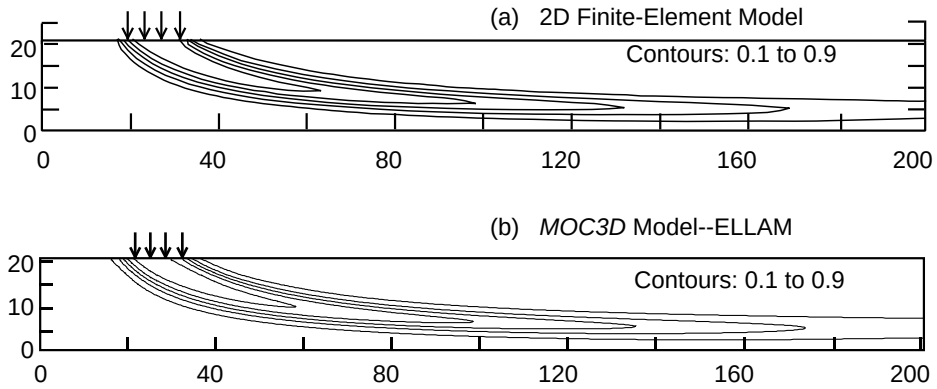

Figure 13. Two-dimensional simulation results for nonuniform-flow test case showing plume positions as contours of relative concentration; (a) finite-element model (modified from Burnett and Frind, 1987, figure 8a), and (b) *ELLAM* solution using *CELDIS* = 30. Contour interval is 0.2 relative concentration.

(1998). Increasing the number of transport time increments produced a solution having a slightly greater downgradient extent, but still short of *MOC3D* results.

As was done for the *MOC3D* tests (Konikow et al., 1996; Kipp et al., 1998), the *ELLAM* grid was expanded laterally to 15 rows having Δy of 1.0 m for the three-dimensional version of this case. Figure 14 shows the transport results in a vertical plane at the middle of the plume for both models for the case in which α_{T_V} = 0.01 m and α_{T_H} = 0.1 m. The *ELLAM* results for the vertical plane in the first row are contoured in figure 14b (because of symmetry, we only simulate half of the plume, as explained by Konikow et al., 1996). The *ELLAM* plume closely matches that calculated by the finite-element model (figure 14a), although the former shows slightly farther downstream migration of low concentrations of solute. As in the two-dimensional case, the *ELLAM* solution provides a closer match to the Burnett and Frind (1987) solution than do the previous *MOC3D* results.

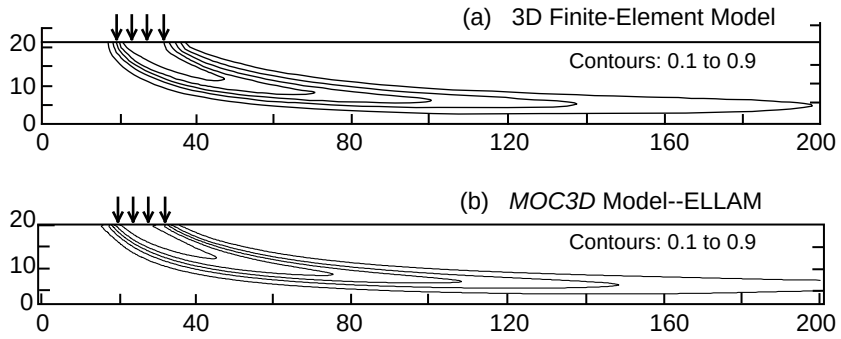

Figure 14. Three-dimensional simulation results for nonuniform-flow test case in which α_{TH} = 0.1 m and α_{TV} = 0.01 m: (a) finite-element model (modified from Burnett and Frind, 1987, figure 8c), and (b) numerical *ELLAM* solution using *CELDIS* = 30. Plume positions are represented by contours of relative concentration; contour interval is 0.2 relative concentration.

Figure 15 shows the results for the case in which the vertical transverse dispersivity is increased by a factor of ten, so that $\alpha_{T_H} = \alpha_{T_V}$ = 0.1 m. The *ELLAM* results for *CELDIS* = 30 yielded concentrations that were noticeably low near the source (near the upgradient end of the plume), so the simulation was repeated using *CELDIS* = 21 (10 time increments). These *ELLAM*

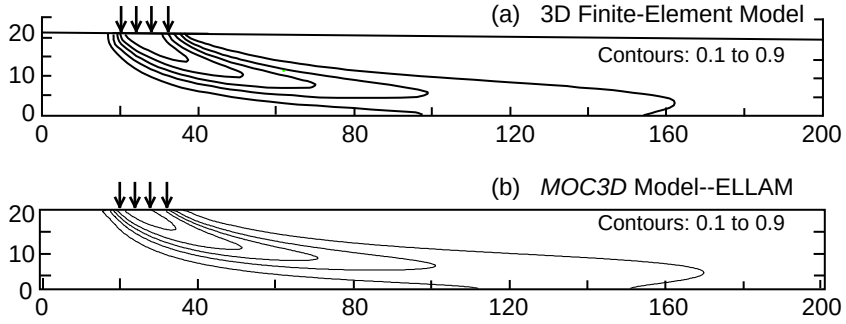

Figure 15. Three-dimensional simulation results for nonuniform-flow test case in which $\alpha_{TH} = \alpha_{TV} = 0.1$ m: (a) finite-element model (modified from Burnett and Frind, 1987, figure 9b), and (b) numerical *ELLAM* solution using *CELDIS* = 21. Plume positions are represented by contours of relative concentration; contour interval is 0.2 relative concentration.

results are illustrated in figure 15b and appear to agree very closely with the results of Burnett and Frind (1987) (figure 15a).

Relative Computational and Storage Efficiency

Computer-memory requirements for *ELLAM* are greater than those for the explicit or implicit *MOC3D* dispersive transport algorithm. The additional arrays required can increase the memory size requirement by as much as a factor of three (see table 5).

The computational effort required by the *ELLAM* simulator is strongly dependent on the size of the problem being solved, as determined by the total number of nodes, the *NS* and *NT* values, and the total number of time increments (controlled by *CELDIS*). The user is cautioned that using values for *NS* and *NT* parameters that are too small for a given problem may lead to inaccurate solutions. Sensitivity testing will help the user determine appropriate values to specify. Analyses indicate that the greatest computational effort, as measured by *CPU* time, is typically expended in the mass tracking routines. For a given problem, computational time may vary significantly as a function of the characteristics of the particular computer on which the

Table 5. Execution times and storage requirements for *MOC3D* and *ELLAM* for selected test cases

	Run Time in CPU-seconds			Array Elements Used[1]		
Problem Description	Explicit	Implicit	ELLAM	Explicit	Implicit	ELLAM
One-Dimensional Steady Flow[2]	7	10	9 *CELDIS*=10.1	11,457	17,400	33,489
Three-Dimensional Steady Flow[2]	404	175	1,366 *CELDIS*=1	897,331	1,602,994	3,344,624
Two-Dimensional Radial Flow and Dispersion[2]	930	445	138 *CELDIS*=5	455,737	499,900	233,564
Point Initial Condition in Uniform Flow[2]	210	310	2,721 *CELDIS*=5	1,728,673	2,406,112	3,384,524
Constant Source in Nonuniform Flow (Two-Dimensional)[3]	13,360	2,450	2,245 *CELDIS*=30	868,951	1,457,602	2,850,056
Constant Source in Nonuniform Flow (Three-Dimensional)[3]	38,117	12,026	4,400 *CELDIS*=30	12,823,151	21,652,034	41,206,836

[1] Data arrays and lists for *MODFLOW* and explicit *MOC3D* are allocated space in one array, the *MODFLOW* "X" array. *ELLAM* also uses an "MX" array for integer arrays.

[2] Data General server with a Motorola 88110 chip running DG Unix 5.4R3.10 with 256MB RAM and a 45 MHz processor was used for this problem. Green Hills Software FORTRAN-88000 was used to compile *MOC3D*.

[3] Silicon Graphics server with an R8000 chip running Irix 6.0.1 with 576MB RAM and a 90 MHz processor was used for this problem. MIPSpro F77 was used to compile *MOC3D*.

simulation is performed, and on which FORTRAN compiler and options were used to generate the executable code.

For a given problem, the *ELLAM* algorithm can often yield an accurate solution more efficiently than the previously documented explicit or implicit *MOC* options. However, this will typically require the use of a *CELDIS* value of 5 or more; the explicit and implicit versions of *MOC* require that *CELDIS* be less than or equal to 1.0. Table 5 shows that *ELLAM* was more efficient for three of the six test problems evaluated.

CONCLUSIONS

The *ELLAM* advective-dispersive transport algorithm presented as an alternative solution method within the *MOC3D* simulator can model the transient, three-dimensional, transport of a

solute subject to decay and retardation. The numerical methods used to solve the governing equations have broad general capability and flexibility for application to a wide range of hydrogeological problems.

The accuracy and precision of the numerical results of the implicit *ELLAM* simulator were evaluated by comparison to analytical and numerical solutions for the same set of test problems as reported for *MOC3D* (Versions 1 and 2), with the instantaneous point source problem modified slightly. These evaluation tests indicate that the solution algorithms in the *ELLAM* model can successfully and accurately simulate three-dimensional transport and dispersion of a solute in flowing ground water. To avoid non-physical oscillations and loss of peak concentrations, care must be taken to use a grid having sufficient mesh density to adequately resolve sharp fronts. The primary advantages of the *ELLAM* code are that fewer transport time steps need be used and that mass is conserved globally. Using *ELLAM* with few time steps can provide an accurate and cost-effective way of discerning salient features of a solute-transport process under a complex set of boundary conditions. Furthermore, the *ELLAM* algorithm eliminates the previous restriction in *MOC3D* that the transport grid be uniformly spaced. The computational effort required by the *ELLAM* simulator is strongly dependent on the size of the problem being solved, as determined by the total number of nodes, the *NS* and *NT* values, and the total number of time increments (controlled by *CELDIS*, a model parameter that is analogous to the Courant number). For test cases in which *ELLAM* was more efficient than the explicit or implicit *MOC* options, use of *CELDIS* values equal to or greater than 5.0 were required.

Acknowledgments. The authors appreciate the helpful model evaluation and review comments provided by USGS colleagues R.W. Healy and K.L. Kipp.

REFERENCES

Burnett, R.D., and Frind, E.O., 1987, Simulation of contaminant transport in three dimensions, 2. Dimensionality effects: Water Resources Research, v. 23, no. 4, p. 695-705.

Celia, M.A., Russell, T.F., Herrera, I., and Ewing, R.E., 1990, An Eulerian-Lagrangian localized adjoint method for the advection-diffusion equation: Advances in Water Resources, v. 13, no. 4, p. 187-206.

Garder, A.O., Peaceman, D.W., and Pozzi, A.L., 1964, Numerical calculation of multidimensional miscible displacement by the method of characteristics: Soc. Petroleum Eng. Jour., v. 4, no.1, p. 26-36.

Harbaugh, A.W., and McDonald, M.G., 1996a, User's documentation for MODFLOW-96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 96-485, 56 p.

Harbaugh, A.W., and McDonald, M.G., 1996b, Programmer's documentation for MODFLOW-96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 96-486, 220 p.

Healy, R.W., and Russell, T.F., 1993, A finite-volume Eulerian-Lagrangian localized adjoint method for solution of the advection-dispersion equation: Water Resources Research, v. 29, no.7, p. 2399-2413.

Heberton, C.I., Russell, T.F., Konikow, L.F., and Hornberger, G.Z., 2000, A three-dimensional finite-volume Eulerian-Lagrangian localized adjoint method (ELLAM) for solute-transport modeling: U.S. Geological Survey Water-Resources Investigations Report 00-4087, 63 p.

Hsieh, P. A., 1986, A new formula for the analytical solution of the radial dispersion problem: Water Resources Research, v. 22, no. 11, p. 1597-1605.

Kipp, K.L., Konikow, L.F., and Hornberger, G.Z., 1998, An implicit dispersive transport algorithm for the U.S. Geological Survey MOC3D solute-transport model: U.S. Geological Survey Water-Resources Investigations Report 98-4234, 54 p.

Konikow, L.F., and Bredehoeft, J.D., 1978, Computer model of two-dimensional solute transport and dispersion in ground water: U.S. Geological Survey Techniques of Water-Resources Investigations, Book 7, Chapter C2, 90 p.

Konikow, L.F., Goode, D.J., and Hornberger, G.Z., 1996, A three-dimensional method-of-characteristics solute-transport model (MOC3D): U.S. Geological Survey Water-Resources Investigations Report 96-4267, 87 p.

McDonald, M.G., and Harbaugh, A.W., 1988, A modular three-dimensional finite-difference ground-water flow model: U.S. Geological Survey Techniques of Water-Resources Investigations, Book 6, Chapter A1, 586 p.

Wexler, E.J., 1992, Analytical solutions for one-, two-, and three-dimensional solute transport in ground-water systems with uniform flow: U.S. Geological Survey Techniques of Water-Resources Investigations, Book 3, Chapter B7, 190 p.

FIGURES

1-15. Diagrams showing:

1. Schematic representation (for a simple case having constant velocity) showing how mass is advected into cell Ω_l at time level t^{n+1} from the inflow boundary, from a fluid source in cell 1, and from storage at time level t^n in cells 1 and 2
2. Schematic representation showing how mass is advected to an outflow boundary at time level t^{n+1} from the inflow boundary, from a fluid source in cell 3, and from storage at time level t^n in all five cells
3. Two-dimensional example illustrating that pre-image of a cell may be irregularly shaped and not easily defined by backtracking from t^{n+1} to t^n
4. Solute concentration versus distance to illustrate one-dimensional advection of known mass distribution from old time level to new time level (Courant number = 1)
5. Examples of spatial distribution of approximate test functions (w_l) for selected one-dimensional cases
6. Schematic representation of tracking mass from an inflow boundary
7. Concentration contours for (a) analytical and (b-d) *ELLAM* numerical solutions in the horizontal plane containing the solute source (layer 1) for three-dimensional solute transport in a uniform steady flow field at t = 400 days
8. Contours of relative concentrations calculated using (a) analytical and (b-e) numerical *ELLAM* models for solute transport in a steady radial flow field
9. Concentration contours for (a) analytical and (b) numerical solutions for transport of a point initial condition in uniform flow in the x-direction at t = 90 days
10. Concentration contours for (a) analytical and (b) *ELLAM* numerical solutions for transport of a dispersed-point initial condition in uniform flow in the x-direction at t = 130 days
11. Concentration contours for (a) analytical and (b) *ELLAM* numerical solutions for transport of a dispersed-point initial condition in uniform flow at 45 degrees to the x-direction at t = 130 days
12. Concentration contours for *ELLAM* numerical solution showing areas of calculated negative concentrations for problem represented in figure 11b
13. Two-dimensional simulation results for nonuniform-flow test case showing plume positions as contours of relative concentration; (a) finite-element model (modified from Burnett and Frind, 1987, figure 8a), and (b) *ELLAM* solution using *CELDIS* = 30
14. Three-dimensional simulation results for nonuniform-flow test case in which α_{TH} = 0.1 m and α_{TV} = 0.01 m: (a) finite-element model (modified from Burnett and Frind, 1987, figure 8c), and (b) numerical *ELLAM* solution using *CELDIS* = 30
15. Three-dimensional simulation results for nonuniform-flow test case in which α_{TH} = α_{TV} = 0.1 m: (a) finite-element model (modified from Burnett and Frind, 1987, figure 9b), and (b) numerical *ELLAM* solution using *CELDIS* = 21

TABLES

1. Base-case parameters used in *ELLAM* simulation of transport from a continuous point source in a three-dimensional, uniform, steady-state flow system
2. Parameters used in *ELLAM* simulation of two-dimensional, steady-state, radial flow case, showing range of values tested for selected numerical parameters
3. Parameters used in *ELLAM* simulation of three-dimensional transport from a point source with flow in the *x*-direction and flow at 45 degrees to *x*- and *y*-axes
4. Parameters used for *ELLAM* simulation of transport in a vertical plane from a continuous point source in a nonuniform, steady-state, two-dimensional flow system (described by Burnett and Frind, 1987)
5. Execution times and storage requirements for *MOC3D* and *ELLAM* for selected test cases