Solution of the Advective-Dispersive Transport Equation Using a Least Squares Collocation, Eulerian-Lagrangian Method

L. R. Bentley and G. F. Pinder
Department of Civil Engineering and Operations Research, Princeton University, Princeton, New Jersey 08540

I. Herrera
Instituto de Geofisica, Universidad Nacional Autonoma de Mexico, Mexico, D. F.

Numerical solution of the advective-dispersive transport equation is difficult when advection dominates. Difficulties arise because of the first-order spatial derivatives which can be eliminated by a local coordinate transformation to the characteristic lines of the first order hyperbolic portion of the equation. The resulting differential equation is discretized using a finite difference in time and finite elements in space employing cubic Hermite basis functions. The residuals at individual collocation points are then computed. The sum of the squares of the residuals is minimized to form the necessary set of algebraic equations. The method has performed well in one-dimensional test problems.

INTRODUCTION

The difficulties arising in the application of numerical approximations to advective-dispersive transport problems are well known and can be attributed to the dual nature of the equation. When transport is advection dominated, the equation behaves as a first-order hyperbolic equation. When transport is dispersion dominated, it behaves as a second-order parabolic equation. Since many practical problems, such as estuary studies, have transport conditions that vary greatly in time and space, a numerical method should be able to function well in transport conditions that vary from advection to dispersion-dominated.

Central difference methods and symmetrically weighted finite-element schemes are well suited to the purely parabolic problem. However, when applied to the first order-term associated with advection, numerical oscillations develop when advection dominates. Much research has been devoted to eliminating the oscillations without introducing artificial numerical diffusion.

In a Lagrangian coordinate system, the first-order term associated with advection does not exist. Several workers (Varoglu [22], Varoglu and Finn [23, 24], O'Neill and Lynch [19]) have achieved good results using Lagrangian schemes. However, the disadvantages of using a deforming grid system such as currently used in Lagrangian methods, seem to preclude the general use of the method.

Numerical Methods for Partial Differential Equations, 5, 227–240 (1989)
© 1989 John Wiley & Sons, Inc. CCC 0749-159X/89/03227-14$04.00

Eulerian-Lagrangian methods were developed to take advantage of the attractive properties of the Lagrangian coordinate system while avoiding the disadvantages of a deforming coordinate system. The methods are implemented in two steps. The first step solves the first-order hyperbolic problem by the method of characteristics. This creates an intermediate solution that accomodates advection. A second step uses the intermediate solution to solve a parabolic problem associated with diffusion. Several approaches have been reported [1, 2, 4, 6, 7, 9, 10, 14, 17, 18, 21]. Encouraging results have been obtained, suggesting that further investigation of this general strategy is warranted.

Least squares collocation (LESCO) can be considered a finite element method. The domain of interest is divided into elements over which a trial function approximates the scalar field. A residual statement of the differential equation is formulated. The residual is then evaluated at a discrete set of locations called collocation points. The coefficients associated with the basis functions are computed by minimizing the sum of the squares of the collocation point residuals.

Several workers have investigated the use of least squares collocation to solve partial differential equations [3, 5, 8, 11, 13, 15, 20], but the method has not become generally popular. Although conceptually simple, the method seems to require more computational effort than other standard methods. As an example, simple test problems that compare the solutions of one-dimensional parabolic PDEs indicate that equivalent accuracy is obtained by LESCO using cubic Hermite basis functions and Galerkin finite elements using a Lagrangian quadratic basis. The resulting matrices for the two methods have equal rank, but the LESCO matrix bandwidth is seven compared to five for the Galerkin finite-element method. In addition, LESCO requires higher order continuity basis functions and more work in the matrix assembly.

However, LESCO is particularly well suited for solving problems that require coordinate transformations or tracking. Being a collocation method, it is possible to sequentially track and evaluate each point individually over a single time step, and then start the next time step with a new set of collocation points. Hence, deforming coordinate systems and continuous particle tracking are avoided. The resulting approach can be thought of as combining an Eulerian-Lagrangian method with LESCO, and we will call the method ELLESCO. We will demonstrate that ELLESCO accurately solves the first-order hyperbolic problem. This step has been described as the main source of error in fractional step or Eulerian-Lagrangian methods (ELM) [2, 9, 12].

In ELLESCO, spatial coordinates are transformed to the characteristics of the first-order hyperbolic portion of the equation, and the first-order term does not explicitly appear in the transformed system. Information at the base of the collocation point characteristic is used in computing the residual associated with that collocation point. Despite these similarities, ELLESCO is different from the Eulerian-Lagrangian methods described earlier. Although information is brought from the previous time level along characteristics, an intermediate solution is not required or computed. ELLESCO is also easily formulated as a central difference approximation in time, as opposed to the backward difference used by most ELMs. It is worth noting that the characteristic that is computed

includes all of the first-order terms, such as those that may arise from a spatially varying dispersion coefficient.

PROBLEM STATEMENT

The general form of the parabolic partial differential equation with one spatial dimension is

$$\frac{\partial U(x,t)}{\partial t} + A(x,t)\frac{\partial U(x,t)}{\partial x} + B(x,t)\frac{\partial^2 U(x,t)}{\partial x^2} + C(x,t)U(x,t) + D(x,t) = 0 \tag{1}$$

where $U(x,t)$ is an unknown scalar field. The boundary conditions can be written

$$G_l U(x,t) = \bar{a}(x,t) \qquad x = x_l \tag{2}$$

where $G_l \equiv$ boundary operator, $\bar{a}(x,t) \equiv$ specified boundary value, $x_l \equiv$ boundary, and $l \equiv$ boundary index, which can be used to specify different conditions at the ends of the domain.

COORDINATE SYSTEM TRANSFORMATION

A coordinate system transformation can be used to eliminate the troublesome first-order term from eq. (1). The appropriate transformation can be found by solving for the characteristics of the first-order equation

$$\frac{\partial U(x,t)}{\partial t} + A(x,t)\frac{\partial U(x,t)}{\partial x} = 0. \tag{3}$$

The resulting characteristic equations lead to the transformation equations from (x,t) space to (ξ,τ) space, that is

$$d\xi = dx - A(x,t)\,dt \tag{4a}$$

$$\tau = t. \tag{4b}$$

The coordinate system transformation defined by eqs. (4a, b) allows the transformation of the partial derivatives of eq. (1) into (ξ,τ) space:

$$\frac{\partial U}{\partial t} = \frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial t} + \frac{\partial U}{\partial \tau}\frac{\partial \tau}{\partial t} = \frac{\partial U}{\partial \tau} - A(\xi,\tau)\frac{\partial U}{\partial \xi} \tag{5a}$$

$$\frac{\partial U}{\partial x} = \frac{\partial U}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial U}{\partial \tau}\frac{\partial \tau}{\partial x} = \frac{\partial U}{\partial \xi} \tag{5b}$$

$$\frac{\partial^2 U}{\partial x^2} = \frac{\partial}{\partial \xi}\left(\frac{\partial U}{\partial \xi}\right)\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \tau}\left(\frac{\partial U}{\partial \xi}\right)\frac{\partial \tau}{\partial x} = \frac{\partial^2 U}{\partial \xi^2}. \tag{5c}$$

The combination of eqs. (1), (4), and (5) yields the partial differential equation in the (ξ,τ) system:

$$\frac{\partial U(\xi,\tau)}{\partial \tau} + B(\xi,\tau)\frac{\partial^2 U(\xi,\tau)}{\partial \xi^2} + C(\xi,\tau)U(\xi,\tau) + D(\xi,\tau) = 0. \tag{6}$$

Not only has the first order term been eliminated in the transformed coordinate system, but also ξ is a constant along the characteristics of eq. (3) (i.e., when $dx - A(x,t)\,dt = 0$), and the partial derivatives with respect to the spatial coordinate are invariant under the coordinate system transformation.

FINITE DIFFERENCE IN TIME

Equation (6) can be modified using the following finite difference approximation in time:

$$\begin{aligned}\frac{U(\xi,\tau_n) - U(\xi,\tau_{n-1})}{\Delta\tau} + \Theta B(\xi,\tau_n)\frac{\partial^2 U(\xi,\tau_n)}{\partial\xi^2} + \\ \Theta C(\xi,\tau_n)U(\xi,\tau_n) + \Theta D(\xi,\tau_n) + \\ (1-\Theta)B(\xi,\tau_{n-1})\frac{\partial^2 U(\xi,\tau_{n-1})}{\partial\xi^2} + \\ (1-\Theta)C(\xi,\tau_{n-1})U(\xi,\tau_{n-1}) + (1-\Theta)D(\xi,\tau_{n-1}) = 0\,.\end{aligned} \tag{7}$$

If $\Theta = 1/2$, the partial derivative with respect to time is a central difference. As noted above, a line of constant ξ corresponds to a characteristic line of eq. (3) in the (x,t) plane (Figure 1). The starting point of the characteristics is arbitrary, so, for convenience, we choose $\xi = x$ at $\tau_n = t_n$.

The spatial coordinate in (x,t) space which corresponds to ξ at the t_{n-1} time level is defined as x^*. Or equivalently, x^* is the spatial coordinate at the time t_{n-1} of the characteristic which originated from (x,t_n), and it can be computed

$$x^* = x + \int_{t_n}^{t_{n-1}} A(x,t)\,dt\,. \tag{8}$$

In the examples that follow, the solution of eq. (8) is trivial, because $A(x,t)$ is constant. However, when $A(x,t)$ is variable, some type of numerical integration

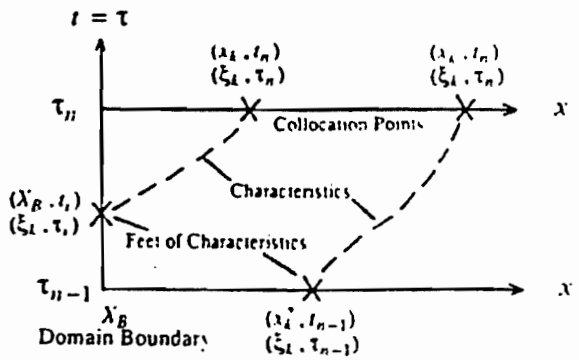

FIG. 1. Spatial relationship between the (ξ,τ) and (x,t) coordinate systems. Dashed lines are the characteristics of eq. (3) and represent traces of constant ξ. Collocation points are tracked along characteristics until they intersect the last time level or a domain boundary.

will be required, and, depending on the complexity of $A(x, t)$ this may cause some difficulties. Although this is an issue which must be addressed by all ELMs, for the remainder of this paper an exact solution of eq. (8) is assumed.

Equations (5b, c) demonstrate that the spatial derivatives are invariant under the coordinate system transformation, and so we can replace the partial derivatives with respect to ξ with partial derivatives with respect to x. From the above discussion, we see $(\xi, \tau_n) = (x, t_n)$ and (ξ, τ_{n-1}) transforms to (x^*, t_{n-1}). Hence, eq. (7) can be rewritten

$$\frac{U(x,t_n) - U(x^*,t_{n-1})}{\Delta t} + \Theta B(x,t_n)\frac{\partial^2 U(x,t_n)}{\partial x^2} + \Theta C(x,t_n)U(x,t_n) + \Theta D(x,t_n) + (1-\Theta)B(x^*,t_{n-1})\frac{\partial^2 U(x^*,t_{n-1})}{\partial x^2} + (1-\Theta)C(x^*,t_{n-1})U(x^*,t_{n-1}) + (1-\Theta)D(x^*,t_{n-1}) = 0. \quad (9)$$

Eq. (1) is now approximated entirely in terms of the untransformed coordinate system.

Recapitulating, we started with the general parabolic partial differential equation and rewrote it in terms of a transformed coordinate system, thereby eliminating the troublesome first order term. We then formed a finite difference in time approximation. We noted that lines of constant ξ were simply the characteristics of a first-order partial differential equation and that the partial derivatives with respect to the spatial coordinate were invariant under the transformation. This allowed us to write the difference equation in terms of the untransformed coordinate system, and this is presented as eq. (9).

LEAST SQUARES COLLOCATION

LESCO uses least squares to minimize the residual errors of eq. (9) at a set of discrete locations called collocation points. To compute the residuals, we must specify the form of the trial function. Collocation procedures require trial functions with C^1 continuity even at element boundaries. Cubic Hermites are well known functions with this property (p. 67, Lapidus and Pinder [16]). Employing cubic Hermites, the scalar field is approximated by

$$\hat{U}(x,t) = \sum_{i=1}^{nbas} \alpha_i(t)\Phi_i(x) \quad (10)$$

where

$\Phi_i(x) \equiv$ cubic Hermite basis functions (two per node),
$\alpha_i(t) \equiv$ coefficients of basis function i at time t (two per node), and
$nbas \equiv$ the number of basis functions.

The trial function is substituted into eq. (9) and is evaluated at each collocation point to form a residual. Similarly, the residuals associated with colloca-

tion points on the boundary are computed by substituting the trial function into eq. (2). The residuals are squared and summed, that is,

$$\varepsilon = \sum_{k=1}^{ncol} w_k R_k^2 + \sum_{l=1}^{nb} w_l R_l^2 \tag{11}$$

where $R_k \equiv$ residual associated with the kth interior collocation point, $R_l \equiv$ residual associated with the lth boundary collocation point, $ncol \equiv$ number of interior collocation points, $nb \equiv$ number of boundary collocation points, and $w \equiv$ weight assigned to a collocation point.

To minimize the sum of the squares of the errors, the derivatives with respect to the coefficients of the trial function, $\alpha_j(t_n)$, are set equal to zero:

$$\sum_{k=1}^{ncol} w_k R_k \frac{\partial R_k}{\partial \alpha_j(t_n)} + \sum_{l=1}^{nb} w_l R_l \frac{\partial R_l}{\partial \alpha_j(t_n)} = 0 \qquad j = 1, nbas. \tag{12}$$

Equation (12) forms the system of equations to be solved.

At this point it is convenient to introduce the operator L:

$$LU(x,t) = B(x,t)\frac{\partial^2 U(x,t)}{\partial x^2} + C(x,t)U(x,t) + D(x,t). \tag{13}$$

To evaluate the residual at a typical interior collocation point, the approximate scalar function, eq. (10), is substituted into the difference equation, eq. (9). The resulting expression is evaluated at both the collocation point locations (x_k, t_n), and the associated locations at the last time level, (x_k^*, t_{n-1}), that is

$$\begin{aligned} R_k = {} & \frac{1}{\Delta t}\sum_{i=1}^{nbas} \alpha_i(t_n)\Phi_i(x_k) - \frac{1}{\Delta t}\sum_{i=1}^{nbas} \alpha_i(t_{n-1})\Phi_i(x_k^*) \\ & + \Theta\left\{\sum_{i=1}^{nbas} \alpha_i(t_n)L\Phi_i(x_k)\right\} + (1-\Theta)\left\{\sum_{i=1}^{nbas} \alpha_i(t_{n-1})L\Phi_i(x_k^*)\right\}. \end{aligned} \tag{14}$$

Equation (14) holds for any collocation point whose characteristic curve remains within the interior of the spatial domain throughout the time interval $t_{n-1} \le t \le t_n$, such as the right collocation point in Figure 1. It then follows

$$\frac{\partial R_k}{\partial \alpha_j(t_n)} = \left(\frac{1}{\Delta t} + \Theta L\right)\Phi_j(x_k). \tag{15}$$

When, as for the left collocation point in Figure 1, the characteristic associated with the collocation point intersects a domain boundary at (x_B, t_i), eqs. (14) and (15) are no longer valid. The time increment is no longer Δt, but becomes

$$\Delta t_i = t_n - t_i. \tag{16}$$

Also, the terms in eq. (9) that were evaluated at (x_k^*, t_{n-1}) must now be evaluated at (x_B, t_i). However, a solution does not exist at t_i so that $\hat{U}(x_B, t_i)$ and $L\hat{U}(x_B, t_i)$ must be approximated. To this purpose, the interpolation parameter β

is introduced

$$\beta = \frac{t_i - t_{n-1}}{\Delta t}. \tag{17}$$

Consider first the case of a non-first type boundary. The parameter β is used to write the following approximations:

$$\hat{U}(x_B, t_i) = \beta \hat{U}(x_k, t_n) + (1 - \beta)\hat{U}(x_B, t_{n-1}) \tag{18}$$

$$L\hat{U}(x_B, t_i) = \beta L\hat{U}(x_k, t_n) + (1 - \beta)L\hat{U}(x_B, t_{n-1}). \tag{19}$$

Substituting eqs. (16), (18), and (19) into eq. (9), we arrive at the residual for a collocation point whose characteristic crosses a non-first type boundary:

$$\begin{aligned} R_k = {} & \frac{1 - \beta}{\Delta t_i} \sum_{i=1}^{nbas} \alpha_i(t_n)\Phi_i(x_k) - \frac{1 - \beta}{\Delta t_i} \sum_{i=1}^{nbas} \alpha_i(t_{n-1})\Phi_i(x_B^*) \\ & + (\Theta + (1 - \Theta)\beta)\left\{\sum_{i=1}^{nbas} \alpha_i(t_n)L\Phi_i(x_k)\right\} \\ & + (1 - \Theta)(1 - \beta)\left\{\sum_{i=1}^{nbas} \alpha_i(t_{n-1})L\Phi_i(x_B^*)\right\}. \end{aligned} \tag{20}$$

Equation (19) also holds for a first type boundary, but Eq. 18 is replaced by:

$$\hat{U}(x_B, t_i) = \overline{U}(t_i) \tag{21}$$

where $\overline{U}(t_i)$ is the specified boundary value at time t_i. This leads to the residual for a collocation point whose characteristic crosses a first type boundary:

$$\begin{aligned} R_k = {} & \frac{1}{\Delta t_i} \sum_{i=1}^{nbas} \alpha_i(t_n)\Phi_i(x_k) - \frac{1}{\Delta t_i}\overline{U}(t_i) \\ & + (\Theta + (1 - \Theta)\beta)\left\{\sum_{i=1}^{nbas} \alpha_i(t_n)L\Phi_i(x_k)\right\} \\ & + (1 - \Theta)(1 - \beta)\left\{\sum_{i=1}^{nbas} \alpha_i(t_{n-1})L\Phi_i(x_B^*)\right\}. \end{aligned} \tag{22}$$

Experience has shown that the preceding approximation works well. Note that Δt_i is never zero, or it would be a boundary collocation point and subject to eq. (2). However, as t_i approaches t_n, the formulation approaches fully implicit. As t_i approaches t_{n-1}, the formulation approaches that of eq. (14).

We now turn to the enforcement of the boundary conditions. Since a collocation procedure is being used, Green's theorem is not applied to the equations, and the first and second type boundary conditions are enforced by the same procedures. Two methods exist for enforcing boundary conditions of the first and second type. The first method, which has been used in the examples presented later, is to enforce the conditions directly in the matrix equations. This can be accomplished because the boundary nodes have a degree of freedom as-

sociated with the value of the function and a degree of freedom associated with the derivative of the function.

The second method is to use a boundary collocation point to enforce the boundary conditions. This is also a method for enforcing a third type boundary condition. Using this approach, the residual associated with a boundary condition is obtained by substituting eq. (10) into eq. (2). The residual, R_l, associated with a boundary collocation point l, is then:

$$R_l = G_l\left\{\sum_{i=1}^{nbas} \alpha_i(t_n)\Phi_i(x_l)\right\} - \bar{a}(x_l, t_n) \tag{23}$$

where G_l is the differential operator that describes the boundary condition that is being enforced. The derivatives of eqs. (20), (22), and (23) follow naturally and are similar in form to eq. (15). Finally, the initial conditions are incorporated by assigning the initial values of $\hat{U}(x_i, t_0)$ at each collocation point. The cubic Hermite basis functions are then least squares fitted to the collocation point values.

All of the relations required to form the system of equations represented by eq. (12) have now been specified. The resulting matrix equation is

$$(\underline{\underline{A}} + \underline{\underline{B}})\underline{\alpha} = \underline{c} + \underline{d}\,, \tag{24}$$

where

$$\begin{aligned} a_{i,j} = &\sum_{k=1}^{n1} w_k\left(\frac{1}{\Delta t} + \Theta L\right)\Phi_j(x_k)\left(\frac{1}{\Delta t} + \Theta L\right)\Phi_i(x_k) \\ &+ \sum_{k=n1+1}^{n2} w_k\left(\frac{1-\beta}{\Delta t_i} + (\Theta + (1-\Theta)\beta)L\right)\Phi_j(x_k) \\ &\cdot\left(\frac{1-\beta}{\Delta t_i} + (\Theta + (1-\Theta)\beta)L\right)\Phi_i(x_k) \\ &+ \sum_{k=n2+1}^{ncol} w_k\left(\frac{1}{\Delta t_i} + (\Theta + (1-\Theta)\beta)L\right)\Phi_j(x_k) \\ &\cdot\left(\frac{1}{\Delta t_i} + (\Theta + (1-\Theta)\beta)L\right)\Phi_i(x_k)\,, \end{aligned} \tag{24a}$$

$$b_{i,j} = \sum_{l=1}^{nb} w_l G_l\Phi_j(x_l) G_l\Phi_i(x_l)\,, \tag{24b}$$

$$\begin{aligned} c_i = &\sum_{k=1}^{n1} w_k\left(\frac{1}{\Delta t} - (1-\Theta)L\right)\left\{\sum_{j=1}^{nbas} \alpha_j(t_{n-1})\Phi_j(x_k^*)\right\}\left(\frac{1}{\Delta t} + \Theta L\right)\Phi_i(x_k) \\ &+ \sum_{k=n1+1}^{n2} w_k\left(\frac{1-\beta}{\Delta t_i} - (1-\Theta)(1-\beta)L\right)\left\{\sum_{j=1}^{nbas} \alpha_j(t_{n-1})\Phi_j(x_B^*)\right\} \\ &\cdot\left(\frac{1-\beta}{\Delta t_i} + (\Theta + (1-\Theta)\beta)L\right)\Phi_i(x_k) \end{aligned} \tag{24c}$$

$$+ \sum_{k=n2+1}^{ncol} w_k \left[\frac{\overline{U}(t_i)}{\Delta t_i} - ((1-\Theta)(1-\beta)\mathrm{L}) \left\{ \sum_{j=1}^{nbas} \alpha_j(t_{n-1}) \Phi_j(x_k^*) \right\} \right]$$

$$\cdot \left(\frac{1}{\Delta t_i} + (\Theta + (1-\Theta)\beta)\mathrm{L} \right) \Phi_i(x_k) .$$

$$d_i = \sum_{k=1}^{nb} w_k \bar{a}(x_k, t_n) \mathrm{G}_k \Phi_i(x_k) . \tag{24d}$$

The three interior collocation point summations associated with $\underline{\underline{A}}$ and $\underline{c}$ are for the interior collocation points whose characteristics remain in the domain (eq. (14)), those which cross non-first type boundaries (eq. (20)) and those which cross first type boundaries (eq. (22)) respectively. The stiffness matrix is symmetric, positive definite and has a bandwidth of seven.

The number and location of collocation points can affect the results of the ELLESCO computation, and the optimum number and location of collocation points remains an open question. Under most circumstances, three collocation points per element spread evenly across the domain will yield excellent results. However, in the presence of large spatial gradients, the solution can be improved by increasing the number of collocation points to five or more. As the number of collocation points increases, the computational effort in tracking and matrix construction increases, but the effort required to solve the resulting matrix equations is unchanged.

The steps required to solve each time step are summarized:

1. Choose the collocation point locations (x_k, t_n).
2. Project the collocation point (x_k, t_n) along its associated characteristic to the previous time level, or until it intersects a spatial boundary.
3. Use the appropriate equation, (14), (20), or (22), to compute the collocation point residual and the derivative of the residual.
4. Add the contribution from step 3 to eq. (12).
5. Repeat for all collocation points.
6. Use eq. (23) to compute the residual and the residual derivatives for each boundary collocation point.
7. Add the contribution from step 6 to eq. (12).
8. Enforce the first and second type boundary conditions in eq. (12).
9. Solve eq. (12) for the coefficients, α_i.
10. Substitute the coefficients, α_i, into eq. (10) to form the C^1 cubic polynomial approximation to the scalar field.

ADVECTIVE-DISPERSIVE TRANSPORT EQUATION

The advective-dispersive transport equation with constant dispersion is written

$$\frac{\partial C(x,t)}{\partial t} + v(t)\frac{\partial C(x,t)}{\partial x} - D\frac{\partial^2 C(x,t)}{\partial x^2} = 0 \tag{25}$$

where $C(x,t) \equiv$ concentration, $v(t) \equiv$ velocity, and $D \equiv$ dispersion coefficient. Equation (25) is a special case of eq. (1), and the method described in the preceding section can be used to solve it. Equation (25) will be used to demonstrate the efficacy of the proposed method and to compare ELLESCO to other methods.

Results of ELLESCO will be compared to the ELM using Lagrangian quadratic basis functions (ELMLQ) which has been described by Baptista et al. [1]. Briefly, the ELMLQ method consists of two steps. First, the node locations of the finite-element mesh are tracked to the last time step. The values at the foot of the characteristic are interpolated using the nodal values of the last time step and then assigned to the node, forming an intermediate solution. Second, the dispersion portion of the equation is solved using the intermediate solution.

An ELM using cubic Hermite base functions (ELMCH) will also be discussed. The ELMCH method was first proposed by Holly and Preissmann [9]. In this method, both the function value and the value of the derivative are obtained by tracking along characteristics to the last time step. The intermediate solution is constructed by using these values to set the values of the two degrees of freedom associated with the cubic Hermites at the nodes. Again, the second step solves the dispersion problem using the intermediate solution. The results for Galerkin finite elements with Lagrangian quadratic basis functions (GALERKIN) will also be displayed for reference.

RESULTS

The results of five test problems are presented in Figures 2 through 6. In all five test problems $v = 0.5$ and $t = 9600$. Analytic solutions are represented by solid lines. Figures 2, 3, and 4 illustrate the advection, $D = 0$, of a Gaussian plume with standard deviation $\sigma = 264$. In Figure 2, the element length is 200, $\Delta t = 96$ and the number of collocation points per element (ncol) is three. The ELLESCO result is indistinguishable from the analytic. The GALERKIN solution has some asymmetry and slight dispersion. Both of the ELM solutions are significantly dispersed, and ELMLQ is oscillating.

In Figure 3, the element length has been increased to 500 and ncol = 5. ELLESCO now shows a 4.6% accumulated error in the peak concentration.

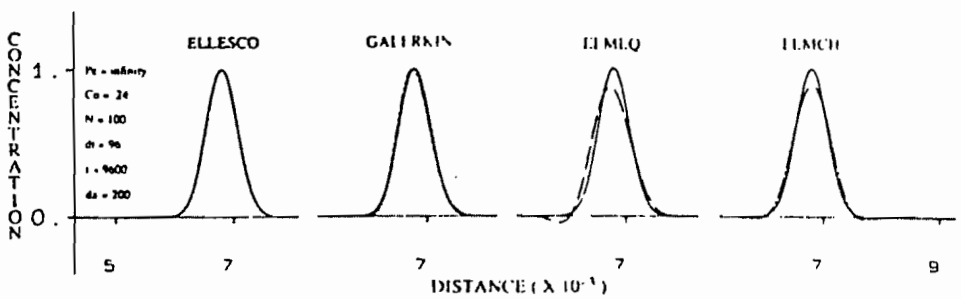

FIG. 2. Comparison of advected Gaussian plumes. Solid lines are analytic solution. Broken lines are computed solutions.

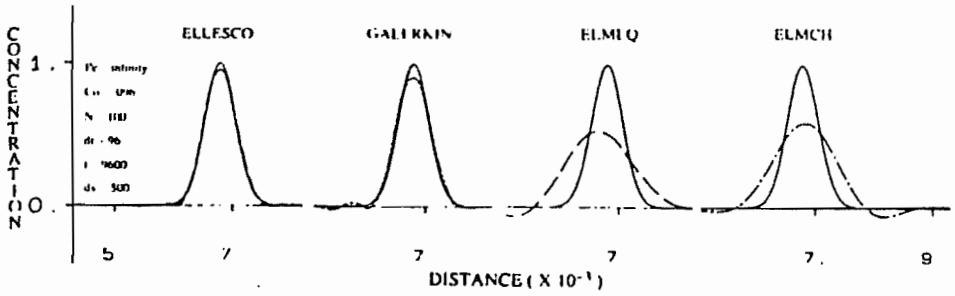

FIG. 3. Comparison of advected Gaussian plumes with increased element length. Solid lines are analytic solution. Broken lines are computed solutions.

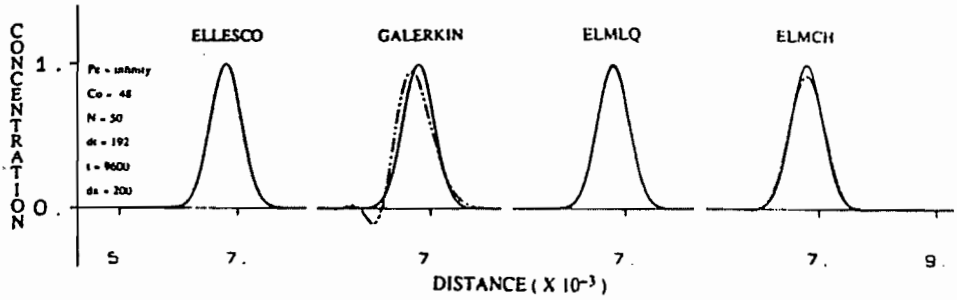

FIG. 4. Comparison of advected Gaussian plumes with increased time step increment. Solid lines are analytic solution. Broken lines are computed solutions.

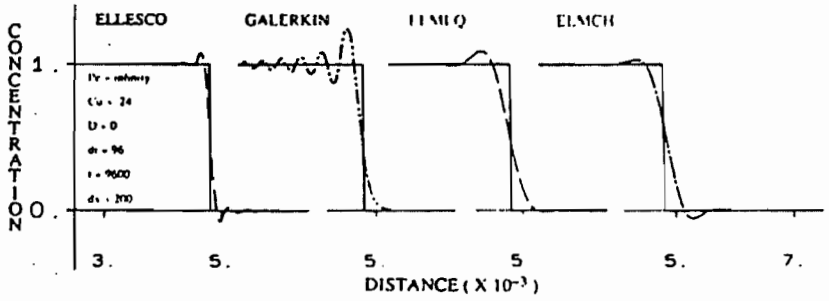

FIG. 5. Comparison of advected fronts. Solid lines are analytic solution. Broken lines are computed solutions.

This error can be reduced to less than 3.5% by taking ncol = 8. In addition, a slight oscillation has developed. The GALERKIN solution has deteriorated in a similar manner, but to a greater degree. The ELM solutions are greatly dispersed and have large oscillations. Note that the ELLESCO solution is still superior to the ELM solutions in Figure 2, where the element length is only 200.

The result in Figure 4 has the element length reset to dx = 200, ncol = 3, and the time increment is doubled to Δt = 192. The advantages of the ELMs

are visible, and the two ELM solutions are quite improved. The ELMLQ solution is superior to the ELMCH solution, but this is due to the Lagrangian quadratic node spacing of 100 yielding Co = 0.96 which is close to 1. When Co = 1, all methods display increased accuracy. The Galerkin solution is dispersed and oscillatory. The ELLESCO solution is essentially exact.

Figures 2 and 3 demonstrate the effects of accumulated interpolation error on the ELMs when many time steps are required and the spatial discretization is large. In these circumstances the GALERKIN solution can be superior. To a much lesser extent, ELLESCO will behave similarly to the ELMs, but appears to at least match the accuracy of the GALERKIN solution. However, when the Co increases, the ELMs become superior to the Galerkin. In all cases, the ELLESCO solution is the best.

Solutions to an advancing front are illustrated in Figures 5 and 6. Initial conditions are $C(x,0) = 0$ and a source of $C(0,t) = 1$ is set at the left boundary. The element length is 200 and ncol = 5. Figure 5 represents the solution to the pure advection (D = 0) problem. The ELLESCO solution has virtually no dispersion. An 8% oscillation has developed ahead and behind the front, and it is mainly due to the Gibb's phenomena. The GALERKIN solution is more dispersed and also has serious trailing oscillations. Both ELMs are seriously dispersed, as well as contaminated by oscillations.

The dispersion coefficient was increased to D = 0.5 for the problem represented by Figure 6. This corresponds to a grid Peclet number of 200 (Pe = 100 for ELMLQ and GALERKIN since the node spacing is half the element length). All the solutions have improved, however only the ELLESCO solution no longer has oscillations or artificial dispersion.

CONCLUSION

An accurate method for the numerical solution of parabolic PDEs with large and variable first-order terms has been presented. The method combines least squares collocation with a coordinate transformation to eliminate the asymmetric first order term associated with numerical oscillations.

The method has several appealing features. Since it is a collocation method, the coordinate transformation can be reduced to tracking individual collocation

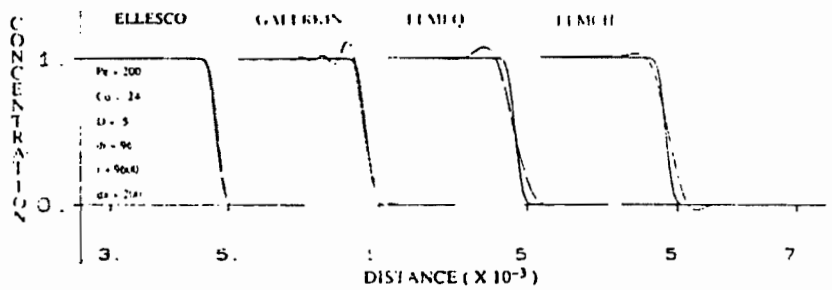

FIG. 6. Comparison of transported fronts. Solid lines are analytic solution. Broken lines are computed solutions.

points. The tracking and the formation of the collocation point residuals is easily done on a point-by-point basis. Deforming grids and continuous particle tracking are avoided. Similarly, the position and number of collocation points can be varied at each time step in order to adjust to changing features, while minimizing computational effort. The method automatically adjusts to spatial and temporal changes in the first order term, allowing it to work well in advection and dispersion dominated flows. As with ELMs, there is no inherent limitation in the size of the time step, and Courant numbers can be greater than 1. In fact, in advection dominated flow, large time steps can improve the solution. Although tracking is required, ELLESCO is a single step method and no intermediate solution is generated. Besides the simplicity of application, the least squares format also provides a natural error estimate in the form of the residuals which are minimized by the procedure.

ELLESCO is computationally competitive in one spatial dimension and may provide significant savings in multi-spatial dimensions. Comparison of Figures 2 and 3 indicates that the ELLESCO results are superior to the ELMLQ results with half the element length. Given that ELLESCO can use twice the element length, the rank of the ELLESCO matrix will be one half the rank of the ELMLQ matrix. The ELLESCO matrix bandwidth is 7 compared to the 5 of the ELMLQs. The number of collocation points per element that must be tracked will generally be 3, but may increase to 5 or 6 in difficult areas. In contrast, ELMLQ is required to track 4 nodes over the same interval.

We wish to acknowledge support from the U.S. Department of Energy, grant number DE-FG02-86ER60453, the Industrial Support Group for Groundwater Contamination Research and the Air Force Office of Scientific Research, Bolling AFB, D.C.

References

[1] A. E. Baptista, E. E. Adams, and K. D. Stolzenbach, *Eulerian-Lagrangian Analysis of Pollutant Transport in Shallow Water*, Ralph M. Parsons Laboratory, MIT, Rpt. No. 296, 1984.

[2] A. E. Baptista, *Solution of Advection-Dominated Transport by Eulerian-Lagrangian Methods Using the Backwards Method of Characteristics*, Ph.D. dissertation, M.I.T., 1987.

[3] Ching-Lung Chang and M. D. Gunzburger, "A finite element method for first order elliptic systems in three dimensions," *Appl. Math. Comp.*, 23, 171–184 (1987).

[4] J. Douglas, Jr. and T. F. Russell, Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures, *SIAM J. Numer. Anal.*, 19, 871–885 (1982).

[5] D. E. Eason, "A review of least-squares methods for solving partial differential equations," *Int. J. Numer. Meth. Eng.*, 10, 1021–1046 (1976).

[6] R. E. Ewing and T. F. Russell, "Multistep Galerkin methods along characteristics for convection-diffusion problems," in *Advances in Computer Methods for Partial Differential Equations*, Vol. IV, IMACS, Rutgers University, New Brunswick, 1981, pp. 28–36.

[7] J. Glass and W. Rodi, "A higher order numerical scheme for scalar transport," *Comp. Meths. Appl. Mech. Eng.*, 31, 337–358 (1982).

[8] P. J. Harley and A. R. Mitchell, "A finite element collocation method for the exact control of a parabolic problem," *Int. J. Numer. Meths. Eng.*, **11**, 345–353 (1977).

[9] F. M. Holly, Jr. and A. Preissmann, "Accurate calculation of transport in two dimensions, *J. Hydr. Div., ASCE*, HY11, 1259–1277 (1977).

[10] F. M. Holly, Jr. and T. Komatsu, "Derivative approximations in the two-point fourth order method for pollutant transport," in *Proceedings of the Conference on Frontiers in Hydraulic Engineering*, ASCE, MIT, Cambridge, 1983, pp. 349–355.

[11] E. N. Houstis and R. E. Lynch, "Evaluation of numerical methods for elliptic partial differential equations," *J. Comp. Phys.*, **27**, 323–350 (1978).

[12] J. P. Huffenus and D. Khaletzky, "The Lagrangian approach of advective term treatment and its application to the solution of Navier-Stokes equations," *Int. J. Numer. Meths. Fluids*, **1**, 365–387 (1981).

[13] B. Joos, *The Least Squares Collocation Method for Solving Partial Differential Equations*, Ph.D. dissertation, Princeton University, 1986.

[14] T. Komatsu, F. M. Holly, Jr., N. Nakashiki, and K. Ohgushi, "Numerical calculation of pollutant transport in one and two dimensions," *J. Hydroscience Hydr. Eng.*, 3, 15–30 (1985).

[15] W. L. Kwok and Y. K. Cheung, "Application of least square collocation technique in finite element and finite strip formulation," *Int. J. Numer. Meths. Eng.*, **11**, 1391–1404 (1977).

[16] L. Lapidus and G. Pinder, *Numerical Solution of Partial Differential Equations in Science and Engineering*, John Wiley & Sons, 1982, 677 pp.

[17] S. P. Neuman, "A Eulerian-Lagrangian numerical scheme for the dispersion-convection equation using conjugate space-time grids," *J. Comp. Phys.*, 41, 270–294 (1981).

[18] S. P. Neuman and S. Sorek, "Eulerian-Lagrangian methods for advection-dispersion," *Finite Elements in Water Resources*, 4, 14.41–14.68 (1982).

[19] K. O'Neil and D. R. Lynch, "Effective and highly accurate solution of diffusion and convection-diffusion problems using moving, deforming grids," in *Proceedings of the 3rd International Conference on Finite Elements in Water Resources*, University of Mississippi, Oxford, 1, 3.67–3.76 (1980).

[20] E. Rosenblueth, "Physical control of numerical solution of parabolic equations," *Eng. Anal.*, 2, 107–110 (1985).

[21] K. Toda and F. M. Holly, "Hybrid numerical method for linear advection-diffusion," *Microsoftware Eng.*, 3, 199–205 (1987).

[22] E. Varoglu, "A finite element model incorporating characteristics for the diffusion-convection equation," in *Proceedings of the 3rd International Conference on Finite Elements in Water Resources*, University of Mississippi, Oxford, 1, 3.3–3.19 (1980).

[23] E. Varoglu and W. L. Finn, "A finite element method for the diffusion-convection equation with constant coefficients," *Adv. Water Resources*, 1, 337–343 (1978).

[24] E. Varoglu and W. L. Finn, "Finite elements incorporating characteristics for one-dimensional diffusion-convection equation," *J. Comput. Phys.*, 34, 371–389 (1980).