An explicit unconditionally stable numerical solution of the advection problem in irrotational flow fields

Alessandra Bascià

Department of Mechanics and Materials, Mediterranean University of Reggio Calabria, Reggio Calabria, Italy

Tullio Tucciarelli

Department of Hydraulic Engineering and Environmental Applications, University of Palermo, Palermo, Italy

[1] A new methodology for the Eulerian numerical solution of the advection problem is proposed. The methodology is based on the conservation of both the zero- and the first-order spatial moments inside each element of the computational domain and leads to the solution of several small systems of ordinary differential equations. Since the systems are solved sequentially (one element after the other), the method can be classified as explicit. The proposed methodology has the following properties: (1) it guarantees local and global mass conservation, (2) it is unconditionally stable, and (3) it applies second-order approximation of the concentration and its fluxes inside each element. Limitation of the procedure to irrotational flow fields, for the 2-D and 3-D cases, is discussed. The results of three 1-D and 2-D literature tests are compared with those obtained using other techniques. A new 2-D test, with radially symmetric flow, is also carried out.   *INDEX TERMS:* 1829 Hydrology: Groundwater hydrology; 1832 Hydrology: Groundwater transport; 1831 Hydrology: Groundwater quality; 3230 Mathematical Geophysics: Numerical solutions; *KEYWORDS:* advection, numerical techniques, transport

1. Introduction

[2] The numerical simulation of advection processes is a crucial issue for many groundwater modeling applications. This is because (1) many transport problems can be reduced to their advective component, which is affected by the only uncertainty of the flow field, and (2) the more general advection-diffusion equation is often solved by splitting techniques, in which the solution of the diffusive component is usually the easiest [*Abbott*, 1979; *Holly and Preissmann*, 1977].

[3] In spite of the importance of the advection problem, computational difficulties remain for its numerical solution, mainly for 2-D and 3-D cases, where nonstructured grids and irregular elements are used.

[4] The available methods can be classified as Eulerian and Lagrangian. Eulerian methods compute the unknown function at the nodes or at the elements of a computational mesh fixed in space, after spatial and time discretization of the PDE. It is well known that the classical Eulerian finite difference or finite element methods provide numerical solutions affected by numerical diffusion or oscillations [*Bella and Grenney*, 1970; *Gray and Pinder*, 1983; *Venezian*, 1984]. A reduction of numerical diffusion can be obtained by evaluating the spatial derivatives starting from the function values at distant grid points, as in the QUICK and QUICKEST procedure [*Leonard*, 1979], that can be coupled to a limiting algorithm like ULTIMATE

[*Leonard*, 1991], aiming at eliminating spurious oscillations. A popular Eulerian approach is the so-called Runge-Kutta discontinuous Galerkin (RKDG) method [*Cockburn and Shu*, 1998]. The RKDG combines a piecewise linear discontinuous finite element spatial approximation of the unknown function with a time discretization that guarantees the so-called "total variation diminishing (TVD)" property, that is an increasing (in time) spatial regularity of the solution. After a spatial discretization of the equation that guarantees approximate Riemann fluxes along the element discontinuities, the proposed time discretization provides, with simple low-order matrix operations, a system of ODEs that can be solved, for given temporal step, using a Runge-Kutta high-order accurate scheme. Other methods are the streamline upwind Petrov-Galerkin [*Brooks and Hughes*, 1982], the Taylor-Galerkin [*Donea*, 1984] and the Galerkin-least squares schemes [*Hughes et al.*, 1989]. If a fully implicit time discretization is used in Eulerian methods, no limitation exists for the choice of the time step; however, this technique requires the solution of large nonsymmetric algebraic systems for each time step, with a fast growth of the numerical effort with the number of elements. Most of the recently adopted Eulerian methods use explicit time discretization, are second-order accurate, but have limitations on the size of the Courant number, that must always be taken smaller than one, corresponding to the well-known Courant-Friedrichs-Lewy (CFL) condition. An exception is an algorithm for the finite difference solution of the 1-D case, proposed by *Ponce et al.* [1979]. The CFL condition fulfillment does not produce a large increment of the numerical effort in structured meshes, because it is not

affected by the number of elements. Nevertheless, it can limit the efficiency of the algorithm in nonstructured meshes, obtained by automatic mesh generators. In this case, the existence of even a single small element can require the use of a small time step for all the elements, with a strong increment of the computational effort and a potential loss of accuracy due to the use of very small Courant numbers in some parts of the domain. The Courant number can be held within the stability limit using different grid refinement techniques in computational elements with changing velocity values [*Sobey*, 1984], but this lead to very large time-consuming procedures, specially in the 2-D and 3-D cases.

[5] In the Lagrangian approaches, the computational grid is not fixed in space, moving along the characteristic lines with the same velocity of the flow field. Then, it is necessary to accurately track the fluid particles and to evaluate their trajectories during their motion. The particle tracking technique is often the critical point of the procedure [*Oliveira and Baptista*, 1998]. Both forward and backward tracking techniques have been proposed in the past, but the search for an accurate method is still in progress [*Bensabat et al.*, 2000; *Pokrajac and Lazic*, 2002]. An example of Lagrangian approach is the second moment method, originally developed for the case of air pollution [*Egan and Mahoney*, 1972] and subsequently applied also to shallow flow [*Nassiri and Babarutsi*, 1997]. The disadvantage of the Lagrangian approaches is that they are difficult to be applied to 2-D and 3-D problems with irregular boundaries and heterogeneous domains. Moreover, many Lagrangian techniques do not guarantee mass conservation.

[6] In the semi-Lagrangian approaches for the solution of the advection-diffusion problem, the use of Lagrangian techniques is restricted to the advective component and limited to the time step used for the solution of the next diffusive problem. The method of characteristics (MOC) solves the advective problem by locating, at each time step, the foot of the characteristic line ending in each node of a fixed grid. This means that a potentially dissipative interpolation of the known values of the surrounding grid nodes must be performed. Many examples of similar methods can be found in the literature, like the two-point fourth-order interpolation [*Holly and Preissmann*, 1977], the finite element characteristic [*Wang et al.*, 1988], the minimax characteristics [*Li*, 1990] methods, as well as the cubic-spline interpolation [*Schohl and Holly*, 1991], or the characteristic Galerkin scheme, recently developed for the 3-D case also [*Kaazempur-Mofrad and Ethier*, 2002]. The MOC based techniques have the advantage of allowing the use of large time steps and preserving the solution from oscillations and numerical diffusion, but they seldom guarantee local and global mass balance [*Chilakapati*, 1999]. Volume tracking techniques [*Van Leer*, 1977; *Rider and Kothe*, 1998] follow the volume evolution of the mass initially present inside each element of the mesh and guarantee its conservation. The semi-Lagrangian approaches, also known as Eulerian-Lagrangian approaches, are nowadays probably the most popular tools for the solution of the advection problem equations [*Celia et al.*, 1990; *Healy and Russell*, 1993].

[7] Recently, some grid-free methods have been proposed, based on the interpolation of scattered data [*Behrens*

and Iske, 2002], but they suffer of the same limitations of the Lagrangian methods.

[8] In the following section an Eulerian procedure is proposed, that embraces several advantages of the Eulerian and Lagrangian approaches. It is explicit, mass conservative, but also unconditionally stable with respect to the Courant number. It adopts a second-order approximation of the unknown advected function in each computational element and shows an average convergence order equal to 2 in 1-D numerical tests with smooth initial concentration. The procedure is restricted to the case of irrotational flow fields, as it is more extensively discussed in section 4. In section 2 the proposed algorithm is presented for the 1-D case, in section 3 it is extended to the 2-D case. Extension from the 2-D to the 3-D case is straightforward. Some benchmark problems are solved in each section, in order to show the features of the method.

2. Algorithm in the 1-D Case

[9] The 1-D advection equation of the unknown concentration function c is:

$$\frac{\partial c}{\partial t} - u \frac{\partial c}{\partial x} = 0, \tag{1}$$

where u is the known velocity and x and t are space and time independent variables. Equation (1) is defined in the $0 \le x \le L, 0 \le t \le T$ integration domain. A unique solution exists given the initial $c(x, 0)$ and the upstream boundary $c(0, t)$ condition.

[10] Divide the space domain in N computational elements, with length $\Delta x = L/N$. Assume a velocity u constant in each element. Call x_1^e, x_2^e the coordinates $(e - 1)\Delta x$, $e\Delta x$ of the two extreme x values of the eth element. Divide also the time domain in N_t time steps, with extension $\Delta t = T/N_t$. Call t_k the time level at the end of the kth time step ($t_k = k\Delta t$). Sort all the elements along the downstream direction. If u is positive everywhere, the ordered sequence is $I_1 = 1$, $I_2 = 2, \ldots, I_N = N$.

[11] Assume a piecewise spatial linear approximation $c^e(x, t)$ of the function c inside each element e at any time (Figure 1). Observe that the function $c^e(x, t)$ can be discontinuous at the node between two elements. Set $c_1^e(t) = c^e(x_1^e, t)$, $c_2^e(t) = c^e(x_2^e, t)$. Assume also a known low-order polynomial time approximation $\xi_2^{e-1}(t)$ of the concentration at the second node of the upstream element $e - 1$ from time level t_k to time level t_{k+1}. A third-order polynomial is used in the implemented code, that is:

$$\xi^e = \xi^{e,0} + \xi^{e,1}t - \xi^{e,2}t^2 - \xi^{e,3}t^3. \tag{2}$$

where ξ^e is a vector with components ξ_1^e and ξ_2^e, that are the time approximations of c at the two nodes of element e.

[12] It is possible to discretize, in each element, the PDE (1) in a system of two ODEs. The first equation is derived from the mass balance between the entering flux, the leaving flux and the average concentration inside the element. The second equation is derived from the conservation of the spatial first-order moment inside the element. The moment is computed with respect to the second node of
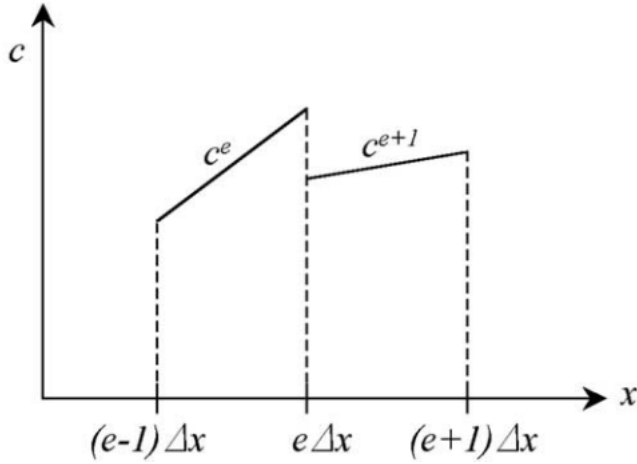
Figure 1. Piecewise linear approximation in space of the concentration c in the 1-D case.

element e, with space coordinate x_2^e. The first equation is given by:

$$\frac{d\left(c_1^e(t) + c_2^e(t)\right)}{dt} - c_1'^e(t) - c_2'^e(t) \approx \left(\xi_2^{e-1}(t) - c_2^e(t)\right)\frac{2u^e}{\Delta x} \quad (3)$$

where u^e is the velocity in element e. Call $\overline{M^e}$ the moment of the linear approximation of c inside element e with respect to its downstream end. Its value is, at any time,

$$\overline{M^e} = c_2^e(t)\frac{\Delta x^2}{2} - \frac{c_1^e(t) - c_2^e(t)}{3}\Delta x^2. \quad (4)$$

[13] Its change in time is due to the variation of both $c_1^e(t)$ and $c_2^e(t)$. These two variations are not independent, but are linked by the advection equation (1). At time level $t + dt$ the moment of the real concentrations corresponding to the piecewise approximation of c at time t is equal to (see Figure 2):

$$M^e(t + dt) - \int_0^{\Delta x} \left[\left(c_2^e(t) + \frac{c_1^e(t) - c_2^e(t)}{\Delta x}s\right)\right.$$

$$\left. \cdot (s - u^e dt)\right] ds + u^e \xi_2^{e-1}(t)dt\,\Delta x - O(dt^2). \quad (5)$$

To obtain the identity:

$$\overline{M^e}(t + dt) - M^e(t + dt). \quad (6)$$

the following equation has to be solved:

$$\frac{d\overline{M^e}}{dt} - \frac{M^e(t + dt) - \overline{M^e}(t)}{dt}. \quad (7)$$

Merging equations (4) and (5) in equation (7) you obtain:

$$\frac{d\overline{M^e}}{dt} - \Delta x\,u^e\left(\xi_2^{e-1}(t) - \frac{c_1^e(t) + c_2^e(t)}{2}\right). \quad (8)$$

Merging equation (4) in equation (8) you obtain:

$$\frac{c_2'^e(t)}{2} + c_1'^e(t) - \left(\xi_2^{e-1}(t) - \frac{c_1^e(t) + c_2^e(t)}{2}\right)\frac{3\,u^e}{\Delta x}. \quad (9)$$

[14] Equation (3) can be coupled with equation (9) and solved in the $c_1^e(t)$, $c_2^e(t)$ unknowns. The system can be written in normal form as:

$$c_1'^e = \left(4\xi_2^{e-1} - c_2^e - 3c_1^e\right)\frac{u^e}{\Delta x}, \quad (10)$$

$$c_2'^e - \left(-2\xi_2^{e-1} - c_2^e + 3c_1^e\right)\frac{u^e}{\Delta x}, \quad (11)$$

and in matrix form as:

$$\frac{d\mathbf{c}^e}{dt} = \mathbf{A}^e\mathbf{c}^e - \mathbf{b}(t). \quad (12)$$

[15] According to the polynomial form of the approximated concentration in equation (2), $\mathbf{b}(t)$ is also a vector of the form:

$$\mathbf{b}(t) = \mathbf{b}^0 + \mathbf{b}^1 t + \mathbf{b}^2 t^2 - \mathbf{b}^3 t^3. \quad (13)$$

[16] See in the Appendix A the relationship between $\mathbf{A}^e$, $\mathbf{b}$ and ξ_2^{e-1}, u^e, Δx. The solution of system (12) at the end of the time step is given by:

$$\mathbf{c}^e - \alpha_1\mathbf{c}_1^e - \alpha_2\mathbf{c}_2^e + \mathbf{v}^0 + \mathbf{v}^1\Delta t + \mathbf{v}^2\Delta t^2 + \mathbf{v}^3\Delta t^3. \quad (14)$$

where α_1 and α_2 are two arbitrary coefficients, that have to be chosen according to the initial concentration node values, $\mathbf{c}_1^e$ and $\mathbf{c}_2^e$ are the two solutions of the homogeneous equations associated to system (12) and $\mathbf{v}^0$, $\mathbf{v}^1$, $\mathbf{v}^2$, $\mathbf{v}^3$ are four vectors that can be computed after substitution of the right-hand side of equation (14) in equation (12). Matrix $\mathbf{A}^e$
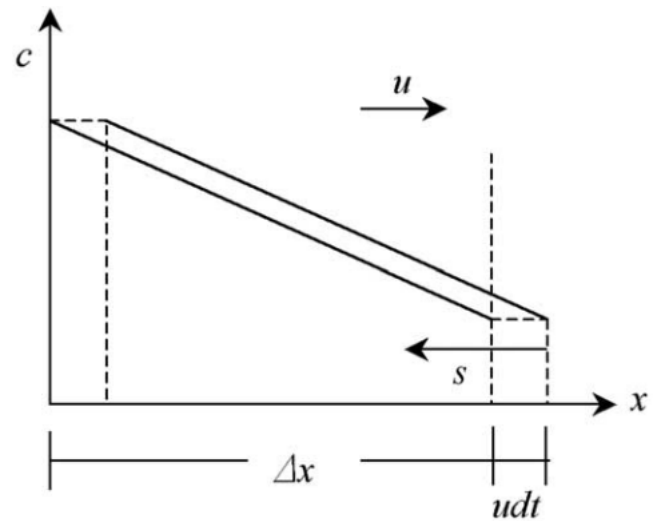

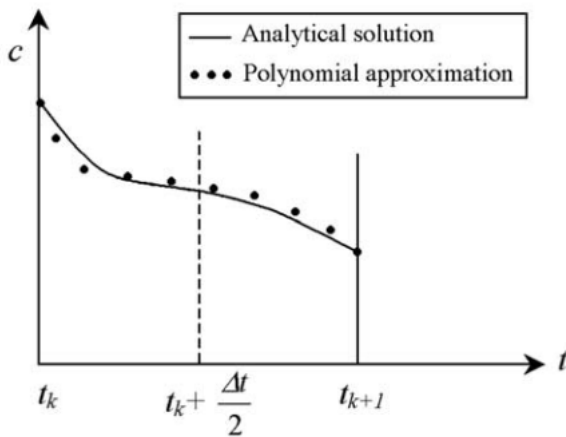**Figure 2.** Evaluation of the spatial moment of c in the 1-D case.

Figure 3. Time cubic approximation of the leaving flux.

can have either two real or two complex conjugate eigenvalues. In the first case call λ_1^e, λ_2^e, and $\mathbf{u}_1^e$, $\mathbf{u}_2^e$ the corresponding eigenvalues and eigenvectors; in the second case, call λ_r^e, λ_i^e, and $\mathbf{u}_r^e$, $\mathbf{u}_i^e$ the real and the imaginary part, respectively, of the conjugate eigenvalues and of the corresponding eigenvectors. The solutions of the homogeneous equations at the end of the time step are, respectively:

$$\mathbf{c}_1^e = \mathbf{u}_1^e e^{\lambda_1^e \Delta t} \qquad \mathbf{c}_2^e = \mathbf{u}_2^e e^{\lambda_2^e \Delta t} \tag{15a}$$

or

$$\mathbf{c}_1^e = e^{\lambda_r^e \Delta t}\left[\mathbf{u}_r^e \cos\left(\lambda_i^e \Delta t\right) \quad \mathbf{u}_i^e \sin\left(\lambda_i^e \Delta t\right)\right]$$
$$\mathbf{c}_2^e = e^{\lambda_r^e \Delta t}\left[\mathbf{u}_r^e \sin\left(\lambda_i^e \Delta t\right) \quad \mathbf{u}_i^e \cos\left(\lambda_i^e \Delta t\right)\right]. \tag{15b}$$

[17] Observe that matrix $\mathbf{A}^e$ is not a function of the concentration, therefore it can be computed and factorized only once for each velocity field distribution. This also holds for the homogeneous solutions (15) that can be stored for each element before starting the time marching computations.

[18] Vectors $\mathbf{v}^0$, $\mathbf{v}^1$, $\mathbf{v}^2$ and $\mathbf{v}^3$ can be obtained by comparing the terms with the same time exponent in the polynomial part of equation (12). This leads to the sequential solution of the following linear systems:

$$\mathbf{A}^e \mathbf{v}^3 = -\mathbf{b}^3 \tag{16a}$$

$$\mathbf{A}^e \mathbf{v}^2 = -\mathbf{b}^2 - 3\mathbf{v}^3 \tag{16b}$$

$$\mathbf{A}^e \mathbf{v}^1 = -\mathbf{b}^1 - 2\mathbf{v}^2 \tag{16c}$$

$$\mathbf{A}^e \mathbf{v}^0 = \mathbf{b}^0 \quad \mathbf{v}^1. \tag{16d}$$

[19] As already pointed out, coefficients α_1, α_2 in equation (14) are computed by forcing the solution to honor the given initial concentration values $\mathbf{c}_0^e$. This leads,
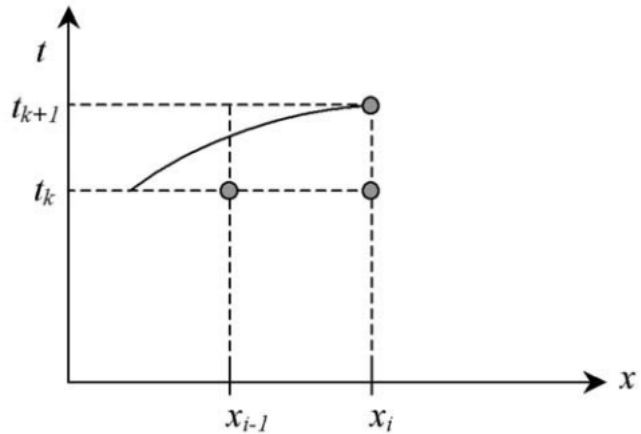

Figure 4a. Characteristic line through the point (x_i, t_{k+1}).

according to the type of homogeneous solution, to the system:

$$\mathbf{c}_0^e = \alpha_1 \mathbf{u}_1^e \quad \alpha_2 \mathbf{u}_2^e \quad \mathbf{v}^0, \tag{17a}$$

or

$$\mathbf{c}_0^e = \alpha_1 \mathbf{u}_r^e + \alpha_2 \mathbf{u}_i^e + \mathbf{v}^0. \tag{17b}$$

Once the ODEs are solved, the concentration at the nodes of the element can be approximated by the cubic polynomials ξ^e by maintaining the initial and the final values, as well as the mean (in time) value of the concentration and of their first-order moments (Figure 3). The exact conservation of the mean concentration $\overline{\xi}_{ij}^e$ at each element node guarantees the mass balance in the element and the global mass conservation; $\overline{\xi}_{ij}^e$ can be obtained by time analytical integration of the solution of equation (12). The mean of the first-order moments can be more easily estimated by numerical integration. See in Appendix B the equations to be solved for the estimation of the coefficients of polynomial (2).

[20] According to the second-order approximation of the concentrations, negative concentration fluxes leaving the
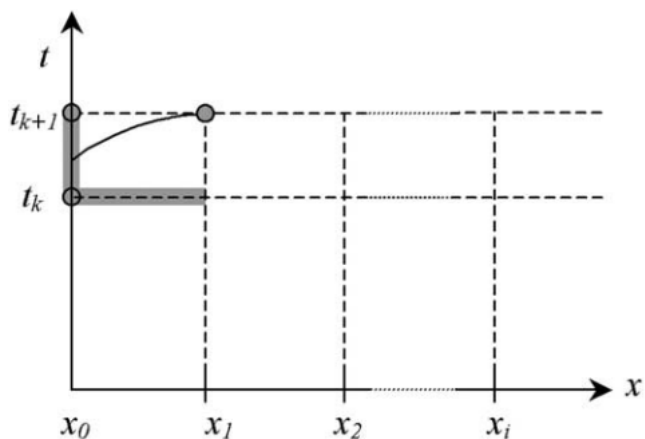

Figure 4b. Possible foot location of the characteristic line (thick segments).
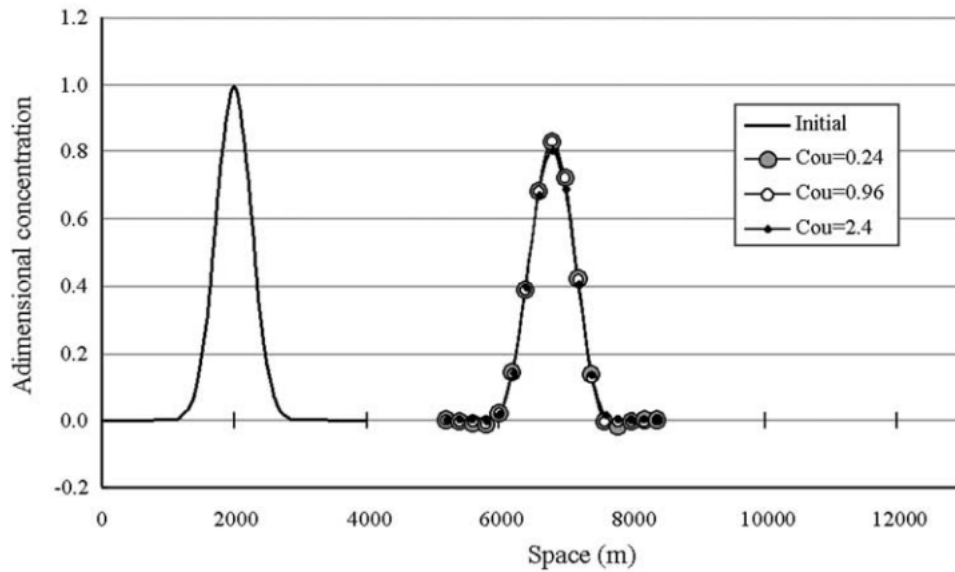
Figure 5. Gaussian concentration test for the 1-D algorithm.

element can be computed while solving equation (12). Negative flux correction can be simply carried out according to the following procedure: after estimation of the polynomial coefficients of the approximating concentration, evaluate the minimum value within the range $0 \leq t \leq \Delta t$ and the corresponding time t_{min}. If the minimum concentration $\xi_j^e(t_{min})$ is negative, compute the root α of the following equation:

$$\min\left(0, \overline{\xi_j^e}\right) = (1 - \alpha)\xi_j^e(t_{min}) + \alpha\overline{\xi_j^e} \qquad (18)$$

and set

$$\xi_j^{e,0} \leftarrow (1 - \alpha)\xi_j^{e,0} + \alpha\overline{\xi_j^e}; \quad \xi_j^{e,i} \leftarrow (1 - \alpha)\xi_j^{e,i} \quad i = 1, 2, 3. \qquad (19)$$

The corrected polynomial concentrations do not alter the mean value. They have only positive values if the mean value is greater than zero. If the mean value is negative the corrected polynomial is a negative constant value. More complex procedures are required to always guarantee positive fluxes. After computation of the unknown coefficients $\xi^{e,0}$, $\xi^{e,1}$, $\xi^{e,2}$, $\xi^{e,3}$ is complete, a new system of ODEs can be solved for the downstream element with index $e + 1$.

[21] Consistency and unconditional stability of the algorithm have been proved in the 1-D case using Fourier analysis, assuming a first-order approximation of the concentration inside the element, a constant in time leaving flux and computing the analytical solution of the resulting ODE along the given time step [*Tucciarelli and Fedele*, 2000].

[22] A useful insight about the question, in the case of second-order approximation of the concentration, is given by the observation of the characteristic line passing through any x_i, t_{k+1} point of the computational domain. Observe in Figure 4a that for large Courant numbers the foot of the characteristic line, at time t_k, falls very far from the original point. The numerical estimate of concentration c at point x_i, t_{k+1} is possible if an initial value is known for the ODE associated to the characteristic line; therefore it is impossible to evaluate the concentration in explicit form only as a function of the concentrations at the points x_{i-1}, t_k and x_i, t_k. This is possible only for the point x_1, t_{k-1}, because in this case the foot of the characteristic line is located along the time axis and the boundary value is known (Figure 4b). The basic idea of the proposed algorithm is to estimate, along with the unknown value at point x_1, t_{k-1}, also a low-order time approximation of the concentration c at the same distance, from time t_k to time t_{k-1}. After this, it is possible to solve the problem at point x_2, t_{k+1} using the computed time approximation as initial value for the new ODE problem, and so on for all the unknown concentrations at time level t_{k+1}. The algorithm stated in section 1 applies this idea in integral form, to guarantee mass conservation. The

Table 1. Convergence Numerical Test for the 1-D Advection of a Gaussian Concentration Wave

Number of Elements	Step Length, m	Cou = 0.24		Cou = 0.96		Cou = 2.4	
		Average Error	Order	Average Error	Order	Average Error	Order
128	100	1.78E-03[a]	2.76	1.78E-03	2.76	1.88E-03	2.81
256	50	2.63E-04	2.62	2.62E-04	2.62	2.69E-04	2.65
512	25	4.28E-05	—	4.28E-05	—	4.29E-05	—

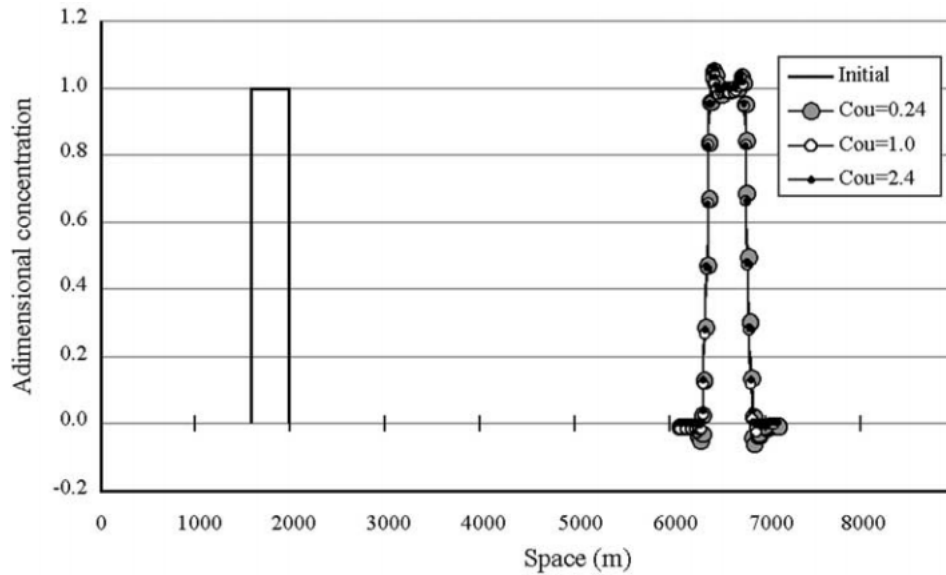[a]Read 1.78E-03 as 1.78×10^{-3}.

Figure 6. Unitary square concentration test for the 1-D algorithm.

use of low-order spatial and time approximations suppresses the propagation of the high-frequency error components of the estimated fluxes from one computational element to the others.

[23] The algorithm is tested first for the following literature case [*Yeh*, 1990]: given the initial condition

$$c(x, 0) = \exp\left(-\frac{(x - x_0)^2}{2\sigma_0^2}\right). \qquad (20)$$

$u = 0.5$ m/s, $\Delta x = 200$ m, $\Delta t = 96$ s, $\sigma_0 = 264$ m, $x_0 = 2000$ m and a mesh with 65 elements, compute the concentration distribution after 100 time steps.

[24] The Courant number of the example is 0.24. Initial condition (20) has been assigned to all the element nodes; the final concentrations shown in Figure 5 are the average concentrations computed at the node shared by two connected elements. Observe in Figure 5 the test results obtained using Courant numbers equal to 0.96 and 2.4. With the two Courant numbers smaller than one, numerical diffusion is almost the same and produces a peak reduction from 1.0 to 0.85. With the Courant number greater than one, the peak is reduced from 1.0 to 0.84, but the shape of the concentration distribution remains almost the same as in the previous cases. Also, no instabilities occur. *Yeh* [1990] reports a peak reduction

from 1.0 to 0.25 for the test results obtained using the Petrov-Galerkin method and a Courant number equal to 0.24.

[25] A numerical convergence test has been carried out using the same example and a mesh density ranging from 32 to 1024 elements. The rates of convergence have been inferred from the values of the average error, defined as:

$$\overline{E} = \frac{1}{N} \sum_{e=1}^{N} \left(\frac{1}{n} \sum_{j=1}^{n} \left| c_j^{e\ computed} - c_j^{e\ exact} \right| \right), \qquad (21)$$

where n is the number of the nodes of each element ($n = 2$ in the 1-D case, $n = 3$ in the 2-D case). A power dependency between the average error and the grid size has been considered, and the exponent has been assumed as the convergence order. Table 1 shows the order of convergence, for different Courant numbers, computed from one mesh density to the next one. Observe that the order of convergence in this case is greater than two.

[26] A second test has been carried out for the same domain and the same velocity field, using as initial concentration condition a unitary square wave whose width is 400 m. The center of the square wave is initially located at 1800 m from the origin of the domain. The final concentrations after a period T = 9600 s, obtained for different values of the Courant number and $\Delta x = 12.5$ m, are shown in Figure 6. The shape of the solution is almost the same for

Table 2. Convergence Numerical Test for the 1-D Advection of a Block-Shaped Concentration Wave

Number of Elements	Step Length, m	Cou = 0.24		Cou = 1.0		Cou = 2.4	
		Average Error	Order	Average Error	Order	Average Error	Order
256	50	1.01E-02	0.77	9.37E-03	0.74	9.65E-03	0.80
512	25	5.92E-03	0.74	5.63E-03	0.76	5.56E-03	0.73
1024	12.5	3.55E-03	—	3.33E-03	—	3.35E-03	—
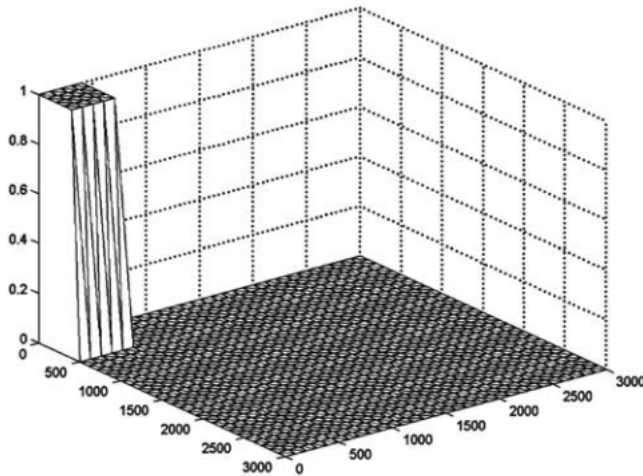
Figure 7. Initial concentration for the 2-D first test case.

Figure 8b. Test 1: Computational mesh with the hypotenuse parallel to the flow direction.

the three different values of the Courant number, approximating quite well a square waveform even when the Courant number is larger than one. The numerical dispersion causes the smoothing of the vertical fronts of the concentration pulse and local maxima and minima arise around the sharp fronts, but again no instabilities are present. The maximum value is located at the beginning of the unitary wave, and its value is 1.063 for Cou = 0.24, 1.064 for Cou = 1.0 and 1.056 for Cou = 2.4. The minimum value is located at the end of the square wave and increases for increasing values of the Courant number; it is equal to −0.049, −0.007 and −0.002, respectively for Cou = 0.24, Cou = 1.0 and Cou = 2.4. The convergence rate has been evaluated, obtaining the values shown in Table 2. Because of the difficulties in reproducing the sharp fronts, the order of convergence is much smaller than in the previous test case.

3. Extension of the Algorithm to the 2-D Case

[27] An important requirement for the application of the algorithm is the possibility of sorting the elements in order to know the time approximation of the entering fluxes before 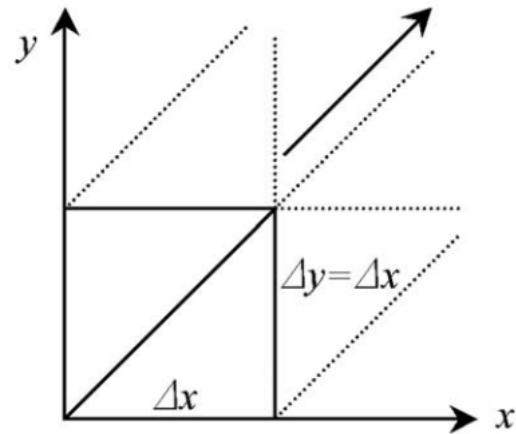each system of ODEs is solved. This is always possible, in 2-D and 3-D cases, if a scalar potential exists such that the direction of its gradient is opposite to the flow direction. In this case, if the elements are ordered from the highest to the lowest potential value, the entering fluxes are always known from the previous solution of the elements with higher potential. For groundwater transport problem, the scalar potential is given by the piezometric head, defined as usual as the sum of the topographical elevation and the pressure height [de Marsily, 1986].

[28] Assume the use of a triangular mesh for the flow field computation. If the finite element method is used and a constant velocity is estimated inside each element, the water fluxes through the common side of two adjacent elements computed using the two element velocities are not necessarily the same. To fix this inconsistency, each element of the original mesh can be divided in four subelements and the velocities can be changed in the subelements surrounding each node to preserve the flux continuity [Kinzelback and Cordes, 1992]. If the finite volume method is used, three fluxes are computed at the sides of each element, where a velocity vector has to be estimated. In the steady state case this can be done by solving a linear system where the fluxes of the u, v components of the velocity are set
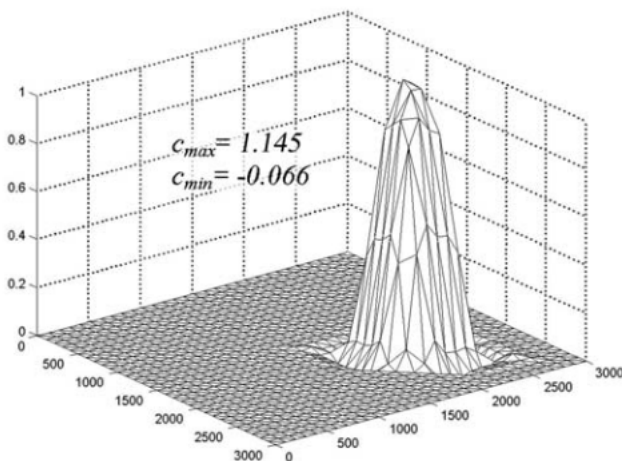

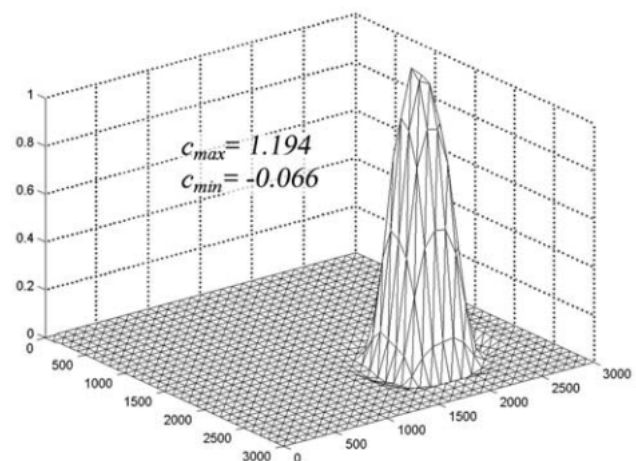**Figure 8a.** Test 1: Solution after $T = 18000$s (mesh 1).

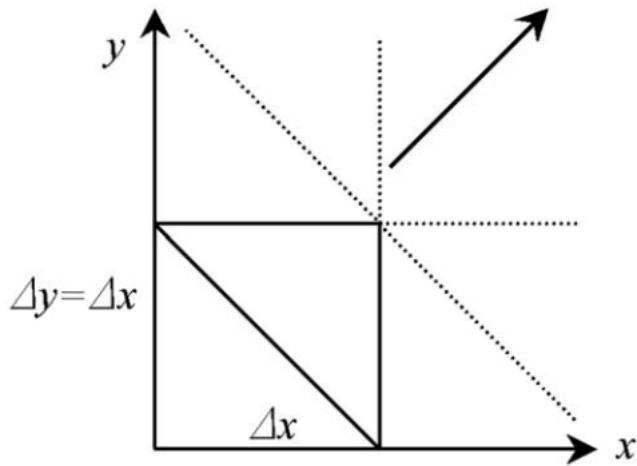**Figure 9a.** Test 1: Solution after $T = 18000$ s (mesh 2).

Figure 9b. Test 1: Computational mesh with the hypotenuse orthogonal to the flow direction.

equal to the computed ones along two of the three sides of the element. According to the steady state assumption, also the third flux of the velocity vector will be equal to the computed one.

[29] The most severe limitation of the finite element method is that the piezometric head changes linearly inside each element; the cell where the mass balance is enforced is a polygon defined by the centers of all the triangles surrounding each node. In 2-D problems, fluxes are always directed from the cells of the nodes with higher to the cells of the nodes with lower piezometric head only if obtuse triangles are missing. Moreover, analytical integration of the concentration spatial moment inside the polygon is awkward. In the finite volume method, the mass balance cell is the triangle itself and the flux is always directed from the element with higher to the element with lower piezometric head. In the following, we assume the flow field to be known from the solution of a finite volume problem and from the subsequent estimation of a single velocity vector inside each element.
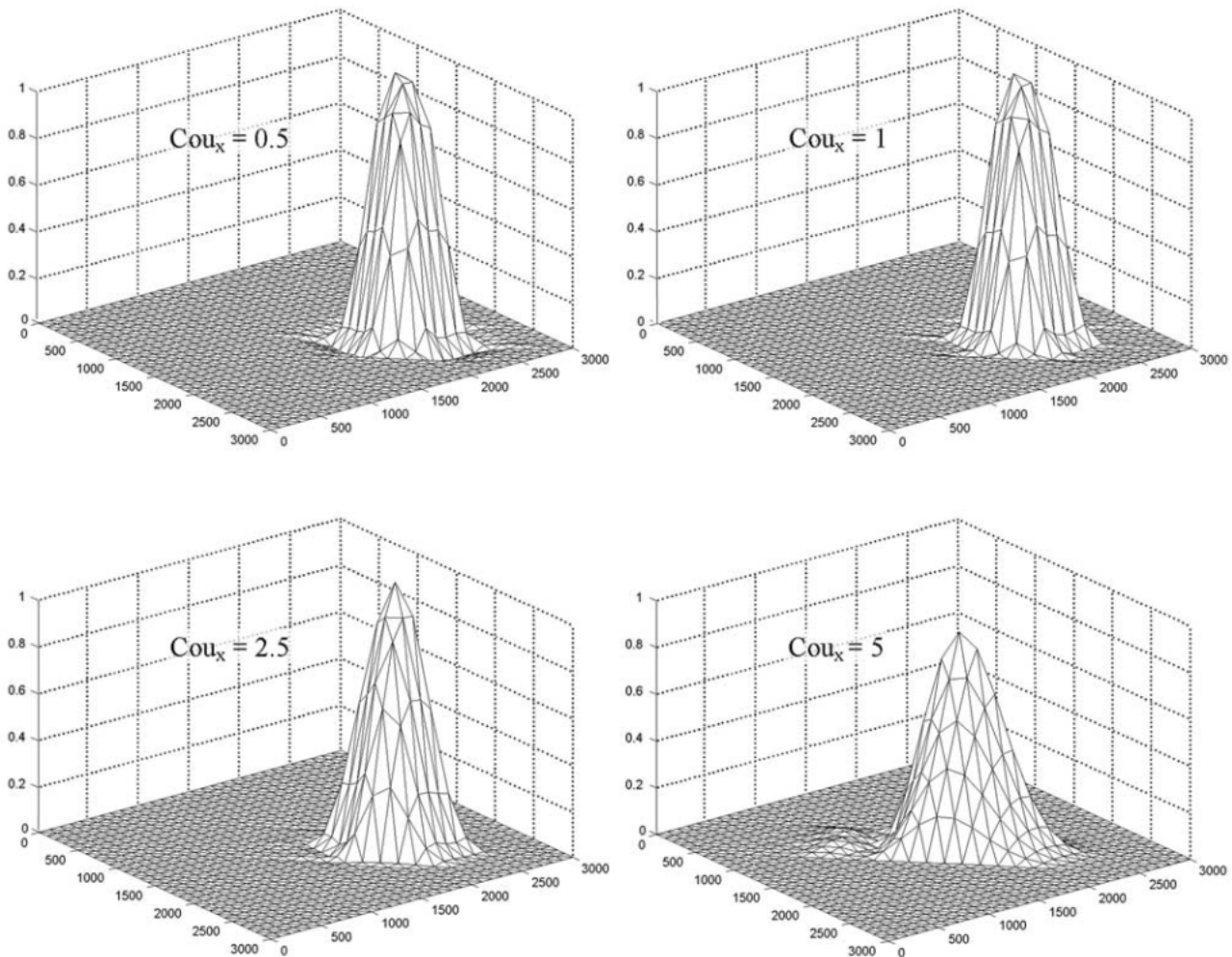

Figure 10. Test 1: Solution after $T = 18000$ s for different Courant numbers and mesh 1.
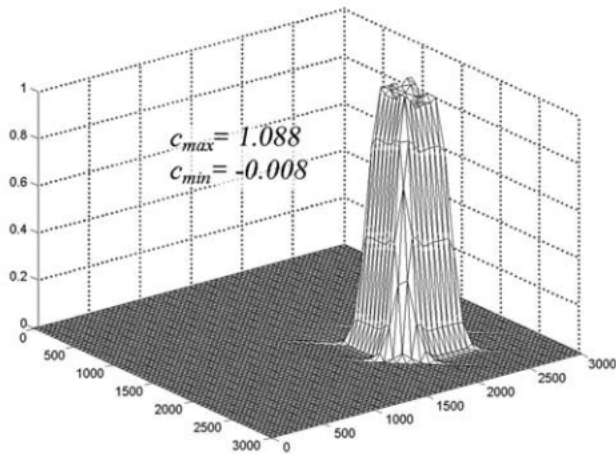
Figure 11. Test 1: Solution after $T = 18000$ s and mesh 1 ($\Delta x = 50$ m).

[30] Given the following 2-D advection problem:

$$\frac{\partial c}{\partial t} - u\frac{\partial c}{\partial x} + v\frac{\partial c}{\partial y} = 0, \tag{22}$$

where u and v represent the x and y velocity components, call $I_1, \ldots, I_N$ the sequence of triangular elements ordered according to the corresponding piezometric head. Assume the following piecewise linear approximation of the concentration inside each element:

$$c^e = \sum_{i=1,3} N_i^e c_i^e, \tag{23}$$

where N_i^e is the Galerkin shape function of x and y coordinates and e is the index of the element where x and y are located. Mass conservation can be guaranteed by the following ODE:

$$\frac{d}{dt}\left(\frac{c_1^e + c_2^e + c_3^e}{3}\right) = \frac{Fe^e(t) - Fl^e(t)}{\sigma^e}. \tag{24}$$

where σ^e is the area of element e, Fe^e and Fl^e are, respectively, the entering and leaving concentration fluxes. The x and y moments can be easily estimated as functions of the x and y coordinates of the element nodes, i.e.,

$$\overline{Mx^e} = \int_{\sigma^e} \sum_{i=1,3} N_i^e c_i^e \sum_{i=1,3} N_i^e x_i^e = \sum_{i=1,3} c_i^e \sum_{j=1,3} x_j^e \alpha_{ij}$$

$$\overline{My^e} = \sum_{i=1,3} c_i^e \sum_{j=1,3} y_j^e \alpha_{ij}, \tag{25}$$

$$\alpha_{ij} = \frac{\sigma^e}{6} \quad \text{if } i = j, \quad \alpha_{ij} = \frac{\sigma^e}{12} \quad \text{if } i \neq j \tag{26}$$
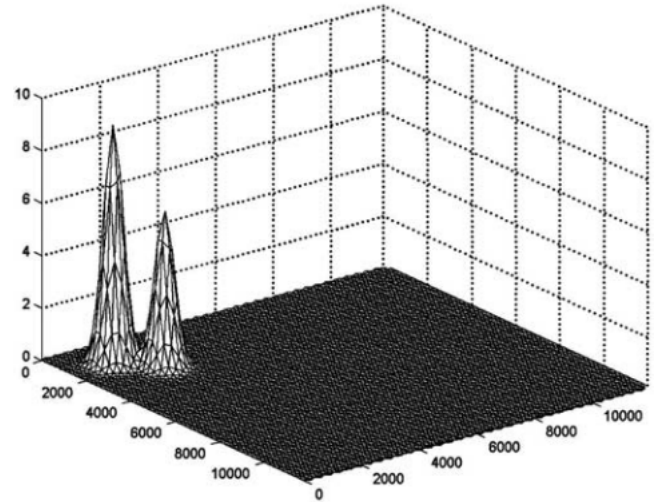

Figure 12. Initital concentration for the 2-D second test case.

[*Huyakorn and Pinder*, 1983]. The same procedure used in the 1-D case can be applied here in order to obtain a time derivative of the first-order moments (25) equal to the derivative obtained from the solution of equation (22). The element moments, in the x and y direction at time $t - dt$, are equal to:

$$Mx^e(t - dt) = \int_{\sigma^e} c^e[(x + u^e dt)]d\sigma + \sum_{i=1,3}\delta_i \int_{L_i^e} \phi_i^e \xi_m(x,y,t)x\,dL_i$$

$$- \sum_{i=1,3}(1 - \delta_i)\int_{L_i^e} \phi_i^e c^e x\,dL_i; \tag{27a}$$

$$My^e(t - dt) = \int_{\sigma^e} c^e[(y + v^e dt)]d\sigma + \sum_{i=1,3}\delta_i \int_{L_i^e} \phi_i^e \xi_m(x,y,t)y\,dL_i$$

$$\sum_{i=1,3}(1 - \delta_i)\int_{L_i^e} \phi_i^e c^e y\,dL_i; \tag{27b}$$

where u^e and v^e are the components of the velocity inside the element in the x and y direction, δ_i is equal to 1 or 0 if the flux is, respectively, entering or leaving the element, L_i^e is the length of the ith side of element e, that is the side following the ith node in counterclockwise direction, ϕ_i^e is the mass flux (positive if leaving the element) per unit length through the same side of the element and ξ_m is the approximated concentration of the entering flux. In the linear integral, c^e represents the concentration at points of the ith side of element e. Linear integrals in equation (27) represent the moment fluxes and can be easily estimated

Table 3. Two-Dimensional Test 1: Convergence Numerical Test

Number of Elements	Step Length, m	$\text{Cou}_x = 0.15$		$\text{Cou}_x = 1.0$		$\text{Cou}_x = 2.5$	
		Average Error	Order	Average Error	Order	Average Error	Order
2×24^2	125	1.99E-02	0.77	1.98E-02	0.76	2.01E-02	0.70
2×48^2	62.5	1.17E-02	0.77	1.17E-02	0.77	1.24E-02	0.75
2×96^2	31.25	6.88E-03	—	6.83E-03	—	7.34E-03	—

Table 4. Parameter Values for Test 2

|	x, m	y, m	C	s, m
Peak 1	1400	1400	10.0	264
Peak 2	2400	2400	6.5	264

assuming c^e as a function of the concentrations at the nodes of the ith side of element e and ξ_m as a function of the concentrations at the nodes of the corresponding side of the upstream element. This provides, for the x and y spatial moment fluxes:

$$\int_{L_i^e} \circ_i^e c^e x \, dL_i = \int_{L_i^e} \circ_i^e \left(c_j^e \frac{s}{L_i^e} + c_i^e \frac{(L_i^e - s)}{L_i^e} \right) \left(x_j^e \frac{s}{L_i^e} + x_i^e \frac{(L_i^e - s)}{L_i^e} \right) ds$$
$$- \phi_i^e L_i^e \left[\frac{\left(c_j^e - c_i^e \right)\left(x_j^e - x_i^e \right)}{3} + \frac{c_j^e x_i^e - c_i^e x_j^e}{2} \right], \quad (28a)$$

$$\int_{L_i^e} \circ_i^e c^e y \, dL_i = \int_{L_i^e} \circ_i^e \left(c_j^e \frac{s}{L_i^e} + c_i^e \frac{(L_i^e - s)}{L_i^e} \right) \left(y_j^e \frac{s}{L_i^e} + y_i^e \frac{(L_i^e - s)}{L_i^e} \right) ds$$
$$- \phi_i^e L_i^e \left[\frac{\left(c_j^e - c_i^e \right)\left(y_j^e - y_i^e \right)}{3} + \frac{c_j^e y_i^e - c_i^e y_j^e}{2} \right] \quad (28b)$$

where j is the index of the node following the ith node in counterclockwise direction. Subtracting moments (25) from (27) and dividing by dt, you get:

$$\frac{d\overline{Mx^e}}{dt} = \sum_{i=1,3} u^e c_i^e \int_{\sigma^e} N_i^e d\sigma + \sum_{i=1,3} \delta_i \int_{L_i^e} \circ_i^e \xi_m(x,y,t) x \, dL_i$$
$$- \sum_{i=1,3} (1 - \delta_i) \int_{L_i^e} \phi_i^e c^e x \, dL_i \quad (29a)$$
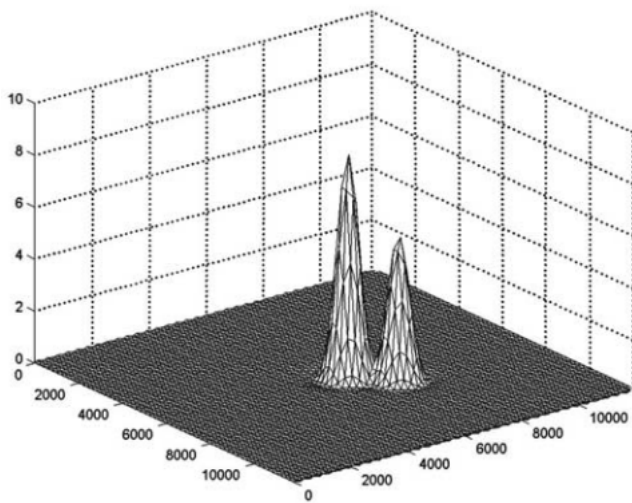
Figure 13a. Test 2: Solution after $T = 9600$ s.
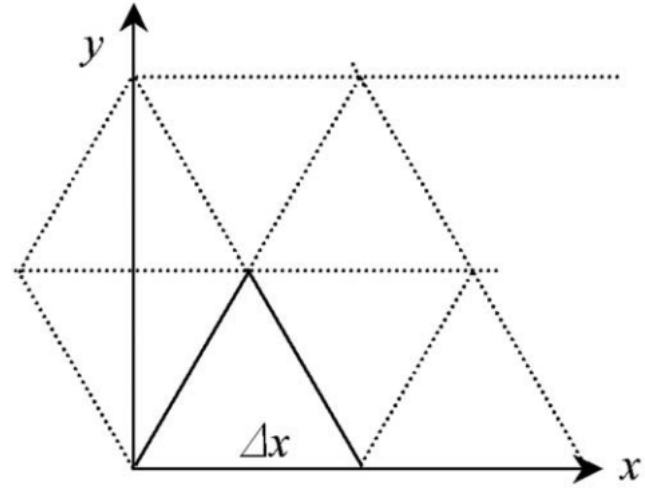
Figure 13b. Test 2: Computational mesh.

$$\frac{d\overline{My^e}}{dt} = \sum_{i=1,3} v^e c_i^e \int_{\sigma^e} N_i^e d\sigma - \sum_{i=1,3} \delta_i \int_{L_i^e} \circ_i^e \xi_m(x,y,t) y \, dL_i$$
$$- \sum_{i=1,3} (1 - \delta_i) \int_{L_i^e} \circ_i^e c^e y \, dL_i. \quad (29b)$$

[31] The average in space concentration $\bar{c}^e$ and the moments given by equation (25) can be written in matrix form as:

$$\begin{pmatrix} \bar{c}^e \\ \overline{Mx^e} \\ \overline{My^e} \end{pmatrix} = \begin{pmatrix} B_{11}^e & B_{12}^e & B_{13}^e \\ B_{21}^e & B_{22}^e & B_{23}^e \\ B_{31}^e & B_{32}^e & B_{33}^e \end{pmatrix} \begin{pmatrix} c_1^e \\ c_2^e \\ c_3^e \end{pmatrix} \quad (30)$$

$$B_{11}^e = B_{12}^e = B_{13}^e = \frac{1}{3}; \quad B_{21}^e = \frac{\sigma^e}{12}\left(2x_1^e - x_2^e + x_3^e\right);$$
$$B_{22}^e = \frac{\sigma^e}{12}\left(x_1^e + 2x_2^e - x_3^e\right); \quad B_{23}^e = \frac{\sigma^e}{12}\left(x_1^e + x_2^e + 2x_3^e\right)$$
$$B_{31}^e = \frac{\sigma^e}{12}\left(2y_1^e + y_2^e + y_3^e\right); \quad B_{32}^e = \frac{\sigma^e}{12}\left(y_1^e + 2y_2^e - y_3^e\right);$$
$$B_{33}^e = \frac{\sigma^e}{12}\left(y_1^e - y_2^e + 2y_3^e\right). \quad (31)$$

Table 5. Test 2: Comparison Between the Proposed Procedure and Other Numerical Schemes[a]

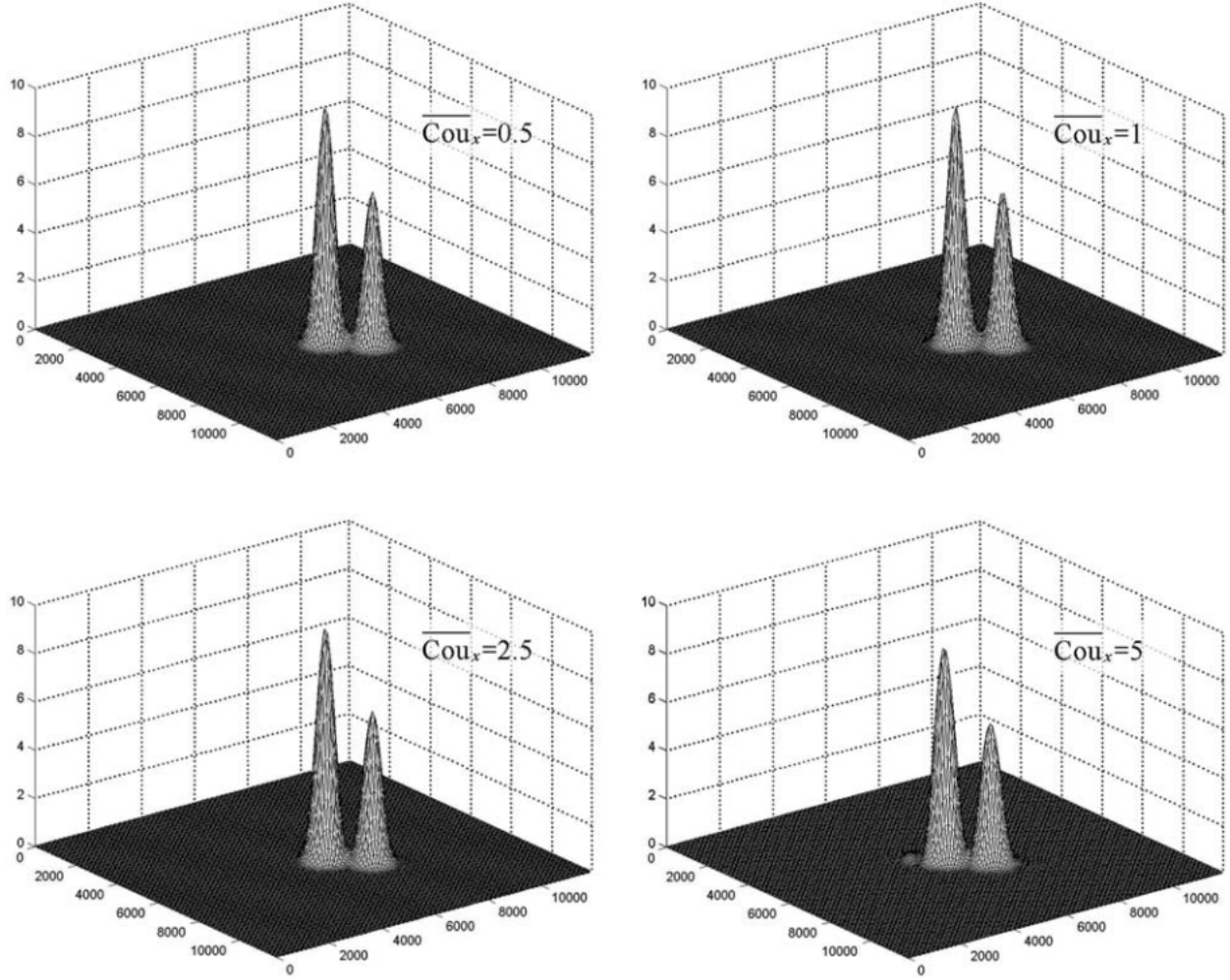| Scheme	Max c	Min c	RMS
Proposed procedure	8.53	−0.059	0.0475
Exact solution	10.	0.	0.
First-order upwind scheme	1.05	0.	0.4109
Leapfrog scheme	5.32	−2.34	0.4723
Lax-Wendroff scheme	4.78	−1.83	0.4027
Six-point scheme	8.52	0.339	0.0594
SOWMAC	8.66	0.231	0.0571
Proposed procedure ($\Delta x/2$)	9.52	−0.0001	0.0146

[a]$\Delta x = 200$ m.

Figure 14. Test 2: Solution after $T = 9600$ s for different average Courant numbers ($\Delta x = 100$ m).

[32] Differentiating equation (30), multiplying both sides by the inverse of the square matrix $\mathbf{B}^e$ and substituting the moment derivatives with the right hand side of equations (24) and (29), the system of ODEs can be written in normal form as:

$$
\begin{pmatrix} \dfrac{dc_1^e}{dt} \\[2mm] \dfrac{dc_2^e}{dt} \\[2mm] \dfrac{dc_3^e}{dt} \end{pmatrix} = \begin{pmatrix} B_{11}^e & B_{12}^e & B_{13}^e \\ B_{21}^e & B_{22}^e & B_{23}^e \\ B_{31}^e & B_{32}^e & B_{33}^e \end{pmatrix}^{-1}
$$
$$
\cdot \begin{pmatrix} \displaystyle\sum_{i=1,3}\frac{\delta_i}{\sigma^e}\int_{L_i^e}\phi_i^e\xi_m dL_i - \sum_{i=1,3}\frac{1-\delta_i}{\sigma^e}\int_{L_i^e}o_i^e c^e dL_i \\[3mm] c^e\sigma^e u^e - \displaystyle\sum_{i=1,3}\delta_i\int_{L_i^e}o_i^e\xi_m x\,dL_i - \sum_{i=1,3}(1-\delta_i)\int_{L_i^e}o_i^e c^e x\,dL_i \\[3mm] \bar c^e\sigma^e v^e + \displaystyle\sum_{i=1,3}\delta_i\int_{L_i^e}\phi_i^e\xi_m y\,dL_i - \sum_{i=1,3}(1-\delta_i)\int_{L_i^e}\phi_i^e c^e y\,dL_i \end{pmatrix},
$$

(32)

that can also be expressed using the same matrix notation of equations (12) and (13) (see Appendix A for the matrix $\mathbf{A}^e$ and vector $\mathbf{b}$ coefficient expressions).

[33] Owing to asymmetry, matrix $\mathbf{A}^e$ can have either three real or one real and two conjugate eigenvalues and eigenvectors. In the first case call λ_1^e, λ_2^e, λ_3^e and $\mathbf{u}_1^e$, $\mathbf{u}_2^e$, $\mathbf{u}_3^e$ the real eigenvalues and eigenvectors, in the second case call λ_3^e and $\mathbf{u}_3^e$ the real eigenvalue and eigenvector, λ_r^e, $\mathbf{u}_r^e$ and λ_i^e, $\mathbf{u}_i^e$

Table 6. Test 2: Maximum and Minimum Concentrations and Root Mean Square Error for Different Average Courant Numbers[a]

$\overline{Cou}_x = \overline{Cou}_y$	Max c	Min c	RMS
0.25	9.61	1.53E-03	0.0076
0.5	9.70	1.24E-03	0.0076
1	9.76	−9.32E-04	0.0075
2.5	9.67	−1.33E-04	0.0098
5	8.73	−1.65E-02	0.0442
10	5.29	−5.82E-03	0.2216
20	3.72	−2.34E-04	0.3026

[a]$\Delta x = 100$ m.

Table 7. Two-Dimensional Test 2: Convergence Numerical Test

Number of Elements	Step Length, m	$\overline{\mathrm{Cou}}_x - 0.15$		$\overline{\mathrm{Cou}}_x - 1.0$		$\overline{\mathrm{Cou}}_x - 2.5$	
		Average Error	Order	Average Error	Order	Average Error	Order
5432	250	1.31E-02	2.33	1.21E-02	2.22	1.48E-02	2.24
21728	125	2.60E-03	2.69	2.59E-03	2.68	3.11E-03	2.84
86912	62.5	4.03E-04		4.03E-04		4.36E-04	

the real and the imaginary part of the two conjugate eigenvalues and eigenvectors. The solution of system (12) at $t = \Delta t$ is given by:

$$\mathbf{c}^e - \alpha_1 \mathbf{c}_1^e + \alpha_2 \mathbf{c}_2^e + \alpha_3 \mathbf{c}_3^e + \mathbf{v}^0 - \mathbf{v}^1 \Delta t + \mathbf{v}^2 \Delta t^2 - \mathbf{v}^3 \Delta t^3, \quad (33)$$

where $\mathbf{c}_1^e$ and $\mathbf{c}_2^e$ are given by equations (15) and

$$\mathbf{c}_3^e - e^{\lambda_3^e \Delta t} \mathbf{u}_3^e. \quad (34)$$

[34] The first three terms of equation (33) represent the solution of the homogeneous part of equation (12). Vectors $\mathbf{v}^0$, $\mathbf{v}^1$, $\mathbf{v}^2$ and $\mathbf{v}^3$ have to be estimated by substitution of solution (33) in equations (12), according to the sequence of equation (16). Coefficients α_1, α_2 and α_3 are computed by forcing the solution of equation (12) to honor the initial concentration values at the three nodes of the element, according to the system:

$$\begin{aligned}
\mathbf{c}_0^e &= \alpha_1 \mathbf{u}_1^e + \alpha_2 \mathbf{u}_2^e - \alpha_3 \mathbf{u}_3^e + \mathbf{v}^0 \quad \text{or} \\
\mathbf{c}_0^e &= \alpha_1 \mathbf{u}_r^e + \alpha_2 \mathbf{u}_i^e + \alpha_3 \mathbf{u}_3^e - \mathbf{v}^0.
\end{aligned} \quad (35)$$

[35] The mean (in time) concentrations at the three nodes of the element can be computed after the solution of the system of ODEs (12), to obtain the polynomial coefficients shown in Appendix B. See also in Appendix C the pseudocode of the algorithm for the 2-D case.

[36] The 2-D algorithm is tested using two literature cases and a third one, in which the analytical solution is known. In the three examples the plotted final concentrations at nodes are evaluated as the average of the concentration values obtained at all the element nodes with the same x-y location.

[37] The first literature case [*Nassiri and Babarutsi*, 1997] is the uniform advection of a block-shaped adimensional concentration pulse, assuming a square space domain and a velocity forming a 45° angle with the domain axes. The block is initially located near the origin (see Figure 7) and its value is equal to 1.

[38] In Figures 8a, 8b, 9a, and 9b the results obtained after a time $T - 18000$ s, using two different meshes of isosceles right-angled triangles, are shown. The first and the second mesh have the hypotenuse, respectively, parallel and orthogonal to the flow direction. The length of the smaller element side is $\Delta x - 100$ m and the adopted time step is $\Delta t - 150$ s. The value of the velocity components is $u - v - 0.1$ m/s. The corresponding Courant numbers in the x and y direction are evaluated trough the following equations:

$$\mathrm{Cou}_x - \frac{u \Delta t}{\Delta x}, \qquad \mathrm{Cou}_y - \frac{v \Delta t}{\Delta y} \quad (36)$$

and are both equal to 0.15.

[39] The final concentration peak is overestimated, a small minimum arises and the sharp fronts of the block

are smoothed, as a consequence of the numerical dispersion, but no oscillations occur, regardless of the mesh orientation. Results are similar to those obtained using the Hermite scheme [*Holly and Preissmann*, 1977], in which the block shape is not preserved and the final value of the peak is overestimated by 20%, while the use of the oscillation-free method HLPA [*Zhu*, 1991] produces a peak underestimation of 27%. The same test is carried out by *Stefanovic and Stefan* [2001], comparing the result obtained using two semi-Lagrangian schemes based on cubic spline interpolation [*Branski and Holley*, 1986] and on cubic Hermite interpolation [*Holly and Preissmann*, 1977]. Both schemes produce overestimated final peaks (by 27% and 12%, respectively) and spurious minima. All the tests are carried out for Courant numbers less than 1.

[40] Observe that all the methods used for comparison are applied with the use of quadrilateral elements forming a regular mesh. Results obtained using the proposed procedure with increasing Courant numbers and the first triangular mesh are shown in Figure 10: observe that no instabilities occur also with Courant numbers greater than 1, even if numerical diffusion increases. Moreover, a strong improvement can be obtained by halving the length of the element sides and multiplying by four the size of the time step, leaving basically unchanged the total computational effort. See in Figure 11 the results obtained in this case with the proposed method using the first triangular mesh.

[41] We performed the numerical convergence test and obtained the values shown in Table 3 for different Courant numbers and the first triangular mesh. The average error has been evaluated through equation (21) using $n = 3$. The values obtained are smaller than one and similar to the values of the analogous 1-D case, due to the numerical
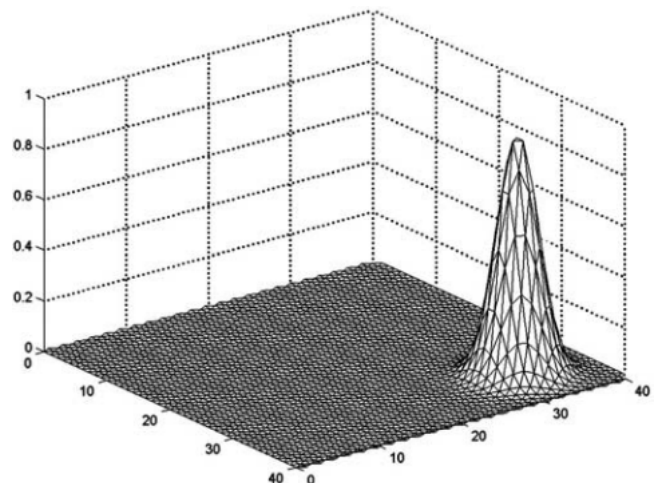
Figure 15. Initial concentration for the 2-D third test case.

Exact solution   ## Numerical solution

Iteration 10

Iteration 30

Iteration 50
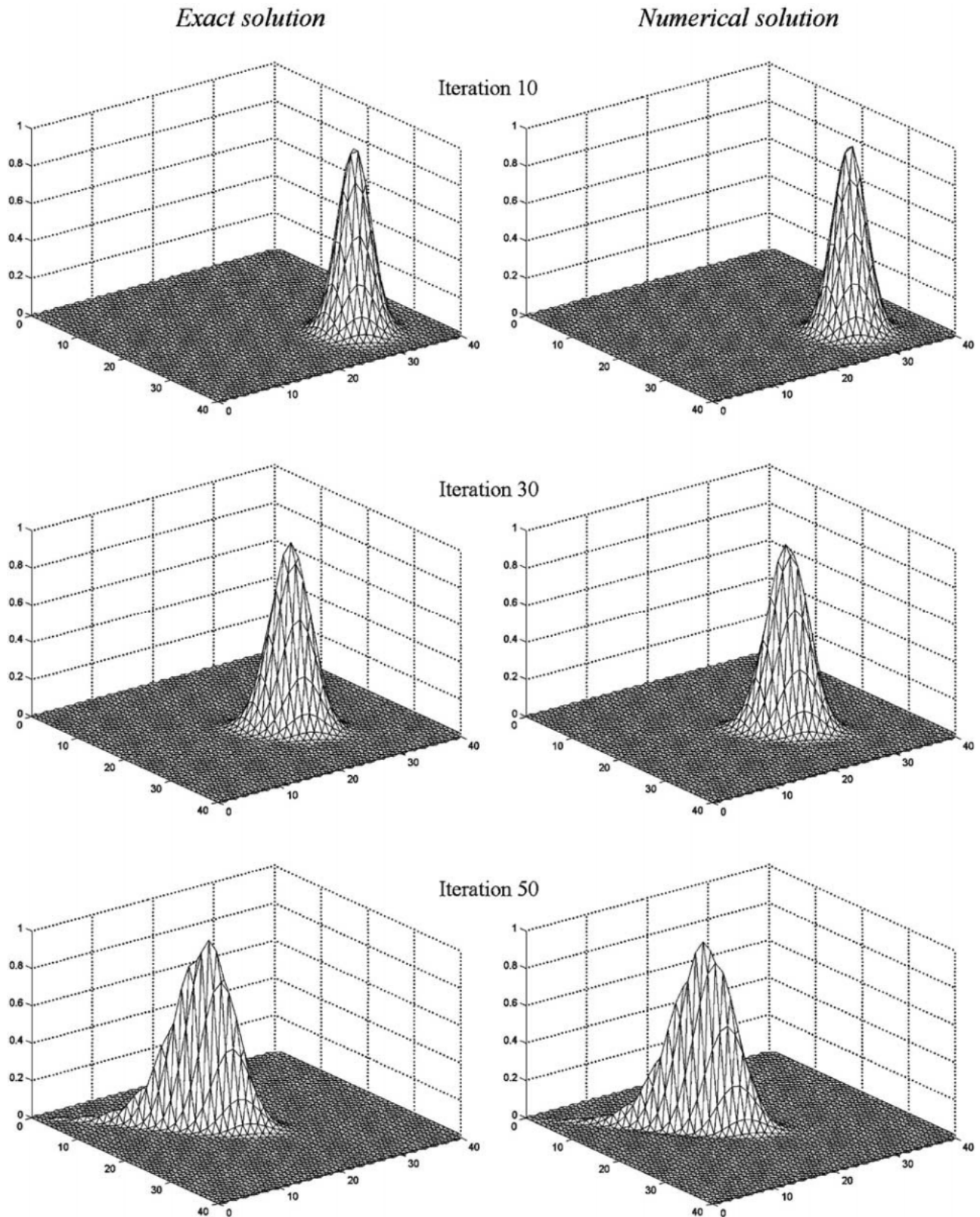
Figure 16. Test 3: Comparison between analytical and numerical results.

difficulties arising also in the 2-D case when trying to reproduce sharp fronts.

[42] The second test case [*Komatsu et al.*, 1997] is the uniform advection of a two-Gaussian peak adimensional

concentration on an indefinite two-dimensional domain (see Figure 12). Again, the velocity vector is oriented at 45° with respect to the axes, and its components are $u = v = 0.5$ m/s. The two-Gaussian initial concentration at point $P(x, y)$ is

Table 8. Test 3: Performance Indicators at Different Iterations

Number of Iterations	Max c	Min c	RMS
10	0.988	−0.0001	0.004
30	0.988	0.0002	0.008
50	0.985	0.0058	0.015

evaluated as:

$$c(P, 0) = C_1 \exp\left(-\frac{(\overline{PP_1})^2}{2\sigma_1^2} \right) + C_2 \exp\left(-\frac{(\overline{PP_2})^2}{2\sigma_2^2} \right) \qquad (37)$$

where P_1 and P_2 are the initial locations of the two peaks, C_1 and C_2 are the peak values and σ_1, σ_2 represent the standard deviations of the two distributions. The distribution parameters used in the test are outlined in Table 4.

[43] In Figure 13a the result obtained, after a time $T = 9600$ s, using a regular equilateral triangular mesh (see Figure 13b) with $\Delta x = 200$ m and $\Delta t = 100$ s are shown. The average Courant numbers are defined as:

$$\overline{Cou}_x = \frac{u\Delta t}{\sqrt{\sigma^e}} = \frac{2\,u\Delta t}{\sqrt[4]{3}\,\Delta x}, \qquad \overline{Cou}_y = \frac{v\Delta t}{\sqrt{\sigma^e}} = \frac{2\,v\Delta t}{\sqrt[4]{3}\,\Delta x}. \qquad (38)$$

and are equal to 0.38 in both directions. In the first row of Table 5 the maximum and minimum concentration values are shown. The root mean square error with respect to the exact solution is also evaluated, as a global measurement of the result quality. In Table 5 the previous performance indicators are compared with indicators of the exact solution and with the results of some others numerical procedures [*Komatsu et al.*, 1997]. The proposed procedure shows a good capability of reproducing the maximum value; it succeeds in avoiding spurious minima and reproducing the shape of the concentration, providing the smallest values of minimum and RMS. As in the previous example, even better results can be obtained by halving the size of the element sides and multiplying by four the original time step. See the corresponding performance indicators in the last row of Table 5. See also, in Figure 14 and Table 6, the results obtained with the more dense mesh using different values of the average Courant number.

[44] The numerical convergence test gives the results shown in Table 7 for different values of the average Courant number. Observe that in this case the order of convergence is always greater than two.

[45] The last test case is the advection of a known Gaussian adimensional concentration pulse in a nonuniform flow field. The flow field has radial symmetry, with a time-constant flow rate extracted at a point of an indefinite confined two-dimensional aquifer of constant thickness l and porosity ω. Assuming the origin of axes as the extraction point, the velocity vector at a distance r from the origin is directed according to the radial direction and its value is:

$$v_r = \frac{Q}{2\pi r \omega l}, \qquad (39)$$

where Q is the extracted flow rate. The adopted velocity field is affected by an estimation error, which is function of the mesh density.

[46] Equation (22) can be reformulated on the radial direction leading to the following 1-D form:

$$\frac{\partial c}{\partial t} - v_r \frac{\partial c}{\partial r} = 0. \qquad (40)$$

The exact solution can be easily found as:

$$c(r, t) = c\left(\sqrt{r^2 - \frac{Qt}{\pi \omega l}}, 0 \right). \qquad (41)$$

In the test we assume $Q = 1$ m³/s, $l = 10$ m, $\omega = 0.1$. The initial concentration at a point P is evaluated as:

$$c(r, 0) = C_0 \exp\left(-\frac{(\overline{PP_0})^2}{2\sigma_0^2} \right) \qquad (42)$$

in which r is the radial abscissa of point P. The peak location P_0 has coordinates $x_0 = 32.5$ m and $y_0 = 32.6$ m, and we assume $C_0 = 1$ and $\sigma_0 = 2$ m (see Figure 15). The numerical procedure has been carried out using a regular equilateral triangular mesh with $\Delta x = 1$ m and a time step $\Delta t = 100$ s. Because of the non uniform velocity field, the average Courant numbers vary from one point to the another. The range of average Courant numbers in x and y directions is [0.02 : 3.18].

[47] In Figure 16 the results of the numerical procedure are compared with the analytical solution, at different iterations. Again, no instabilities occur and the underestimation of the peak value is small. In Table 8 the maximum and minimum values are shown, together with the RMS value, at different times. Observe that the RMS maintains a small value throughout the iterations. The numerical convergence test performed in this case gives the results shown in Table 9.

4. Mass Conservation and Irrotational Velocity Field Limitation

[48] Element concentrations, at the end of each time step, are given by the analytical solution of the system of ODEs given by the conservation differential equations of the mass, as well as of the x and y first-order spatial moments. Because of this, the local mass balance is satisfied for given estimation of the total entering flux, that is:

$$\overline{Fe}^e = \frac{c^e(t + \Delta t) - c^e(t)}{\Delta} l + \overline{Fl}^e, \qquad (43)$$

where $c^e(t)$ and $c^e(t + \Delta t)$ are the mean (in space) concentration values at the beginning and at the end of the time step, $\overline{Fe}^e$ is the given time average of the total entering flux and $\overline{Fl}^e$ is the time average of the computed total leaving flux of element e. Entering fluxes are set equal to

Table 9. Two-Dimensional Test 3: Convergence Numerical Test

Number of Elements	Step Length, m	Average Error	Order
66	8	3.04E-02	1.63
264	4	9.80E-03	1.53
1056	2	3.39E-03	

the approximation of the leaving fluxes of the upstream elements by equations (A2) and (A4) (see Appendix A). Equation (B7) (see Appendix B) guarantees that the mean (in time) element concentrations, estimated by the solution of the system of ODEs, are equal to the mean of the approximated ones. Because the entering volumetric flux is, for each side of each element, equal to the flux leaving from the common side of the next upstream element, this also implies that, for the same side, the mean of the assigned entering flux is equal to the mean of the computed leaving flux of the next upstream element. Because of this, the sum of equations (43) provides:

$$\sum_u \overline{Fe}_u = \sum_e \frac{\tilde{c}^e(t + \Delta t) - \tilde{c}^e(t)}{\Delta t} + \sum_d \overline{Fl}_d \qquad (44)$$

where $\overline{Fe}_u$ is the total mean flux entering in each upstream boundary element and $\overline{Fl}_d$ is the total mean flux leaving from each downstream boundary element. Equation (44) guarantees the global mass conservation.

[49] The proposed algorithm has been developed for the solution of problems where velocity is proportional to the gradient of a scalar potential and vorticity is zero. The relationship between vorticity (or rotationality) and potential can be found in any introductory book of water wave mechanics, like the text of *Dean and Dalrymple* [1992]. Velocity fields displaying vorticity, like shallow water bodies with high-frequency waves or velocity fields producing scours around piers of a bridge cannot be treated according to the proposed numerical scheme. On the other hand, because the methodology requirement is to have fluxes always moving from points with higher to points with lower scalar potential, the technique can also be applied if the more general condition

$$\mathbf{v} = -\mathbf{K}(H)\,\mathrm{grad}H \qquad (45)$$

holds, where $\mathbf{v}$ is the velocity vector, H is the scalar potential and $\mathbf{K}$ is a semipositive definite matrix. This implies that the algorithm can be applied for the simulation of all the groundwater transport problems where the Darcy's law holds, with saturated or unsaturated, isotropic or anisotropic porous medium.

[50] A scalar potential also exists in shallow water flow fields if the inertial terms are neglected in the depth-averaged form of the Navier-Stokes equations, called Saint-Venant equations. This simplification of the momentum equation can be adopted in all the flow-routing problems where the upstream flow wave has a large enough time period [*Tsai*, 2003]. An application of the early version of the algorithm to this problem, using a piecewise constant approximation of the unknown water depth, is given by *Noto and Tucciarelli* [2001] for the 1-D network case and by *Tucciarelli and Termini* [2000] for the 2-D case. In both the groundwater transport and shallow water applications the scalar potential is the piezometric head. In the first case piezometric heads are known from the previous solution of the flow problem and in the second case they have to be iteratively computed using a fractional step methodology. The fractional step methodology splits the equations in a nonlinear advective component, which is solved with the

proposed technique, and in a linear diffusive component that is solved with a standard Galerkin method.

5. Conclusions

[51] The proposed algorithm for the numerical solution of the advection problem has the following appealing properties: (1) unconditional stability, (2) second-order approximation of the unknown concentration within each computational element, and (3) local and global mass conservation. The 1-D and 2-D tests suggest a computational accuracy similar to other explicit second-order methods, but the unconditional stability of the algorithm allows the use of non structured meshes and the choice of a time step based on the average size of the elements and the average norm of the velocity. This should make the algorithm competitive about the time computation required for each time step. In the 2-D case (see the pseudocode in Appendix C) the solution of five factorized linear systems of order three for each element are needed, along with the computation of the corresponding right hand sides. Of course, a preliminary work is required for the factorization of the matrices and the element ordering according to their scalar potential. Further improvement of the results, in the case of initially discontinuous functions, can be obtained with the use of a function limiter as will be discussed in a future work. The major limitation of the algorithm is the need of a scalar potential for the velocity field, which includes, however, quite large classes of environmental and engineering problems. Another limitation is the serial structure of the computations, which impairs, in the 1-D case, the use of parallel computing; this limitation is partially avoided in 2-D and 3-D cases, where several elements can be solved simultaneously along different flux pipes.

Appendix A: Coefficients of the Linear Differential System

[52] The matrix and vector elements in equation (12) are

One-dimensional case

$$A^e_{1,1} = \frac{-3u^e}{\Delta x} \quad A^e_{1,2} = \frac{-u^e}{\Delta x}$$
$$A^e_{2,1} = \frac{3u^e}{\Delta x} \quad A^e_{2,2} = \frac{u^e}{\Delta x} \qquad (A1)$$

$$b_1 = \frac{4u^e}{\Delta x}\xi_2^{e-1} \quad b_2 = \frac{-2u^e}{\Delta x}\xi_2^{e-1}. \qquad (A2)$$

Two-dimensional case

$$A^e_{i,j} = \sum_{m=1}^{3} (\mathbf{B}^e)^{-1}_{i,m} U_{m,j}; \qquad (A3)$$

where coefficients of matrix $\mathbf{B}^e$ are given by equation (31). Call jp and jm the nodes following and preceding node j in counterclockwise direction; the coefficients of matrix U are the following:

$$U_{1,j} = -\frac{\delta_{j}O^e_j L^e_j + \delta_{jm}O^e_{jm}L^e_{jm}}{2\sigma^e}, \qquad (A4)$$

$$U_{2,j} - \delta_j \phi_j^e L_j^e \left(\frac{x_{jp}^e}{3} - \frac{x_j^e}{2} - \frac{x_{jp}^e}{2} \right) - \delta_{jm} \phi_{jm}^e L_{jm}^e$$

$$\cdot \left(\frac{x_j^e - x_{jm}^e}{3} + \frac{x_{jm}^e}{2} \right) + \frac{u^e \sigma^e}{3}, \tag{A5}$$

$$U_{3,j} - \delta_j \sigma_j^e L_j^e \left(\frac{y_{jp}^e}{3} - \frac{y_j^e}{2} - \frac{y_{jp}^e}{2} \right)$$

$$- \delta_{jm} \sigma_{jm}^e L_{jm}^e \left(\frac{y_j^e - y_{jm}^e}{3} + \frac{y_{jm}^e}{2} \right) + \frac{v^e \sigma^e}{3}. \tag{A6}$$

[53] The vector coefficients of the polynomial equation (13) have components

$$b_i^n - \sum_{j-1}^{3} (\mathbf{B}^e)_{i,j}^{-1} V_{j,n} \qquad n - 0, 1, 2, 3. \tag{A7}$$

[54] Call $\xi_j^{m,n}$ and $\xi_{jp}^{m,n}$ the polynomial approximation of the concentrations at the nodes of the upstream element m sharing the side with nodes j and jp, or the assigned upstream boundary concentration. Elements of matrix $\mathbf{V}$ are the following:

$$V_{1,n} - - \sum_{j-1}^{3} \frac{(1 - \delta_j)\left(\xi_j^{m,n} - \xi_{jp}^{m,n}\right) \sigma_j^e L_j^e}{2\sigma^e}, \tag{A8}$$

$$V_{2,n} - - \sum_{j-1}^{3} (1 - \delta_j) \phi_j^e L_j^e$$

$$\cdot \left[\left(\xi_{jp}^{m,n} - \xi_j^{m,n} \right) \frac{x_{jp}^e}{3} - \frac{x_j^e}{2} - \frac{\xi_{jp}^{m,n} x_j^e}{2} - \frac{\xi_j^{m,n} x_{jp}^e}{2} \right]. \tag{A9}$$

$$V_{3,n} = - \sum_{j-1}^{3} (1 - \delta_j) \sigma_j^e L_j^e$$

$$\cdot \left[\left(\xi_{jp}^{m,n} - \xi_j^{m,n} \right) \frac{y_{jp}^e - y_j^e}{3} + \frac{\xi_{jp}^{m,n} y_j^e + \xi_j^{m,n} y_{jp}^e}{2} \right]. \tag{A10}$$

Appendix B: Polynomial Approximation

[55] The mean values of the homogeneous solution of equation (12) in the 1-D case are given by:

$$\bar{c}_1^e - \frac{\mathbf{u}_1^e}{\Delta t \lambda_1^e} e^{\lambda_1^e \Delta t} \qquad \bar{c}_2^e - \frac{\mathbf{u}_2^e}{\Delta t \lambda_2^e} e^{\lambda_2^e \Delta t}, \tag{B1a}$$

or

$$\bar{c}_1^e - \frac{\lambda_r^e e^{\lambda_r^e \Delta t}}{\Delta t \left[\left(\lambda_r^e \right)^2 + \left(\lambda_i^e \right)^2 \right]} \left[\mathbf{u}_r^e \left(\frac{\lambda_i^e}{\lambda_r^e} \sin(\lambda_i^e \Delta t) - \cos(\lambda_i^e \Delta t) \right) \right.$$

$$\left. - \mathbf{u}_i^e \left(\sin(\lambda_i^e \Delta t) - \frac{\lambda_i^e}{\lambda_r^e} \cos(\lambda_i^e \Delta t) \right) \right] - \frac{\lambda_r^e \mathbf{u}_r^e + \lambda_i^e \mathbf{u}_i^e}{\Delta t \left[\left(\lambda_i^e \right)^2 + \left(\lambda_r^e \right)^2 \right]}$$

$$\bar{c}_2^e = \frac{\lambda_r^e e^{\lambda_r^e \Delta t}}{\Delta t \left[\left(\lambda_i^e \right)^2 + \left(\lambda_r^e \right)^2 \right]} \left[\mathbf{u}_i^e \left(\frac{\lambda_i^e}{\lambda_r^e} \sin(\lambda_i^e \Delta t) - \cos(\lambda_i^e \Delta t) \right) \right.$$

$$\left. + \mathbf{u}_r^e \left(\sin(\lambda_i^e \Delta t) - \frac{\lambda_i^e}{\lambda_r^e} \cos(\lambda_i^e \Delta t) \right) \right] - \frac{\lambda_r^e \mathbf{u}_i^e - \lambda_i^e \mathbf{u}_r^e}{\Delta t \left[\left(\lambda_i^e \right)^2 + \left(\lambda_r^e \right)^2 \right]}. \tag{B1b}$$

[56] In the 2-D case the following equations complete the set (B1):

$$\bar{c}_3^e - \mathbf{u}_3^e \frac{e^{\lambda_3^e \Delta t}}{\lambda_3^e \Delta t}, \tag{B2}$$

[57] The following equation guarantees the equality of the initial value ($t = 0$):

$$\xi^{e,0} = \mathbf{c}_0^e. \tag{B3}$$

[58] The following equation guarantees the equality of the final value ($t = \Delta t$):

$$\xi^{e,0} + \xi^{e,1} \Delta t + \xi^{e,2} \Delta t^2 + \xi^{e,3} \Delta t^3 - \mathbf{c}^e. \tag{B4}$$

[59] The following equation guarantees the equality of the mean value:

$$\xi^{e,0} + \xi^{e,1} \Delta t/2 + \xi^{e,2} \Delta t^2 \; 3 + \xi^{e,3} \Delta t^3/4$$

$$- \alpha_1 \bar{c}_1^e + \alpha_2 \bar{c}_2^e + v_0 + v_1 \Delta t/2 + v_2 \Delta t^2 \; 3 + v_1 \Delta t^3/4. \tag{B5}$$

[60] The following equation approximates the first-order moments:

$$\xi^{e,1} \Delta t^2/2 + \xi^{e,2} \Delta t^3/3 + \xi^{e,3} \Delta t^4/4 - \xi^{e,4} \Delta t^5/5 - \mathbf{M} \xi^e. \tag{B6}$$

[61] In the 2-D case equation (B5) is replaced by the following:

$$\xi^{e,0} + \xi^{e,1} \Delta t/2 + \xi^{e,2} \Delta t^2 \; 3 + \xi^{e,3} \Delta t^3/4$$

$$- \alpha_1 \bar{c}_1^e \; | \; \alpha_2 \bar{c}_2^e \quad \alpha_3 \bar{c}_3^e \; | \; v_0 \; | \; v_1 \Delta t/2 \; | \; v_2 \Delta t^2/3 \; | \; v_1 \Delta t^3/4. \tag{B7}$$

Appendix C: Pseudocode for the 2-D Case

[62] Solve the flow field
    Begin cycle for $e = 1, \ldots, N$
    Compute u^e, v^e, σ^e, ϕ_i^e, L_i^e, $i = 1, \ldots, 3$
    Compute $\mathbf{A}^e$ (equations (A1)–(A6))
    Compute λ_1^e, λ_2^e, λ_3^e, $\mathbf{u}_1^e$, $\mathbf{u}_2^e$, $\mathbf{u}_3^e$ or λ_r^e, λ_i^e, λ_3^e, $\mathbf{u}_r^e$, $\mathbf{u}_i^e$, $\mathbf{u}_3^e$
    Compute $\mathbf{c}_1^e$, $\mathbf{c}_2^e$, $\mathbf{c}_3^e$ (equations (15) and (33)) and $\bar{c}_1^e$, $\bar{c}_2^e$, $\bar{c}_3^e$ (equations (B1) and (B2)).
    End cycle
[63] Sort the element according to the decreasing piezometric head. Call **I** the vector of the ordered element indexes.
    Begin cycle for $k - 1, \ldots, N_t$
      Begin cycle for $i - 1, \ldots, Ne - I_i$
      Compute $\mathbf{b}^0$, $\mathbf{b}^1$, $\mathbf{b}^2$, $\mathbf{b}^3$ (Equations (A7)–(A10))
      Compute v^3 (equation (16a))
      Compute v^2 (equation (16b))
      Compute v^1 (equation (16c))
      Compute v^0 (equation (16d))
      Compute α_1, α_2, α_3 (equation (34))
      Compute $\mathbf{c}^e$ (equation (32)) at $t - \Delta t$

Compute the vector coefficients of the
  approximating concentrations of c^e: $\xi^{e,0}$, $\xi^{e,1}$,
$\xi^{e,2}$, $\xi^{e,3}$ (equations (B3), (B4), (B6), (B7))
Update initial values: $c^e \rightarrow c_0^e$
End cycle
End cycle

Notation

A^e	matrix of the system of ODEs.
$b(t)$	known vector of the system of ODEs.
b^0, b^1, b^2, b^3	vector coefficients of the third-order polynomial b.
B^e	square coefficient matrix.
c	unknown concentration function.
c^e	piecewise linear approximation of c in the eth element.
c^e	concentration vector of the c^e at the nodes of the eth element.
c_i^e	values of c^e at the ith node of the eth element.
c_0^e	vector of the initial concentration values in element e.
c_1^e, c_2^e, c_3^e	solutions of the homogeneous problem coupled to the system of ODEs.
$\bar{c}^e$	spatial average value of concentration in element e.
$\overline{c}_1^e$, $\overline{c}_2^e$, $\overline{c}_3^e$	time mean value of the homogeneous solutions at element e.
C_0, C_1, C_2	adimensional concentration peak values.
Cou_x, Cou_y	courant numbers in the x and y directions;
$\overline{Cou_x}$, $\overline{Cou_y}$	average Courant numbers in the x and y directions.
$\overline{E}$	average error between the exact and the computed solution.
Fe^e, Fl^e	entering and leaving concentration fluxes.
$\overline{Fe}^e$, $\overline{Fl}^e$	time average value of the total entering and leaving fluxes at element e.
$\overline{Fe}_u$	time average value of the total flux entering in the upstream boundary elements.
$\overline{Fl}_d$	time average value of the total flux leaving from the downstream boundary elements.
H	scalar potential.
I	vector of the ordered element indexes.
$I_1, I_2, \ldots, I_N$	sorting indexes of the N elements.
K	semipositive definite matrix;
l	constant thickness of an indefinite confined 2-D aquifer.
L	domain space length in the 1-D case.
L_i^e	length of the ith side of the element e.
M^e	moment of the real concentrations in the 1-D case.
$\overline{M}^e$	moment of the linear approximation of c inside element e with respect to its downstream end (1-D case).
Mx^e, My^e	moments of the real concentration at element e in x and y directions (2-D case).
$\overline{Mx}^e$, $\overline{My}^e$	moments of the linear approximation of c inside element e (2-D case).
$M\xi^e$	vector of the numerically estimated first-order concentration moments at the element nodes.
n	number of nodes of each element.
N	number of computational elements.
N_t	number of time steps.
N_i^e	Galerkin shape function of the ith node inside the eth element.
P	generic point of the spatial domain.
P_0, P_1, P_2	locations of the concentration peaks.
Q	flow rate.
r, s, x, y	spatial abscissas.
t	temporal abscissa.
t_{min}	time corresponding to the minimum approximated concentration.
T	domain time length.
u, v, v_r	known velocity components in the x, y an r directions.
u^e, v^e	known components of the velocity vector at element e.
u_1^e, u_2^e, u_3^e	real eigenvectors.
u_j^e	imaginary part of a complex eigenvector.
u_r^e	real part of a complex eigenvector.
U	coefficient matrix for the evaluation of A^e.
v	velocity vector.
v^0, v^1, v^2, v^3	vector coefficients of the zero, first, second and third power of t in the c^e solution.
V	coefficient matrix for the evaluation of the b vector coefficients.
x_i	spatial coordinate of the generic point i.
x_0, y_0	coordinates of the initial location of the concentration peaks.
x_i^e, y_i^e	coordinates of the ith node of the element e.
α	numerical weighting coefficient.
α_{ij}	numerical coefficients for the evaluation of the spatial moment of c.
$\alpha_1, \alpha_2, \alpha_3$	arbitrary coefficients of the solutions of the homogeneous system.
δ_i	adimensional coefficient equal to 0 or 1.
$\Delta x, \Delta y$	spatial steps.
Δt	time step.
$\lambda_1^e, \lambda_2^e, \lambda_3^e$	real eigenvalues.
λ_i^e	imaginary part of a complex eigenvalue.
λ_r^e	real part of a complex eigenvalue.
ξ^e	polynomial time approximation vector of the concentrations at the nodes of element e.
ξ_i^e	elements of vector ξ^e, i $-$ 1, $\ldots$, 3 (function of t).
$\xi^{e,0}, \xi^{e,1}, \xi^{e,2}, \xi^{e,3}$	vector coefficients of ξ^e.
$\overline{\xi}_i^e$	mean (in time) value of ξ_i^e
ξ_m	approximated concentration of the entering flux.

$\sigma_0, \sigma_1, \sigma_2$ standard deviation of the c distributions.

σ_e area of element e.

σ_i^e flux per unit length trough the ith side of the element e.

ω porosity value.

References

Abbott, M. B. (1979), *Computational Hydraulics*, Pitman, London.

Behrens, J., and A. Iske (2002), Grid-free adaptive semi-Lagrangian advection using radial basis functions, *Comput. Math. Appl.*, *43*, 319–327.

Bella, D. A., and W. J. Grenney (1970), Finite-difference convection errors, *J. Sanit. Eng. Div. Am. Soc. Civ. Eng.*, *96*(6), 1361–1375.

Bensabat, J., Q. Zhou, and J. Bear (2000), An adaptive pathline-based particle tracking algorithm for the Eulerian-Lagrangian method, *Adv. Water Resour.*, *23*, 383–397.

Bramski, J. M., and E. R. Holley (1986), Advection calculation using spline schemes, in *Proceedings of Water Forum '86*, vol. 2, edited by M. Karamouz et al., pp. 1807–1814, Am. Soc. of Civ. Eng., Reston, Va.

Brooks, A. N., and T. J. R. Hughes (1982), Stream-line upwind Petrov-Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Eng.*, *32*, 199–259.

Celia, M. A., T. F. Russell, I. Herrera, and R. E. Ewing (1990), An Eulerian-Lagrangian localized adjoint method for the advection-diffusion equation, *Adv. Water Resour.*, *13*(4), 187–206.

Chilakapati, A. (1999), A characteristic-conservative model for Darcian advection, *Adv. Water Resour.*, *22*(6), 597–609.

Cockburn, B., and C.-W. Shu (1998), The Runge-Kutta discontinuous Galerkin method for conservation laws V. Multidimensional systems, *J. Comput. Phys.*, *141*, 199–224.

Dean, G. R., and R. A. Dalrymple (1992), *Water Wave Mechanics for Engineers and Scientists*, *Adv. Ser. Ocean Eng.*, vol. 2, World Sci., River Edge, N. J.

de Marsily, G. (1986), *Quantitative Hydrogeology*, Academic, San Diego, Calif.

Donea, J. (1984), A Taylor-Galerkin method for convective transport problems, *Int. J. Numer. Methods Eng.*, *20*, 101–119.

Egan, B., and J. R. Mahoney (1972), Numerical modelling of advection and diffusion of urban area source pollutants, *J. Appl. Meteorol.*, *11*(2), 312–321.

Gray, W. G., and G. F. Pinder (1983), An analysis of the numerical solution of the transport equation, *Water Resour. Res.*, *12*, 547–555.

Healy, R. W., and T. F. Russell (1993), A finite-volume Eulerian-Lagrangian localized adjoint method for solution of the advection-dispersion equation, *Water Resour. Res.*, *29*, 2399–2414.

Holly, F. M., and A. Preissmann (1977), Numerical alternatives in transient stream response, *J. Hydraul. Div. Am. Soc. Civ. Eng.*, *103*(11), 1259–1277.

Hughes, T. J. R., L. P. Franca, and G. M. Hulbert (1989), A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least squares method for advective-diffusive equations, *Comput. Methods Appl. Mech. Eng.*, *19*, 173–189.

Huyakorn, P. S., and G. F. Pinder (1983), *Computational Methods in Subsurface Flow*, Academic, San Diego, Calif.

Kaazempur-Mofrad, M. R., and C. R. Ethier (2002), An efficient characteristic Galerkin scheme for the advection equation in 3-D, *Comput. Methods Appl. Mech. Eng.*, *191*, 5345–5363.

Kinzelback, Z., and G. Cordes (1992), Continuous Groundwater velocity fields and path lines in linear, bilinear and trilinear finite elements, *Water Resour. Res.*, *28*, 2903–2912.

Komatsu, T., K. Ohgushi, and K. Asai (1997), Refined numerical scheme for advective transport in diffusion simulation, *J. Hydraul. Eng.*, *123*(1), 41–50.

Leonard, B. P. (1979), A stable and accurate convective modelling procedure based on quadratic upstream interpolation, *Comput. Methods Appl. Mech. Eng.*, *19*, 59–98.

Leonard, B. P. (1991), The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection, *Comput. Methods Appl. Mech. Eng.*, *88*, 17–74.

Li, C. W. (1990), Advection simulation by minimax-characteristics method, *J. Hydraul. Eng.*, *116*(9), 1138–1144.

Nassiri, M., and S. Babarutsi (1997), Computation of dye concentration in shallow recirculating flow, *J. Hydraul. Eng.*, *123*(9), 793–805.

Noto, V., and T. Tucciarelli (2001), The DORA algorithm for network flow models with improved stability and convergence properties, *J. Hydraul. Eng.*, *127*(5), 380–391.

Oliveira, A., and A. M. Baptista (1998), On the role of tracking on Eulerian-Lagrangian solutions of the transport equation, *Adv. Water Resour.*, *21*, 539–554.

Pokrajac, D., and R. Lazic (2002), An efficient algorithm for high accuracy particle tracking in finite elements, *Adv. Water Resour.*, *25*, 353–369.

Ponce, V. M., Y. H. Chen, and D. B. Simons (1979), Unconditional stability in convection computations, *J. Hydraul. Div. Am. Soc. Civ. Eng.*, *105*(9), 1079–1086.

Rider, W. J., and D. B. Kothe (1998), Reconstructing volume tracking, *J. Comput. Phys.*, *141*, 112–152.

Schohl, G. A., and F. M. Holly Jr. (1991), Cubic-spline interpolation in Lagrangian advection computation, *J. Hydraul. Eng.*, *117*(2), 248–253.

Sobey, R. J. (1984), Numerical alternatives in transient stream response, *J. Hydraul. Eng.*, *110*(6), 749–772.

Stefanovic, D. L., and H. G. Stefan (2001), Accurate two-dimensional simulation of advective-diffusive-reactive transport, *J. Hydraul. Eng.*, *127*(9), 728–737.

Tsai, C. W. (2003), Applicability of kinematic, noninertia and quasi-steady dynamic wave model to unsteady flow routing, *J. Hydraul. Eng.*, *129*(8), 613–627.

Tucciarelli, T., and F. Fedele (2000), An efficient double order solution of the groundwater contaminant transport problem, paper presented at XIII International Conference on Computational Methods in Water Resources, Univ. of Calgary, Calgary, Alberta, Canada, June.

Tucciarelli, T., and D. Termini (2000), Finite-element modeling of floodplain, *J. Hydraul. Eng.*, *126*(6), 416–424.

Van Leer, B. (1977), Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection, *J. Comput. Phys.*, *23*, 276–299.

Venezian, G. (1984), Discrete models for advection, *J. Hydraul. Eng.*, *110*(10), 1325–1339.

Wang, J. D., S. V. Cofer-Shabica, and J. C. Fatt (1988), Finite element characteristic advection model, *J. Hydraul. Eng.*, *114*(9), 1098–1114.

Yeh, G. T. (1990), A Lagrangian-Eulerian method with zoomable hidden line-mesh approach to solving advection-dispersion equations, *Water Resour. Res.*, *26*, 1133–1144.

Zhu, J. (1991), A low-diffusive and oscillation-free convection scheme, *Commun. Appl. Numer. Methods*, *7*(3), 225–232.

A. Bascià, Department of Mechanics and Materials, Mediterranean University of Reggio Calabria, Via Graziella, Feo di Vito, 89100 Reggio Calabria, Italy. (bascia@ing.unirc.it)

T. Tucciarelli, Department of Hydraulic Engineering and Environmental Applications, University of Palermo, Viale delle Scienze, 90128 Palermo, Italy.