Solution of the transport equations using a moving coordinate system

OLE KROGH JENSEN and BRUCE A. FINLAYSON

Department of Chemical Engineering, University of Washington, Seattle, Washington 98195, USA
(Received 4 March 1979)

A convection–diffusion equation arises from the conservation equations in miscible and immiscible flooding, thermal recovery, and water movement through desiccated soil. When the convection term dominates the diffusion term, the equations are very difficult to solve numerically. Owing to the hyperbolic character assumed for dominating convection, inaccurate, oscillating solutions result. A new solution technique minimizes the oscillations. The differential equation is transformed into a moving coordinate system which eliminates the convection term but makes the boundary location change in time. We illustrate the new method on two one-dimensional problems: the linear convection–diffusion equation and a non-linear diffusion type equation governing water movement through desiccated soil. Transforming the linear convection diffusion equation into a moving coordinate system gives a diffusion equation with time dependent boundary conditions. We apply orthogonal collocation on finite elements with a Crank–Nicholson time discretization. Comparisons are made to schemes using fixed coordinate systems. The equation describing movement of water in dry soil is a highly non-linear diffusion-type equation with coefficients varying over six orders of magnitude. We solve the equation in a coordinate system moving with a time-dependent velocity, which is determined by the location of the largest gradient of the solution. The finite difference technique with a variable grid size is applied, and a modified Crank–Nicholson technique is used for the temporal discretization. Comparisons are made to an exact solution obtained by similarity transformation, and with an ordinary finite difference scheme on a fixed coordinate system.

INTRODUCTION

The numerical solution of convection–diffusion problems is notoriously difficult when convection dominates because the equation then assumes a hyperbolic character. The convection–diffusion equation is:

$$\frac{\partial c}{\partial t} + Pe\frac{\partial c}{\partial x} = \frac{\partial^2 c}{\partial x^2} \tag{1}$$

We study solutions with the following boundary conditions.

$$\left.\begin{array}{ll} c = 1 & x = 0 \\[2mm] \dfrac{\partial c}{\partial x} = 0 & x = 1 \end{array}\right\} \; t > 0$$

$$c = 0 \quad x \geqslant 0 \quad t = 0$$

Clearly at high values of the Peclet number Pe the transient equation (1) is hyperbolic in character.

Price et al.[20] were the first to recognize that the difficulties are due mainly to the spatial discretization. They proved that a finite difference solution with a central difference approximation will not oscillate provided:

$$Pe\Delta x \leqslant 2 \tag{2}$$

They also showed that when a first order difference

expression is used for the convective term, the finite difference solution will not oscillate at all. Similar criteria as equation (2) can be obtained for weighted residual methods[6,12]. The limits get slightly larger for higher order methods.

Other methods have been used for examining the oscillations. Stone et al.[24] analysed the transient equation with a Fourier-series method. They compared a single component of the Fourier series to the same component in the exact solution of the difference equation. Gray and Pinder[18], Gresho et al.[10] and Runchal[22] made similar comparisons of finite difference formulations with weighted residual methods. The general conclusion is that weighted residual methods are better based on equal number of nodes. Lantz[14] and Chaudhari[4] examined the truncation error of the finite difference method applied to the transient equations. They found that backward schemes introduced artificial dispersion and thereby smoothed the front, and that this effect could be reduced by a judicious choice of spatial and temporal step sizes.

Oscillations will be introduced into the transient solution if the temporal integration is not appropriate. Price et al.[20] showed that an explicit, Euler scheme will not oscillate provided:

$$\Delta t \leqslant 1/\lambda_m \tag{3}$$

where λ_m is the maximum eigenvalue of the matrix resulting from the discretized equations. An implicit, backward Euler temporal integration method does not

oscillate at all, whereas a Crank–Nicholson method (equal weighting at the old and new times) does not oscillate when:

$$\Delta t \leqslant 2/\lambda_m \qquad (4)$$

Similar guidelines are provided by Smith *et al.*[23] for other temporal integration schemes. We assume in further discussion that the temporal integration scheme used is sufficient to avoid oscillations due to the time discretization.

Lantz[14] combined the spatial and temporal truncation errors in equation (1) by using $\Delta x = Pe\Delta t$ in an explicit formulation using backward differences for the convective derivative. Van Genuchten[25] and Laumbach[15] have both made equivalent combinations (using higher order temporal integrations) to improve the accuracy of the calculations. As yet these manipulations have not been extended to non-linear problems, although Lantz[14] achieves some success in special non-linear problems.

Summarizing, we know how to eliminate oscillations in the solution due to the temporal integration, and for some methods we know the spatial increment needed to eliminate oscillations due to the convective term. Unfortunately, the spatial increment (Δx) needed is frequently much too small and we are forced to degrade the quality of the solution by introducing artificial dispersion in the numerical scheme. We propose an alternative method, the moving coordinate system, that permits larger spatial increments (Δx) over part of the domain without degrading the solution from oscillations or artificial dispersion.

We first present some alternative solution methods as applied to the convective diffusive problems and then describe the moving coordinate system as a numerical method. Two examples are given, the linear convection diffusion equation (1) and a non-linear diffusion type equation. For both examples exact solutions and appropriate numerical schemes are discussed. The results using both a moving coordinate system and conventional methods are compared in accuracy and computation cost. For the non-linear diffusion equation an eigenvalue analysis is performed, which further clarifies the numerical problems.

ALTERNATIVE SOLUTION METHODS

A great variety of solution methods have been applied to the convection–diffusion equation: finite difference and weighted residual formulations (Galerkin, collocation, etc.). Techniques used to improve the efficiency include variable interpolation and the method of characteristics.

The finite difference formulation initiated by Peaceman and Rachford[17] was followed by schemes of increasing complexity. Stone and Brian[24] introduced higher order spatial schemes while Laumbach[15] used high order in both space and time. Unfortunately, the complex schemes cannot always be adopted to more difficult non-linear problems in two dimensions.

The Galerkin finite element method was applied by Price *et al.*[19] and provided better results than the simple finite difference schemes, especially for large *Pe*, 10^3–10^6. Young[27] examined a class of weighted residual methods including orthogonal collocation and moment methods. Huyakorn[11] used an upstream Galerkin method; Chien[5]

used an upstream finite difference scheme. Both add artificial dispersion to dampen the oscillations.

While capable of solving the convection–diffusion equation, the conventional schemes require a large number of elements, nodes and timesteps when convection dominates. Extensive storage space and computer time are necessary for two-dimensional cases. Therefore alternate techniques are desirable.

Price *et al.*[19] introduced a variable interpolation technique by adding points before the front when it advances and removing those behind the front after it passes. This technique greatly improves the efficiency but is not easily adopted to more difficult problems. The results do, however, show that the small elements dictated by equation (2) are needed only in the vicinity of the front; large elements can be used elsewhere. Garder *et al.*[9] used a method of characteristics. The differential equations are solved on a fixed grid and with a set of moving points. The velocity of these points are determined by the characteristics. Diffusion is modelled by a marker-and-cell technique, which limits the accuracy. Errors smaller than 2% were not achieved and computer storage is a limiting factor in two dimensions.

MOVING COORDINATE SYSTEM

The moving coordinate system is based on the observation that only near the sharp change in the solution (front) are small elements needed and elsewhere only a few large elements are necessary. The moving coordinate system (abbreviated MCS) involves a transformation of the differential equation from one coordinate system fixed in time to a coordinate system which moves with the velocity of the front. Starting from a general parabolic equation:

$$\frac{\partial c}{\partial t} + g(c, x, t)\frac{\partial c}{\partial x} = \frac{\partial}{\partial x}\left(d(c, x, t)\frac{\partial c}{\partial x}\right) \qquad (5)$$

the following transformation is performed:

$$\xi = x - \int_0^t \lambda(t')dt'$$

$$\eta = t \qquad (6)$$

$\lambda(t)$ is the velocity of the front, so the integral in equation (6) represents the distance the front has travelled. This velocity can be specified by the user or determined numerically from the solution, as shown below. We now have:

$$\frac{\partial c}{\partial t} = \frac{\partial c}{\partial \xi}\frac{\partial \xi}{\partial t} + \frac{\partial c}{\partial \eta}\frac{\partial \eta}{\partial t} = -\lambda(\eta)\frac{\partial c}{\partial \xi} + \frac{\partial c}{\partial \eta}$$

$$\frac{\partial c}{\partial x} = \frac{\partial c}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial c}{\partial \eta}\frac{\partial \eta}{\partial x} = \frac{\partial c}{\partial \xi}$$

$$\frac{\partial^2 c}{\partial x^2} = \frac{\partial^2 c}{\partial \xi^2} \qquad (7)$$
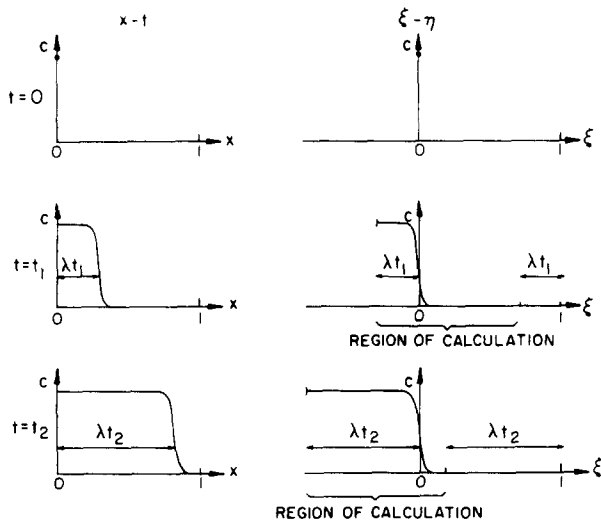
Figure 1. *Fixed and moving coordinate system.*

and equation (5) can be rewritten as:

$$\frac{\partial c}{\partial \eta} = [g(c, \xi, \eta) - \lambda(\eta)]\frac{\partial c}{\partial \xi} = \frac{\partial}{\partial \xi}\left(d(c, \xi, \eta)\frac{\partial c}{\partial \xi}\right) \qquad (8)$$

The boundary conditions must also be transformed. For example, the boundary condition from equation (1) becomes:

$$
\left.
\begin{aligned}
c = 1 \text{ at } \xi &= -\int_0^\eta \lambda(\eta')\mathrm{d}\eta' \\[2ex]
\frac{\partial c}{\partial \xi} = 0 \text{ at } \xi &= 1 - \int_0^\eta \lambda(\eta')\mathrm{d}\eta'
\end{aligned}
\right\} \quad \eta > 0 \qquad (9)
$$

$$c = 0 \text{ at } \xi > 0 \qquad \eta = 0$$

The location of the boundary now changes with time. The revised problem, equations (8, 9) is exact and merely another version of equations (5, 1).

The improvement in MCS comes from an optimum location of nodes. Since the front remains fixed in the moving coordinate system, small elements or more grid points can be concentrated there (where they are needed to eliminate oscillations) but larger elements or fewer grid points can be used elsewhere (where they are not needed). It is the use of large elements away from the front that gives rise to the economy. In a fixed coordinate system the front eventually appears everywhere, so that small elements are needed throughout the domain, leading to excessive computation cost. The key to economy is to pick the right frontal velocity, $\lambda(\eta)$, and use a proper distribution of elements. The difference between calculating in a fixed or moving coordinate system is illustrated in Fig. 1.

Considering the linear equation (1), the front moves with a constant velocity $Pe(\lambda(t) = Pe)$, so we get simply the diffusion equation:

$$\frac{\partial c}{\partial \eta} = \frac{\partial^2 c}{\partial \xi^2} \qquad (10)$$

and the boundary conditions are then:

$$
\left.
\begin{aligned}
c &= 1 \text{ at } \xi = -Pe\eta \\
c &= 0 \text{ at } \xi = 1 - Pe\eta
\end{aligned}
\right\} \quad \eta > 0 \qquad (11)
$$

$$c = 0 \qquad \xi > 0 \qquad \eta = 0$$

With the MCS an LU decomposition must be performed every time the boundary node changes, whereas in a fixed coordinate system only one LU decomposition is done per problem. This disadvantage of MCS is unimportant for non-linear problems and less important for temporal integration methods with a variable step size.

For a non-linear problem the frontal velocity is not known *a priori* and we must measure it by either following a specific concentration level or the largest gradient. This later choice is illustrated in the second example.

The boundary conditions can be applied in two ways. If the solution at the boundary is known to have a zero gradient (or nearly so) we can just apply the boundary condition at the closest node. The location of the boundary in the numerical calculations then proceeds in jumps during the integration. If the solution at the boundary has a non-zero gradient, then the above procedure would introduce significant error. In that case we move the location of the last element so that there is always a node at the actual boundary location. Both procedures are used below.

The distribution of nodes is closely related to the broadening of the initial profile throughout the integration. An estimate of the diffusion length can be found from the solution of the diffusion equation in an infinite medium with an initial step change. The solution is[3]:

$$c = \operatorname{erfc}(x/\sqrt{4Dt})$$

For a change in c of 99% $(c = 0.01)$ we get:

$$\frac{x}{\sqrt{4Dt}} \simeq 1.8$$

so that the diffusion length is:

$$L = 3.6\sqrt{Dt} \qquad (12)$$

Here $D = 1$ and since $\max t \approx 1/Pe$, the maximum L will be:

$$L \simeq 3.6/\sqrt{Pe} \qquad (13)$$

Equation (13) can therefore be used to estimate the element location. Sufficiently small elements must be included in a region of length L about the front if the accuracy is not to be degraded. For a non-linear problem the Peclet number must be estimated in order to obtain the diffusion length.

Application of MCS to two-dimensional problems has been done[13] and will be reported separately. If we compare the MCS to the technique of moving small elements with the front, the MCS method involves less bookkeeping, and the interpolation and adjustment of the solution occurs at the boundary, where the solution is well behaved, rather than at the front, where it is not. The MCS is closely related to the method of characteristics. If there

Table 1.   Comparison of MCS with conventional scheme (both using OCFE)

	Entry	Scheme	NE	NCOL	$\Delta t \times 10^n$	CPU*	Error at $Pe \cdot t = 0.088$	Error at $Pe \cdot t = 0.444$	Maximum oscillations at $Pe \cdot t = 0.444$
$Pe = 877.9$	1	MCS	4	2	2.5	3.3	0.03	0.04	
	2	–	6	2	2.5	5.1	0.033	0.005	
	3	–	6	2	10	1.0	0.04	0.01	
	4	–	8	2	2.5	6.7	0.008	0.008	
	5	–	14	2	1.25	15.1	0.004	0.001	
	6	OCFE	5	2	2	3.2	~0.1	~0.1	0.1
	7	–	15	2	2.5	8.1	0.11	0.05	0.04
	8	–	20	3	2.5	17.0	0.03	0.004	0.0022
					$\Delta t \times 10^8$		x		
$Pe = 87790$	9	MCS	12	2	5	4.5	0.5	0.02	0.005
	10	–	14	2	2.5	8.1	0.5	0.01	0.004
	11	–	16	2	1.25	17.0	0.5	0.004	
	12	OCFE	50	3	1	70.5	0.0	0.003	0.004
	13	–	50	3	0.5	152.0	0.5	~0.1	~0.1
	14	–	50	5	1	140.0	0.0	0.003	

* CPU on CDC 6400

were no second order terms MCS would be a method of characteristics. With second order terms, though, the method of characteristics is not valid. We model the dispersion caused by the second order terms by solving the second order equation (which, in the moving coordinate system, represents diffusion about the front). In contrast Garder et al.[9] use a marker-and-cell technique to model the dispersion. Finally, any method of spatial discretization can be used: we employ both orthogonal collocation on finite elements and finite difference methods in the MCS.

EXAMPLE I: CONVECTION-DIFFUSION EQUATION

The first example is MCS applied to equation (1). The exact solution in an infinite domain is[16]:

$$c(x, t) = 1/2[\text{erfc}(x - Pet)/2\sqrt{t}) + e^{Pex}\text{erfc}(x + Pet)/2\sqrt{t})]$$

which can be used as an approximation for a finite domain provided $c(1, t) \leqslant 10^{-16}$. The numerical scheme and results are discussed.

Numerical method

We can apply any numerical method to solve equations (10, 11). As we are only interested in $\eta < 1/Pe$ the discretization only covers $[-1, 1]$. We choose to apply orthogonal collocation on finite elements (OCFE)[2], giving a set of ordinary differential equations and algebraic equations. We use the notation $j = 1, \ldots, NT$ to identify the jth node at different ξ. Within each element the notation $J = 1, \ldots, NP$ denotes the local numbering systems within the element, which has NP nodes. In the kth element $j = (NP - 1)(k - 1) + J$, since the NPth node of the $(k - 1)$th element is the same as the 1st node of the kth element. We let $c_I^{(k)}$ be the concentration of the Ith node in the kth element. The equations are:

interior collocation: $\dfrac{dc_j}{d\eta} = \dfrac{1}{\Delta \xi_k^2} \displaystyle\sum_{I=1}^{NP} B_{JI} c_I^{(k)}$

continuity: $\dfrac{1}{\Delta \xi_k} \displaystyle\sum_{I=1}^{NP} A_{NP,I} c_I^{(k)} = \dfrac{1}{\Delta \xi_{k+1}} \displaystyle\sum_{I=1}^{NP} A_{1,I} c_I^{(k+1)}$

Using a Crank–Nicholson time discretization gives:

$$\frac{c_j^{n+1} - c_j^n}{\Delta \eta} = \frac{1}{2\Delta \xi_k^2} \sum_{I=1}^{NP} B_{JI}(c_I^{(k),n+1} + c_I^{(k),n})$$

where $c_j^n = c_j(\eta = n\Delta\eta)$. The set of equations can be written as:

$$\mathbf{Cc}^{n+1} = \mathbf{Dc}^n \tag{14}$$

A similar set of equations could be developed using finite difference or Galerkin finite element methods.

The boundary conditions are applied by modifying the matrices $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{c}^n$. For points to the left of the location of the boundary, $j < IL$, $\xi_j < Pe\eta$, we replace $\mathbf{C}$ and $\mathbf{D}$ by the identity matrix $\mathbf{I}$ and $\mathbf{c}^n$ by 1. Similarly for right boundary points we do the same thing if $j > IR$, $\xi_j > 1 - Pe\eta$. The modified matrices are:

$$C_{ji}^* = \begin{cases} I_{ji} \text{ for } j < IL \text{ or } \xi_j < -Pe\eta \\ C_{ji} \text{ for } IL < j < IR \\ I_{ji} \text{ for } j > IR \text{ or } \xi_j > 1 - Pe\eta \end{cases}$$

with a similar definition for D_{ij}

$$\mathbf{c}^{*n} = \begin{cases} 1 & j < IL \\ \mathbf{c}^n & IL < j < IR \\ 0 & j > IR \end{cases}$$

The problem is then:

$$\mathbf{C}^*\mathbf{c}^{*n+1} = \mathbf{D}^*\mathbf{c}^{*n}$$

$\mathbf{C}^*$ is decomposed using an LU decomposition appropriate to its structure and $\mathbf{c}^{*n+1}$ is calculated. Since the location of the boundary changes in time, the LU decomposition must be performed every time the boundary location, $-Pe\eta$ or $1 - Pe\eta$ passes another collocation point.
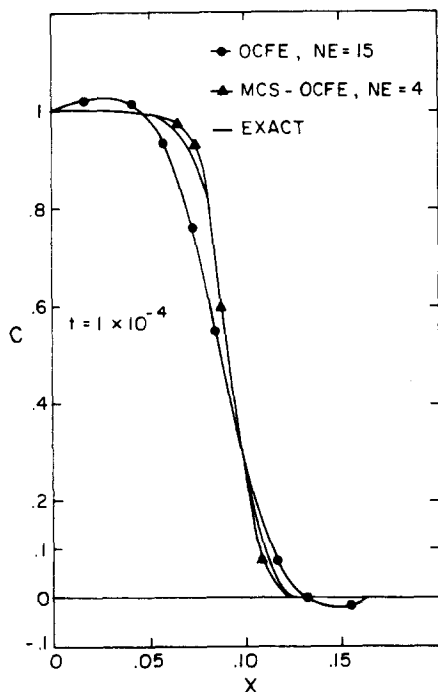
Figure 2. Comparison of standard OCFE versus moving coordinate system. $Pe = 877.9$, $NE = 4$, $NCOL = 2$, $\Delta t = 2.5 \times 10^{-6}$. *Element nodes at* $\xi = -1$, -0.05, 0.0, 0.05, 1.
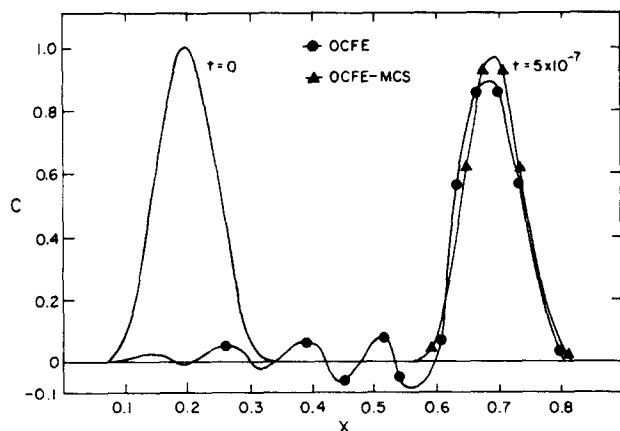

Figure 3. Comparison of standard OCFE versus moving coordinate system. Gaussian initial distribution $Pe = 10^6$, $NE = 13$, $NCOL = 2$, $\Delta t = 2.5 \times 10^{-9}$.

Results

The advantage of using MCS for problems with dominating convection is illustrated by comparisons using the same spatial discretization (here collocation) with both moving coordinate system and a fixed coordinate system. The convection–diffusion equation is solved with $Pe = 877.9$ and 87790.0, with results given in Table 1.

Figure 2 compares a conventional scheme to the MCS-scheme using 4 elements. The scheme with a fixed coordinate system oscillates whereas the MCS solution does not. To eliminate the oscillations in the fixed coordinate scheme requires 20 elements, and an increase of computer time by a factor of 5 (compare entries 1 and 8 in Table 1). Conversely, for non-oscillating solutions, MCS uses only 20% of the computation time needed with a fixed coordinate system.

Another comparison is made in Fig. 3 with a Gaussian initial distribution and $Pe = 10^6$. Both schemes use the same number of elements and the same time-step. Severe oscillations appear in the conventional scheme whereas MCS gives none. The moving coordinate system works best for large Pe and oscillations are not introduced provided the initial conditions can be fit without oscillations. As time proceeds the profiles become less steep due to the dispersion indicated by equation (13), and oscillations are less likely.

MCS is compared to the method of variable interpolation, Price *et al.*[19], in Fig. 4. The measure of accuracy is given by the maximum deviation from the exact solution

$$E(t) = \operatorname*{Max}_{i} \{ |c(x_i, t) - c_{\text{exact}}(x_i, t)| \}$$

Both cases have negligible error caused by the time integration. The error decreases with time for the same number of elements and the MCS is more accurate at small times (when the front is steepest). In contrast to the method of characteristics by Garder *et al.*[9], which showed a minimum error of about 2%, the MCS will converge for increasing number of elements and with smaller timesteps, giving errors of 0.1% for the case shown in Fig. 4.

The elements are not all the same size in MCS–OCFE. When small elements are used near the front the solution will be good at small times (when all changes occur near the front). At larger times, changes also occur further away from the front so that elements are needed there, too. Thus schemes which have slightly bigger elements may be preferred at later times. Figure 5 illustrates the error as a function of time for solutions obtained using different grids.

In Table 1 are shown results, errors, computation time, number of elements and Δt for $Pe = 877.9$ and 87790. It is clear that MCS is superior to the conventional schemes. By having the same number of elements (entries 5 and 7 for 15 elements) the error is decreased by a factor 20–40, but the computation time is only two times longer. For the same computation time (entries 1 and 6 for ~ 3 s) the conventional scheme is unreliable (10%) while MCS gives fairly accurate results (3%). For the same computation time (entries 5 and 8 for 16 s) the error is decreased by a factor of 4–8 and MCS gives no oscillations while the conventional scheme does.
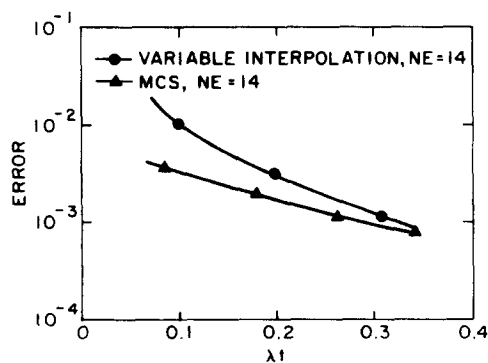

Figure 4. Comparison of moving coordinate system to method with variable interpolation[19], $Pe = 877.9$, MCS $= entry$ 4 *in Table* 1.
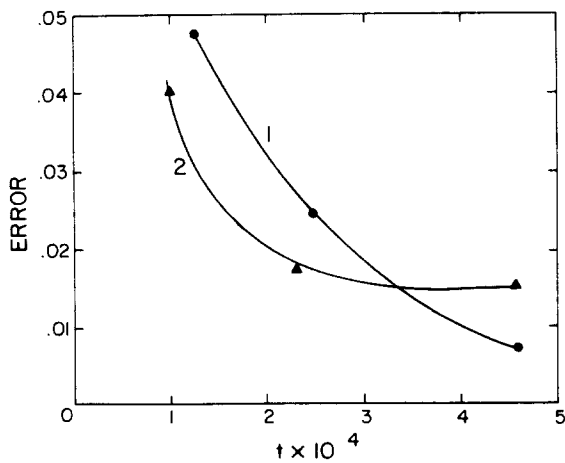
Figure 5. Error as function of time. Effect of mesh location. $Pe = 877.9$, $NE = 5$, $\Delta t = 2.5 \times 10^{-6}$. 1, Element nodes at $\xi = -1$, -0.2, -0.05, 0.05, 0.2, 1; 2, element nodes at $\xi = -1$, -0.12, -0.04, 0.04, 0.12, 1.

For $Pe = 87790$, the MCS is even better. Table 1 shows results for both MCS and conventional OCFE. Comparing entries 13 and 10 the standard OCFE gives excessive oscillation, but MCS gives only 1% error and used only 5% of the computer time.

We conclude that the moving coordinate system is very effective in improving the efficiency of numerical solutions to the convective–diffusion equation. The solutions do not suffer from severe oscillations or from excessive numerical dispersion, yet are still economical to calculate. Further improvement could be made if a variable time step algorithm is incorporated. Since the front is nearly stationary in η, with only slight broadening, larger time steps are possible with MCS than with a fixed coordinate system. Several decompositions are made in MCS during the integration and the timestep can therefore easily be changed based on the local truncation error. Such an algorithm is used with success for two-dimensional cases[13].

EXAMPLE II: NON-LINEAR DIFFUSION–FLOW THROUGH POROUS MEDIA

Another problem whose numerical solution oscillates arises in moisture transport in desiccated soil. A simplified model for the transport was made by Finlayson[7]. The one-dimensional mass balance for a single liquid phase flowing through a porous media is:

$$\frac{\partial}{\partial t}(\varphi \rho S) = -\frac{\partial}{\partial y}(\rho q) \tag{15}$$

where φ is the porosity, ρ is the liquid density, S is saturation (liquid volume/non-solid volume), and q is the volumetric flux. Time is t and distance is y, gravity is neglected. The neglect of gravity allows a similarity solution (see below), makes oscillations more evident, and does not appreciably affect the difficulty of the problem. Darcy's law is assumed:

$$\rho q = -\frac{\rho k}{\mu}\frac{\partial p'}{\partial y}$$

where μ is the liquid viscosity and p' is the liquid pressure.

The equations are non-dimensionalized to give:

$$-\frac{dS}{dP_c}\frac{\partial p}{\partial t} = \frac{\partial}{\partial x}\left(k_r\frac{\partial p}{\partial x}\right) \tag{16}$$

where k_r is the relative permeability, k/k_0, and

$$p = p'/(\rho g L) \qquad t = \tau k_0 \rho g/(\mu L \varphi) \qquad x = y/L$$

Both k_r and S depend on the capillary pressure, which is the air pressure (taken as zero) minus the water pressure:

$$p_c = -p \tag{17}$$

The variations of S and k_r with p_c are taken as:

$$\frac{S - S_r}{1 - S_r} = \frac{1}{1 + (p_c L/A1)^\alpha} \qquad k_r = \frac{1}{1 + (p_c L/B1)^\beta} \tag{18}$$

For a typical soil we use*:

$$S_r = 0.32, \quad L = 100 \text{ cm}, \quad A1 = 231 \text{ cm}, \quad B1 = 146 \text{ cm},$$

$$\alpha = 3.65, \qquad \beta = 6.65$$

The boundary conditions are:

$$\left.\begin{aligned} p &= BPO \text{ for } x = 0 \\[1em] \frac{\partial p}{\partial x} &= 0 \text{ for } x = 1 \end{aligned}\right\} t > 0 \tag{19}$$

$$p = BP1 \text{ for } x > 0 \qquad t = 0$$

Finlayson[7] describes the mathematical difficulties which are associated with solving this equation. The drier the soil is (decreasing BP1), and the more water imposed at $x = 0$ (BPO $= 0$, soil saturated), the more difficult the problem is to solve. A similarity solution is derived to give the exact solution for small times. An eigenvalue analysis further quantifies the numerical difficulties in this problem. The numerical techniques used for solving the non-linear equation in both fixed and moving coordinate systems are described, and comparisons are made between them.

Similarity solution

Equation (16) can be transformed with the similarity variable $\xi = x/\sqrt{t}$.

$$0 = -\frac{1}{2}\xi\frac{ds}{dp_c}\cdot\frac{dp}{d\xi} + \frac{d}{d\xi}\left(k_r\frac{dp}{d\xi}\right) \tag{20}$$

and

$$p = BPO \text{ for } \xi \to \infty$$

$$p = BP1 \text{ for } \xi \to 0$$

---

* The parameters chosen are for a gravelly coarse soil, and varying the initial pressure BP1 enables us to make the problem difficult to solve. A fine soil gives similar results without so severe a choice of BP1.
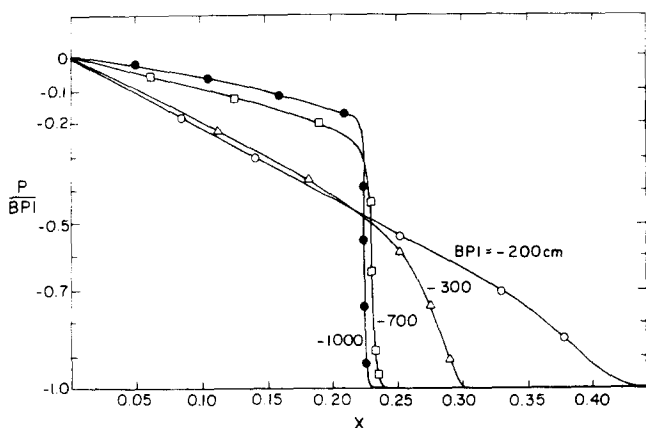
Figure 6. Similarity solution of the non-linear equation at t = 0.01.

A numerical solution is obtained by setting a time derivative on the left hand side, assuming an initial condition and placing the right hand boundary at a large distance instead of infinity. This equation is integrated to steady-state, which then is the solution to equation (20). A finite difference technique together with the GEARB package was used and the right hand boundary was placed at $\zeta = 6$. Integration to time $= 10$ was sufficient to reach steady-state.

The solutions are shown in Fig. 6 at $t = 0.01$ for four different initial soil dryness. As can be seen the steepest pressure profiles occur for the driest soils.

Eigenvalue analysis

Finlayson[7] investigated the effect of the initial dryness of the soil on the stiffness ratio of the eigenvalues. Here we look at the effect of various discretizations. Discretizing equation (16) using orthogonal collocation on finite elements gives a set of non-linear ordinary differential equations:

$$\sum_i C_{ji}(p) \frac{dp_i}{dt} = \sum_i D_{ji}(p)p_i \qquad (21)$$

The Jacobian for equation (21) can be expressed as:

$$J = \frac{\tilde{C}}{\tilde{C}_{P_R}} (C^{-1} \cdot D) \qquad (22)$$

and the eigenvalues of J can be found given a solution p and discretization at a certain time[26].

The results obtained at $t = 0.005$ and $t = 0.015$ for BP1 $= -300$ cm and BPO $= 5$ cm are given in Table 2 for various discretizations. At $t = 0.015$ the front is between $x = 0.20$ and $x = 0.30$. For uniform discretization the problem becomes stiffer as time proceeds, since the stiffness ratio SR becomes larger. If the system of equations is linear and uncoupled, the absolute value of the reciprocal of the eigenvalue is a time constant. For example

$$\varepsilon \frac{dy}{dt} = -y$$

has an eigenvalue of $-1/\varepsilon$ and a time constant of ε. If ε is small the eigenvalue is large. If we view the eigenvalues of equation (22) as the reciprocal of the time constants for the

nodes in the system, then if the time constant becomes small the eigenvalue becomes large. The time constant at a nodes is then $(-dS/dp_c)_i$. Since the boundary pressure is $+5$ cm for $p \to 0$ the value of $-dS/dp_c$ approaches zero, giving rise to small time constants or large eigenvalues. As time proceeds more and more nodes are in this situation, and the largest eigenvalue increases. The lowest eigenvalue is associated with nodes at other p where $-dS/dp_c$ takes reasonable values, and it remains constant since there are always nodes having p between BPO and BP1. Thus the stiffness ratio increases with time.

For the non-uniform discretizations we take the numerical solution at the appropriate $t(= 0.005$ or $0.015)$ obtained using a uniform grid $x = 0.05$. This solution is interpolated onto a new, non-uniform grid, and the eigenvalues of the Jacobian, equation (22), are calculated. In discretization no. 1 we place two additional elements between $x = 0.2$ and 0.3, and remove elements beyond $x = 0.3$ (where $p = BP1$ is a constant); the eigenvalues are unaffected. If elements are removed near $x = 0$, discretization no. 2, the maximum eigenvalue is reduced. Discretization no. 3 removes just one more of these elements and the stiffness ratio is reduced two orders of magnitude.

The change in discretization from 1 to 3 where nodes near $x = 0$ are removed gives rise to a reduction of five orders of magnitude in the stiffness ratio. In discretization no. 4 elements in the front are removed, but the stiffness ratio does not change.

In a moving coordinate system, we can concentrate nodes around the front and have few nodes elsewhere. Then the maximum absolute value of the eigenvalue will be lower and larger timesteps should then be possible.

Solution methods — fixed coordinate system

The standard finite difference technique to solve equation (16) uses a central difference approximation:

$$-\frac{dS}{dp_c}\bigg|_i \frac{dp_i}{dt} = \frac{1}{\Delta x}\left[k_{i+1/2}\frac{p_{i+1}-p_i}{\Delta x} - k_{i-1/2}\frac{p_i-p_{i-1}}{\Delta x} \right] \qquad (23)$$

The permeabilities at the half nodes can be evaluated in different ways[8]. Here we use the average permeability which corresponds to a second order method $O(h^2), k_{i+1/2} = 1/2(k_i + k_{i+1})$. A modified Crank–Nicholson method was used to integrate the equations in the form:

Table 2. Eigenvalues of non-linear problem, BP1 $= -300$ cm, BPO $= 5$ cm

Time	Discretization	Max. (EV)	Min. (EV)	$SR = \dfrac{\text{max.} (EV)}{\text{min.} (EV)}$
0.005	uniform $NE = 20$	6.9×10^7	0.16	4.3×10^8
0.015	uniform $NE = 20$	4.9×10^{10}	0.23	2.2×10^{11}
	non-uniform			
	$NE = 10$, No. 1	4.9×10^{10}	0.23	2.2×10^{11}
	$NE = 8$, No. 2	3×10^7	0.21	1.4×10^8
	$NE = 7$, No. 3	5×10^5	0.22	2.2×10^6
	$NE = 4$, No. 4	5×10^5	0.22	2.2×10^6

No. 1 disc	0.0 0.05	0.1 0.15	0.2 0.225	0.25	0.275	0.3 0.65	1
No. 2 disc:	0.0	0.1	0.2 0.225	0.25	0.275	0.3 0.65	1
No. 3 disc:	0.0		0.2 0.225	0.25	0.275	0.3 0.65	1
No. 4 disc.	0.0		0.2	0.25		0.3	1

Table 3. Comparison of MCS with conventional scheme (both using FD) for non-linear flow through porous media

Symbol in Fig. 7	Entry	Case	Scheme	NE	α	$\Delta t \times 10^{+5}$	Error (%)	CPU* (sec)
	1	I	CDA	21	0.5	20	6	2
	2		CDA	81	1	5	1	28
	3		MCS–CDA	23	1	20	6	2
	4		MCS-CDA	45	0.5	5	0.5	18
						$\Delta t \times 10^{+7}$		
7a O	5	II	CDA	120	1	2.5	33	60
7a □	6		CDA	600	1	2	26	402
7a △	7		CDA	600	1	1	12	804
7b □	8		MCS-FW	62	1	2.5/1.25	~0	57
7b O	9		MCS-FW	62	1	2.5	3	36
7b △	10		MCS-FW	42	1	2.5	22	24

* CPU on CDC 6400
CDA = central difference approximation; FW = CDA for second derivative, forward difference form for first derivative.

$$\sum_i C_{ji}(p)\frac{dp_i}{dt} = \sum_i D_{ji}(p)p_i$$

If we let $p_i^n = p_i$ $(t = n\Delta t)$ we used:

$$\sum_i C_{ji}(p^n)\frac{p_i^{n+1} - p_i^n}{\Delta t} = \sum_i D_{ji}(p^n)[(1-\alpha)p_i^n + \alpha p_i^{n+1}]$$

All function evaluations are explicit and no iterations are performed, in contrast to a regular Crank–Nicholson method.

Solution method — moving coordinate system

We apply a moving coordinate system to equation (16) as described previously and obtain:

$$-\frac{dS}{dp_c}\frac{\partial p}{\partial \eta} = -\lambda(\eta)\frac{1}{2}\frac{\partial S}{\partial p_c}\cdot\frac{\partial p}{\partial \xi} + \frac{\partial}{\partial \xi}\left(k_r\frac{\partial p}{\partial \xi}\right) \qquad (24)$$

the boundary conditions are now

$$p = BP0 \text{ at } \xi = -\int_0^\eta \lambda(t')dt'$$

$$\eta > 0$$

$$\frac{dp}{d\xi} = 0 \text{ at } \xi = 1 - \int_0^\eta \lambda(t')dt' \qquad (25)$$

$$p = BP1 \text{ for } \xi > 0 \text{ and } \eta = 0$$

We discretize equation (24) in a finite difference method by letting ξ_i be the location of the ith node, p_i the pressure at the ith node, and $h_i = \xi_{i+1} - \xi_i$. The derivatives are taken as:

$$\frac{\partial}{\partial \xi}\left(k_r\frac{\partial p}{\partial \xi}\right)\bigg|_i = \frac{2}{h_i + h_{i-1}}[k_r^{i+1/2}(p_{i+1} - p_i)/h_i - k_r^{i-1/2}(p_i$$

$$- p_{i-1})/h_{i-1}]$$

$$\frac{\partial p}{\partial \xi}\bigg|_i \frac{p_{i+1} - p_{i-1}}{h_i + h_{i-1}}$$

Blottner[1] shows these formulae are correct to $O(h_i^2)$ provided the mesh is graded such that:

$$\xi_{i+1} - 2\xi_i + \xi_{i-1} = O(h_i^2)$$

A forward difference can also be used for the first derivative since $dS/dp_c \leqslant 0$.

$$\frac{\partial p}{\partial \xi}\bigg|_i = (p_{i+1} - p_i)/h_i$$

This formula is correct to $O(h_i)$.

To determine the velocity of the front, the largest gradient at any time is calculated. When the position at which this occurs has moved from one node to another, the coordinate system is moved the same distance so as to keep the largest gradient fixed at the origin. From the discrete values of the velocity an integrated velocity can be calculated in order to set the boundary conditions.

It was found important to set the boundary conditions exactly by changing the prefixed node location rather than simply using the nearest points as in the linear case. Thus the end elements are adjusted so that a node exists at the ξ given in equation (25).

Results

The two methods are compared in two cases:

(I)   BP0 = 5 cm   BP1 = −300 cm

(II)   BP0 = 0 cm   BP1 = −1000 cm

For case (I) the front is not too steep and the efficiency of the two methods is almost the same (see Table 3). The error is defined as the maximum deviation from the similarity solution at any node in the domain. The regular finite difference scheme with 21 nodes (run 1) gives a 6% error but to get a more accurate solution many more nodes are needed (81 for run 2) and this requires a smaller timestep. For the moving coordinate system (MCS) fewer nodes (run 4) achieve the same error. MCS is more efficient by a small factor.

Figure 7 shows results for case (II) (steeper front) using both schemes and the advantage of MCS becomes significant. Here the error is defined as percentage misplacement of front (at $p = BP1/2$). With the regular finite difference scheme and 120 nodes (run 5), the velocity of the front is too high, and taking smaller timesteps did not affect the solution. Run 6 with 600 nodes is closer to the exact solution and if Δt is decreased in run 7 the profile moves slowly towards the solution (see Fig. 7a).

In contrast MCS with only 62 nodes (run 8) and a larger timestep gives almost the correct solution (see Fig. 7b). A similar solution was achieved using both fewer nodes and larger timestep (run 10) but then smoothing occurs. By only taking larger timesteps the front is placed incorrectly (run 9). The forward difference compared to the central difference approximation for the first derivative was found to place the front more correctly but at the cost of some smoothing, and the forward difference is therefore preferred.

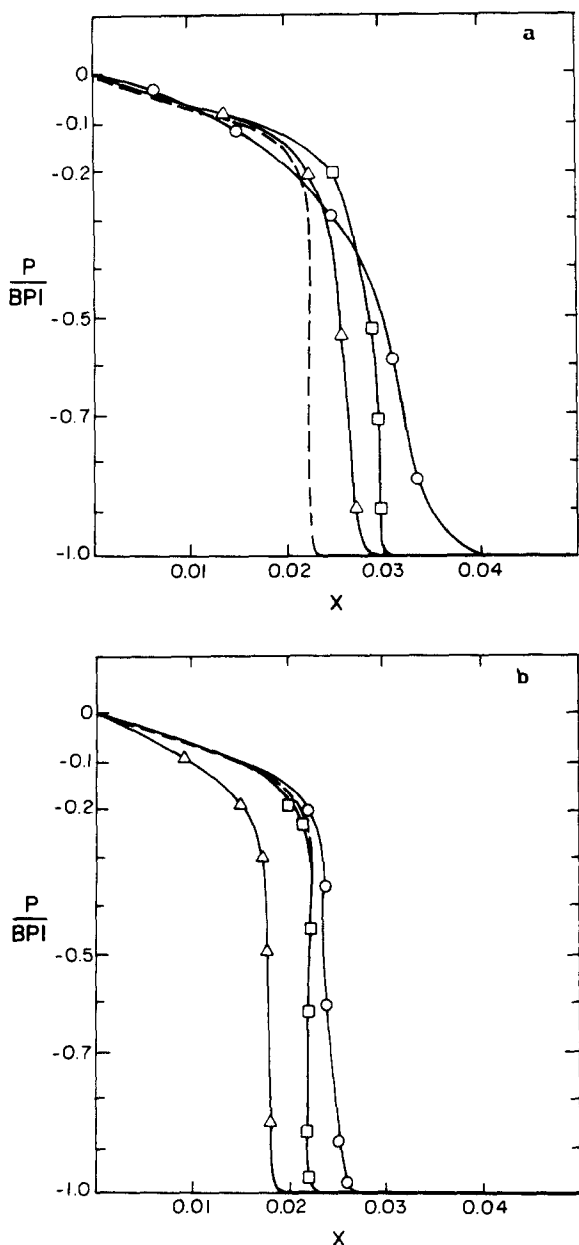An example of the distribution of nodes for run 8 to 10 is

Table 4. Distribution of nodes for run 8 to 10 (MCS)

Node	No. of grid points between	
	Run 8, 9	Run 10
−1	1	1
−0.5	2	2
−0.1	4	2
−0.02	6	3
−0.001	5	5
−0.0001	10	5
0	10	5
0.00005	10	7
0.0001	10	5
0.01	2	2
0.02	4	2
0.1	2	2
1		

(flow in porous media) develops a very steep front. The moving coordinate system can be applied to this case with a time dependent velocity. MCS is more efficient for problems with steep profiles. Computer time could be reduced by a factor of forty in one case.

ACKNOWLEDGEMENT

REFERENCES

1   Blottner, F. G. Variable grid scheme applied to turbulent boundary layers, *Comp. Meth. Appl. Mech. Eng.* 1974, 4, 179
2   Carey, G. F. and Finlayson, B. A. Orthogonal collocation on finite elements, *Chem. Eng. Sci.* 1975, 30, 587
3   Carslaw, H. S. and Jaeger, J. C. *Conduction of heat in solids*, Oxford Univ. Press, London, 1959
4   Chaudhari, N. M., An improved numerical technique for solving multidimensional miscible displacement, *Soc. Pet. Eng. J.* 1971, 11, 277
5   Chien, T. C. A general finite difference formulation for Navier-Stokes equation, *Comp. Fluids* 1977, 5, 15
6   Christie, I., Griffiths, D. F., Mitchell, A. R. and Zienkiewicz, O. C. Finite element methods for second order differential equations with significant first derivatives, *Int. J. Num. Meth. Eng.* 1976, 10, 1389
7   Finlayson, B. A. Water movement in desiccated soils, *Proc. 1st Int. Conf. Finite Elements in Water Resources*, Princeton 1976 (Ed. W. G. Gray, G. F. Pinder and C. A. Brebbia), pp. 3.91-3.106.
8   Finlayson, B. A. and Nelson, R. W. A preliminary investigation into the theory and techniques of modeling the natural moisture movement in unsaturated sediments. *Report BCSR*-40, Boeing Computer Services, Richland, WA, Sept. 1977
9   Garder, A. O., Jr., Peaceman, D. W. and Pozzi, A. L., Jr., Numerical calculation of multi-dimensional miscible displacement by method of characteristics, *Soc. Pet. Eng. J.* 1964, 4, 26
10   Gresho, P. M., Lee, R. S. and Sani, R. S. Advection-dominated flows, with emphasis on the consequences of mass lumping, *Prepr. Second Int. Symp. Finite Element Methods in Flow Problems*, Italy June, 1976, pp. 743-756
11   Huyakorn, P. S. An upwind finite element scheme for improved solution of the convection-diffusion equation, *Report no.* 76-WR-2, Water Resources Program, Princeton University, 1976
12   Jensen, O. K. and Finlayson, B. A. Oscillation limits for weighted residual methods, *Int. J. Num. Meth. Eng.* submitted
13   Jensen, O. K. Numerical modelling with a moving coordinate system application to flow through porous media, *Ph.D. Thesis*, University of Washington, 1980
14   Lantz, R. B., Quantitative evaluation of numerical diffusion, *Soc. Pet. Eng. J.* 1971, 11, 315
15   Laumbach, D. D. A high accuracy finite-difference technique for treating the convection-diffusion equation, *Soc. Pet. Eng. J.* 1975, 15, 517

Figure 7. Comparison, standard versus moving FD; nonlinear flow through porous media at $t = 1 \times 10^{-4}$. *Symbols defined in Table 3.* (a), *Fixed coordinate system,* (b), *MCS; —— , similarity solution (exact).*

shown in Table 4. The distance between nodes varies by a factor of 5×10^4.

The gain in efficiency achieved by using MCS is highest for the very steep front. In case (II), if we compare run 7 and run 10, we use about 15 times fewer nodes and timesteps about 3 times greater to achieve nearly the same accuracy. MCS is thus about 40 times faster than the same method on a fixed grid, or uses only 2.5% of the computation time.

CONCLUSIONS

Solving convection–diffusion equations on a moving coordinate system (MCS) rather than a fixed coordinate system can give better solutions at less cost when the Peclet number is large. Often factors of 10 or higher in improved accuracy or reduced cost are possible.

The solution to a non-linear diffusion type equation

16 Marino. M. A. Numerical and analytical solutions of dispersion in finite. absorbing porous medium. *Water Res. Bull.* 1974, **10**, 81

17 Peaceman, D. W. and Rachford, H. H. Jr., Numerical calculation of multi-dimensional miscible displacement. *Soc. Pet. Eng. J.* 1962, **2**, 327

18 Pinder, G. F. and Gray, W. G. *Finite element simulation in surface and sub-surface hydrology*. Academic Press, New York, 1977

19 Price, H. S., Cavendish, J. C. and Varga, R. S. Numerical methods of higher-order accuracy for diffusion-convection equation. *Soc. Pet. Eng. J.* 1968, **8**, 293

20 Price, H. S., Varga, R. S. and Warren, J. E. Application of oscillation matrices to diffusion-convection equations. *J. Math. Phys.* 1966, **45**, 301

21 Rachford, H. H. Numerical calculation of immiscible displacement by a moving reference point method. *Soc. Pet. Eng. J.* 1966, **6**, 87

22 Runchal, A. K. Comparative criteria for finite-difference formulations for problems of fluid flow, *Int. J. Num. Meth. Eng.* 1977, **11**, 1667

23 Smith, I. M. Integration in time of diffusion and diffusion-convection equations. *Proc. 1st Int. Conf. Finite Elements in Water Resources, Princeton* 1976, (eds. W. G. Gray, G. F. Pinder and C. A. Brebbia) pp. 1.3-1.20

24 Stone, H. L. and Brian, P. L. T. Numerical solution of convective transport problems, *AIChE J.* 1963, **9**, 681

25 VanGenuchten, M. T. On the accuracy and efficiency of several numerical schemes for solving convective–dispersive equation, *Proc. 1st Int. Conf. Finite Elements in Water Resources, Princeton* 1976, (eds. W. G. Gray, G. F. Pinder and C. A. Brebbia), pp. 1.71-1.90

26 Wilkinson, J. H. *The algebraic eigenvalue problem*. Clarendon Press, Oxford, 1965.

27 Young, L. C. A preliminary comparison of finite element methods for reservoir simulation, *Adv. Comp. Methods Partial Differential Equations* 1977, pp. 307-320

NOMENCLATURE

A, B, C*, D* (modified) collocation matrices
A1, B1   constants for description of S, k_r

BP0	boundary pressure
BP1	initial pressure
c	dimensionless concentration
D	diffusion coefficient
E	error
EV	eigenvalue
g	acceleration of gravity
h	spatial increment
$IJ; k, i, j$	index space discretization
J	Jacobian
k, k_r	(relative) permeability
L	length of domain
n	index
NP	number of interior collocation point plus two
NT	total number of points
p, p_c	(capillary) pressure
Pe	Peclet number
q	flux
S, S_r	(residual) saturation
x, y, ξ	dimensionless spatial length
t, η	dimensionless time
$\propto$	Crank–Nicholson parameter
β	functional parameter
ε	time constant
λ	frontal velocity
λ_m	maximum eigenvalue
μ	viscosity
ξ	similarity variable
ρ	density
φ	porosity
ψ	difference equation function
Δ	increment