ELSEVIER

Comput. Methods Appl. Mech. Engrg. 127 (1995) 181–201

Computer methods in applied mechanics and engineering

An operator splitting algorithm for coupled one-dimensional advection–diffusion–reaction equations

Liaqat Ali Khan*, Philip L.-F. Liu

School of Civil and Environmental Engineering, Cornell University, Ithaca, NY 14853, USA

Received 31 May 1994; revised 20 October 1994

Abstract

An operator splitting algorithm for a system of one-dimensional advection–diffusion–reaction equations, describing the transport of non-conservative pollutants, is presented in this paper. The algorithm is a Strang type splitting procedure incorporating contributions from the inhomogeneous terms by the Duhamel's principle. The associated homogeneous equations are split into advection, diffusion and reaction equations, and solved by a backward method of characteristic, a finite-element method and an explicit Runge–Kutta method, respectively. The boundary conditions applicable to the split equations are derived. Numerical analyses of the algorithm, consisting of the stability, the accuracy and the convergence of the solution procedure, are presented. The composite algorithm is second-order accurate in time and space and conditionally stable. The numerical characteristics of the algorithm are demonstrated by several examples.

1. Introduction

An operator splitting algorithm for the numerical solution of a coupled system of advection–diffusion–reaction equations, governing the transport of non-conservative pollutants in porous media, is presented in this paper. The primary objectives of the paper are to describe the splitting algorithm, and to analyze the characteristics of the numerical procedure. Therefore, to simplify the analyses and the presentation, a one-dimensional system of equations is considered. The system of equations can be expressed as [1]

$$\partial_t c_i = \mathscr{L} c_i + \mu_i c_{i-1}\,, \quad i = 1, \ldots, I\,, \tag{1}$$

where $c_i(t, x)$ is the concentration of the ith. pollutant species, I is the number of species in the transport process, $\mu_i(t, x) = R_{i-1}\lambda_{i-1}/R_i$, $\lambda_i(t, x)$ and $R_i(t, x)$ are the first-order reaction/decay and the retardation coefficients, respectively. In Eq. (1), $\mu_1 \equiv 0$ and $\mathscr{L}$ is an operator such that

$$\mathscr{L} = \sum_{j=1}^{J} \mathscr{L}_j\,, \tag{2}$$

where $J = 3$,

$$\mathscr{L}_1 = -\frac{u}{R_i}\partial_x\,, \qquad \mathscr{L}_2 = \frac{1}{R_i}\partial_x(D\,\partial_x)\,, \qquad \mathscr{L}_3 = -\lambda_i\,, \quad \forall\, i\,, \tag{3}$$

* Corresponding author. Present address: 1 Lethbridge Plaza, HydroQual, Inc., Mahwah, NJ 07430, USA.

SSDI 0045-7825(95)00839-X

the variable x and t are the distance and time coordinates, and $u(t,x)$ and $D(t,x)$ are the specific discharge (Darcy's velocity) and the diffusion coefficient, respectively.

The solutions of the governing equations are sought in a domain Ω, subjected to the following initial conditions

$$c_i(t,x) = f_i^0(x), \qquad t=0, x\in\Omega, \quad \forall i, \tag{4}$$

and boundary conditions

$$c_i(t,x) = f_i^1(t), \quad t>0, x\in\Gamma_1, \quad \forall i, \tag{5}$$

$$-\frac{uc_i}{R_i} + \frac{D}{R_i}\partial_x c_i = f_i^2(t), \qquad t>0, x\in\Gamma_2, \quad \forall i, \tag{6}$$

where Γ_1 and Γ_2 are the boundaries of Ω, and f_i^0, f_i^1 and f_i^2 are known functions.

The coupled system of equations (1) results from sequential reaction kinetics when the products of one reaction are the reactants of the following reaction. Such sequential reactions are applicable to a large class of environmental pollutant transport problems [2]. For example, the decay of radionuclides or the deficit of oxygen in a stream caused by organic pollutants [2]. The growth, decay and transport of microorganisms in the subsurface result in similar equations [3, 4]. The chemical transformations of organic contaminants in the subsurface are frequently approximated by sequential reaction kinetics. Additional examples of groundwater contaminant transport problems governed by the system of equations(1) can be found in McNab and Narasimhan [5].

In the last two decades, a large number of operator splitting algorithms has been reported for the solution of the advection–diffusion equation for conservative pollutant ($I=1$ and $\mu_1=0$). However, the number of splitting algorithms for the advection–diffusion–reaction equations is quite limited. A frequently cited splitting algorithm for a system of advection–diffusion–reaction equations is by Wheeler and Dawson [3] and Dawson and Wheeler [4]. Their algorithm is based on the following split equations

$$\partial_t c_i = (\mathscr{L}_1 + \mathscr{L}_2)c_i, \qquad \partial_t c_i = \mathscr{L}_3 c_i + \mu_i c_{i-1}, \quad i=1,\ldots,I, \tag{7}$$

where the operator $\mathscr{L}_3$ is a non-linear function of the dependent variables, and $\mu_i \neq 0$ for the microbial mass transport equation. The above split equations are solved by a characteristic finite-element method and a Runge–Kutta method, respectively. The algorithm has been used by Wheeler et al. [6] to study biodegradation of hydrocarbons in groundwater. Rifai and Bedient [7], Kinzelbach et al. [8] and Miller and Rabideau [9] have adopted a similar splitting scheme, but having used different numerical procedures for the split equations (7). A random walk method has also been utilized by Tompson and Dougherty [10] to solve the split equations (7).

Recently, Valocchi and Malmstead [11] have presented analyses of a first-order and a second-order accurate splitting algorithms for the split equations (7) with $I=1$, $\mu_1=0$ and constant u, D, λ_1 and R_1. The accuracy of the two algorithms differs by an order of magnitude only if u, D, λ_1 and/or R_1 are functions of x and/or $c_1(t,x)$. 'A variant of the normal operator-splitting algorithm' proposed by Valocchi and Malmstead [11], and latter adopted by Miller and Rabideau [9], is an operator splitting algorithm introduced by Strang [12] in 1968. The conclusions of their analyses [11] are (i) there is an error inherent in the splitting algorithms of the class represented by Wheeler and Dawson [3], and (ii) the performance of the second-order accurate splitting algorithm is significantly better than the first-order accurate algorithm. However, these conclusions are contrary to the well known results for constant coefficient linear equations [13]. The equivalence of the two splitting algorithms analyzed by Valocchi and Malmstead [11] is shown latter in this paper.

A different class of operator splitting algorithm is described by Ding and Liu [16, 17]. The advection–diffusion equations are further split and Eqs. (1) are replaced by

$$\partial_t c_i = \mathscr{L}_1 c_i, \qquad \partial_t c_i = \mathscr{L}_2 c_i, \qquad \partial_t c_i = \mathscr{L}_3 c_i + \mu_i c_{i-1}, \quad i=1,\ldots,I. \tag{8}$$

An important advantage of this type of splitting is that the hyperbolic and the parabolic differential operators are separated, and appropriate numerical procedures can be used for the split equations.

Ding and Liu [16, 17] have used a backward method of characteristic, a finite element-method and a semi-analytic procedure to solve the advection, the diffusion and the reaction equations, respectively. The algorithm for the advection equation is based on a solution procedure similar to Holly and Preissmann [18]. Unlike the algorithm by Wheeler and Dawson, the latter algorithm has not been widely used, and a comprehensive numerical analysis of the algorithm for the system of equations (1) is also lacking.

To adopt the theory of operator splitting algorithms presented in Strang [12], LeVeque and Oliger [13], Yanenko [14] and Marchuk [15] for the system of equations (1), the operators $\mathscr{L}$ and $\mathscr{L}_i, i = 1, \ldots, I$, are defined at time $t = t_{n+1/2}$ in a computational time step, where n is a time step counter, $t_n = nk$ and $k = t_{n+1} - t_n$ is the time step. This assumption, used in the rest of the paper, implies that in each time step $\mathscr{L}$ and $\mathscr{L}_i$'s are associated with known values of $u(t_{n+1/2}, x)$, $D(t_{n+1/2}, x)$, $R_i(t_{n+1/2}, x)$ and $\lambda_i(t_{n+1/2}, x)$. The operators $\mathscr{L}$ and $\mathscr{L}_i$ are locally constant in time, thus allowing the integration of Eq. (1) with respect to t and presenting the mathematical basis of splitting the advection–diffusion–reaction equation. The resulting algorithm with $\mu_i = 0$ is consistent with the theory of operator splitting–the sequence in which the split equations should be solved to maintain a given order of accuracy in time, stability of the composite algorithm, and boundary conditions for the split equations – presented in the previous studies [13, 14].

With the exception of $i = 1$, Eqs. (1) are inhomogeneous equations. Therefore, by the Duhamel's principle [19, 20], the solutions of Eqs. (1) over a time step, k, can be expressed as

$$c_i(t_{n+1}, x) = c_{i,h}(t_{n+1}, x) + c_{i,p}(t_{n+1}, x) , \quad \forall i , \tag{9}$$

where

$$c_{i,h}(t_{n+1}, x) = \exp\{k\mathscr{L}\}c_i(t_n, x) , \tag{10}$$

$$c_{i,p}(t_{n+1}, x) = \int_{t_0'}^{t'} \mu_i \exp\{(t' - \tau)\mathscr{L}\}c_{i-1}(\tau, x)\, d\tau , \tag{11}$$

where $t' = t_{n+1}$, $t_0' = t_n$ and $\exp\{k\mathscr{L}\}$ is a time evolution operator [21]. The first term on the right-hand side of Eq. (9) is a complementary function representing the solution of the following homogeneous equation

$$\partial_t c_{i,h} = \mathscr{L}c_{i,h} , \quad i = 1, \ldots, I . \tag{12}$$

The second term in Eq. (9) is the particular integral. For the system of equations (1), $c_{i,p}(t_{n+1}, x)$ is a known function of $c_{i-1}(t_{n+1}, x)$. In principle, all numerical procedures, including operator splitting algorithms, for the system of equations (1) should satisfy Eq. (9). However, operator splitting algorithms solving the split equations (7) or (8) sequentially [3, 4, 16, 17] do not satisfy this requirement unless $\mu_i \equiv 0$ for all i. Therefore, in this paper a second-order accurate operator splitting algorithm for the system of equations (1) is presented. The proposed algorithm satisfies Eq. (9) up to $\mathcal{O}(k^2)$.

A disadvantage [22, 23] of operator splitting algorithms is that the boundary conditions specified for the governing equations are not applicable to the split equations. In Eq. (1), advection, diffusion and reaction are simultaneous processes. The effects of these simultaneous processes are also reflected in the boundary conditions (5) and (6). However, in the splitting algorithms, these processes are applied sequentially to the initial condition to approximate the solution of the governing equation (see Section 2). As a result, when using operator splitting algorithms, it is necessary to derive boundary conditions applicable to the split equations. In this paper, second-order accurate Dirichlet boundary conditions for the split equations are presented.

The review of splitting algorithms presented in this paper has been confined to operator splitting algorithms for the advection–diffusion–reaction equations only. However, some of the issues addressed in the paper (i) construction of a second-order accurate algorithm for a homogeneous equation and (ii) boundary conditions for the split equations are common to operator splitting for the advection–diffusion equation and the Navier–Stokes equations. As pointed out by Demkowicz et al. [22], 'most of the existing methods in the literature are based, however, on first-order splitting'. Moreover, 'little work has been done to study the effects of boundary conditions on the performance and accuracy of

splitting methods' [22]. A recent statement by Perot [23] that 'in general, the method is first-order accurate in time, and serious confusion and/or disagreement concerning boundary conditions and the details of the methods implementation exists' further reinforces the observations made by Demkowicz et al. [22].

The rest of this paper is arranged as follows; The basic theory of operator splitting algorithm is presented in Section 2. The numerical procedures for the split equations are described in Section 3. The boundary conditions applicable to the split equations are derived in Section 4. Errors introduced by splitting the governing equations, and the use of numerical procedures are analyzed in Section 5. Stability analysis of the composite algorithm is presented in Section 6. Several numerical experiments to demonstrate the simulation accuracy and the numerical characteristics of the algorithm are described in Section 7. The summary and conclusions of the paper are presented in Section 8.

2. Theory of operator splitting algorithms

The theory of operator splitting algorithms for homogeneous differential equations can be found in Marchuk [15] and Yanenko [14]. In this paper the theory of Strang type [12] splitting algorithm, as analyzed by LeVeque and Oliger [13] for hyperbolic equations, is utilized to split the system of homogeneous equations (2). The composite solution procedure consists of solving the homogeneous equation (12) by an operator splitting algorithm and the numerical evaluations of the integrals (11). As noted in the previous section, the operators $\mathscr{L}$ and $\mathscr{L}_i$, $i = 1, \ldots, I$, are locally constant in time.

The solution of Eq. (12) as represented by Eq. (10) can be expressed as

$$c_{i,h}(t_{n+1}, x) = \exp\left\{k \sum_{j=1}^{J} \mathscr{L}_j\right\} c_{i,h}(t_n, x) \approx \prod_{j=1}^{J} \exp\{k\mathscr{L}_j\} c_{i,h}(t_n, x) , \quad \forall i . \tag{13}$$

If the operators $\mathscr{L}_j$'s commute, i.e. $\mathscr{L}_j\mathscr{L}_l = \mathscr{L}_l\mathscr{L}_j$ for $j \neq l$, then no error is committed in replacing the sum of the exponential operators by their product [13], and the approximation in Eq. (13) should be replaced by an equality. However, the operators $\mathscr{L}_j$'s commute only if u, D, λ_i and R_i are all constants. Otherwise, Eq. (13) forms the basis of an $\mathbb{O}(k)$ accurate splitting algorithm for Eq. (12), and an approximation, $\hat{c}_{i,h}(t, x)$, to $c_{i,h}(t, x)$ is obtained.

As noted by Strang [12], an $\mathbb{O}(k^2)$ accurate splitting algorithm can be formulated by making the solution procedure symmetric

$$\hat{c}_{i,h}(t_{n+1}, x) = \prod_{j=1}^{J-1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \exp\{k\mathscr{L}_J\} \prod_{j=J-1}^{1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \hat{c}_{i,h}(t_n, x) , \quad \forall i . \tag{14}$$

For $J = 3$, Eq. (14) implies the sequential solutions of the following split equations subjected to the indicated initial conditions and time steps

$$\partial_t \hat{c}^1_{i,h} = \mathscr{L}_1 \hat{c}^1_{i,h} , \qquad \hat{c}^1_{i,h}(t_n, x) = \hat{c}_{i,h}(t_n, x) , \quad t \in [t_n, t_{n+1/2}] , \tag{15}$$

$$\partial_t \hat{c}^2_{i,h} = \mathscr{L}_2 \hat{c}^2_{i,h} , \qquad \hat{c}^2_{i,h}(t_n, x) = \hat{c}^1_{i,h}(t_{n+1/2}, x) , \quad t \in [t_n, t_{n+1/2}] , \tag{16}$$

$$\partial_t \hat{c}^3_{i,h} = \mathscr{L}_3 \hat{c}^3_{i,h} , \qquad \hat{c}^3_{i,h}(t_n, x) = \hat{c}^2_{i,h}(t_{n+1/2}, x) , \quad t \in [t_n, t_{n+1}] , \tag{17}$$

$$\partial_t \hat{c}^2_{i,h} = \mathscr{L}_2 \hat{c}^2_{i,h} , \qquad \hat{c}^2_{i,h}(t_{n+1/2}, x) = \hat{c}^3_{i,h}(t_{n+1}, x) , \quad t \in [t_{n+1/2}, t_{n+1}] , \tag{18}$$

$$\partial_t \hat{c}^1_{i,h} = \mathscr{L}_1 \hat{c}^1_{i,h} , \qquad \hat{c}^1_{i,h}(t_{n+1/2}, x) = \hat{c}^2_{i,h}(t_{n+1}, x) , \quad t \in [t_{n+1/2}, t_{n+1}] , \tag{19}$$

and set $\hat{c}_{i,h}(t_{n+1}, x) = \hat{c}^1_{i,h}(t_{n+1}, x)$ for $i = 1, 2, \ldots, I$. In the above equations, $\hat{c}^j_{i,h}(t, x)$, $j = 1, \ldots, J$, are some intermediate values of $\hat{c}_{i,h}(t, x)$ obtained from the solutions of the split equations. The dependent variable $\hat{c}_{i,h}(t, x)$ as approximated by Eqs. (15)–(19) is defined only at the beginning and the end of a computational cycle: $\hat{c}^1_{i,h}(t_n, x) = \hat{c}_{i,h}(t_n, x)$, $\hat{c}^1_{i,h}(t_{n+1}, x) \approx \hat{c}_{i,h}(t_{n+1}, x)$, and $\hat{c}^j_{i,h}(t, x) \neq \hat{c}_{i,h}(t, x)$ for $j = 2, \ldots, J-1$ and $t \in [t_n, t_{n+1}]$.

Frequently, a transport process involves sub-problems of widely different times scales. For example, the problem may be advection dominated or the pollutant species may be highly reactive. The time scales of the advection, the diffusion and the reaction processes may be significantly different. The fastest process will determine the time step, k, that can be used for the numerical implementation of the algorithm (14). However, splitting algorithms can be designed so that appropriate time step can be used for each of the split equations. In this paper the use of a smaller time step, known as sub-cycling, for the reaction equation is presented. Let $\bar{k}$ be an appropriate time step for the numerical integration of the reaction equation, $\partial_t \hat{c}_{i,h} = \mathscr{L}_J \hat{c}_{i,h}$, with $m\bar{k} = k$, where $m \geqslant 1$ is a positive integer. Then algorithm (14) should be replaced by

$$\hat{c}_{i,h}(t_{n+1}, x) = \prod_{j=1}^{J-1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} [\exp\{\bar{k}\mathscr{L}_J\}]^m \prod_{j=J-1}^{1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \hat{c}_{i,h}(t_n, x), \quad \forall i, \tag{20}$$

and the solution step represented by Eq. (17) will consist of a sequence of m-step procedure. A similar approach can be used if the sub-cycling of the other transport processes are necessary.

Eqs. (13) and (14) indicate how the solutions of the split equations are advanced in a time step. The splitting procedure of the type described by Ding and Liu [17] corresponds to algorithm (13), and therefore it is only $\mathbb{O}(k)$ accurate. However, if $J = 2$ with $\mathscr{L}_1$ representing the combined advection–diffusion operator, $\mathscr{L}_2 = -\lambda_i$ and $\mu_i \equiv 0$, then both the algorithms (13) and (14) are essentially equivalent for n time steps. Let $nk = T$, where T is the simulation period. Therefore, algorithm (14) advances the solution of Eq. (12) from $t = 0$ to $t = T$ as

$$\hat{c}_{i,h}(T, x) = \left[\exp\left\{\frac{k}{2}\mathscr{L}_1\right\} \exp\{k\mathscr{L}_2\} \exp\left\{\frac{k}{2}\mathscr{L}_1\right\}\right]^n \hat{c}_{i,h}(0, x). \tag{21}$$

Eq. (21) follows [14, 21] from the properties of the semi-group formed by the exponential operators and the Huygens–Hadamard principle, i.e. successive solutions of a Cauchy problem with time steps $t_{n+1} - t_n = k$, $n = 0, 1, \ldots, T/k$, is equivalent to the solution of the Cauchy problem in the interval [0, T]. The previous equation can be rearranged to

$$\hat{c}_{i,h}(T, x) = \exp\left\{\frac{k}{2}\mathscr{L}_1\right\} [\exp\{k\mathscr{L}_2\} \exp\{k\mathscr{L}_1\}]^{n-1} \exp\{k\mathscr{L}_2\} \exp\left\{\frac{k}{2}\mathscr{L}_1\right\} \hat{c}_{i,h}(0, x), \tag{22}$$

by combining consecutive $\exp(k\mathscr{L}_1/2)$ operators. The expression within the square brackets, $[\cdot]$, represents the splitting algorithm (13) being applied $n - 1$ time steps. A comparison of Eq. (22) with algorithm (13), expressed for n time steps, indicates that algorithms (13) and (14) differ from each other only at the first and the last step of the computations. By the consistency of numerical procedures, the computational error per time step is generally small. Therefore, algorithm (13) behaves like an $\mathbb{O}(k^2)$ accurate procedure for a sufficiently large n such that $1 \ll n \ll k^{-2}$. Consequently, Eq. (22) indicates that the performances of the first- and the second-order splitting algorithms analyzed by Valocchi and Malmstead [11] should not be significantly different even for non-linear equations. For linear constant coefficient equations the algorithms are identical.

3. Numerical procedures

An important advantage of the operator splitting algorithm described in the previous section is that appropriate numerical procedures for different classes of partial differential equations can be combined to obtain the solution of the governing equation. This section describes a procedure for the numerical evaluations of the integrals (11), and the numerical solutions of the split equations (15)–(19). As the numerical procedures for the split advection and diffusion equations are quite well known, they are briefly described for the information necessary for error analyses in Section 5.

3.1. Advection equations

The advection equations (15) and (19) and can be written as

$$\frac{\mathrm{D}\hat{c}^1_{i,h}}{\mathrm{D}t} = 0\,, \quad i = 1, \ldots, I, \tag{23}$$

where $\mathrm{D}/\mathrm{D}t = \partial_t - \mathscr{L}_1$. Eq. (23) indicates that $\hat{c}^1_{i,h}(t, x)$ is invariant along the characteristic

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \tilde{u}_i\,, \quad \tilde{u}_i = \frac{u}{R_i}\,, \quad \forall\, i\,, \tag{24}$$

and its exact solution is

$$\hat{c}^1_{i,h}(t_{n+1}, x) = \hat{c}^1_{i,h}(t_n, x - X_i)\,, \quad \forall\, i\,, \tag{25}$$

where X_i, the foot of characteristic path, is the solution of Eq. (24) in a time step. However, the evaluation of $\hat{c}^1_{i,h}(t_n, x - X_i)$ requires a spatial interpolation as $x - X_i$ does not necessarily coincide with a computational node. Following Holly and Preissmann [18], a C^1 continuous Hermite interpolation function is used for the spatial interpolation of concentration. The numerical dispersion and dissipation associated with Hermite interpolation functions are significantly less than the corresponding Lagrangian interpolation functions [18]. As a result, simulation characteristics of operator splitting algorithms [17, 24–26] for the advection–diffusion equation (12), utilizing Hermite interpolation functions, are very encouraging for advection dominated transport problems.

As C^1 continuous Hermite interpolation function is of finite accuracy, an approximation, $\tilde{c}^1_{i,h}(t_{n+1}, x_l)$, to $\hat{c}^1_{i,h}(t_{n+1}, x_l)$, where $(t_n, x_l) = (nk, lh)$ and h is the nodal spacing, is obtained from the numerical solution of Eq. (23). For one-dimensional problems, the interpolation function [27] in local coordinate, η, can be expressed as

$$\tilde{c}^1_{i,h}(t_{n+1}, \eta) = \sum_{l=1}^{2} \{H_l \tilde{c}^1_{i,h,l} + H^\eta_l \partial_\eta \tilde{c}^1_{i,h,l}\}\,, \quad \forall\, i\,, \tag{26}$$

where

$$H_l(\eta, \eta_l) = -\frac{1}{4}\eta_l(\eta + \eta_l)^2(\eta - 2\eta_l)\,, \qquad H^\eta_l(\eta, \eta_l) = +\frac{1}{4}(\eta + \eta_l)^2(\eta - \eta_l)\,, \tag{27}$$

such that $\eta \in [-1, 1]$, l is the local node number, $\eta_l = +1$ or -1 is the local coordinate of a node, and by Eq. (25) $\tilde{c}^1_{i,h,l}$ and $\partial_\eta \tilde{c}^1_{i,h,l}$ are defined at $t = t_n$. The concentration gradient, $\partial_\eta \tilde{c}^1_{i,h,l}$, in Eq. (26) is determined by solving the following equation [17, 18]

$$\frac{\mathrm{D}\widehat{cx}^1_{i,h}}{\mathrm{D}t} = -\partial_x \tilde{u}_i \partial_x \hat{c}^1_{i,h}\,, \quad \forall\, i\,, \tag{28}$$

where $\widehat{cx}^1_{i,h} = \partial_x \hat{c}^1_{i,h}$. Eq. (28) is obtained by differentiating Eq. (23) with respect to x. In the present study, the computational domain is discretized by line elements and Eq. (24) is integrated backward in time by an explicit second-order accurate Runge–Kutta method.

3.2. Diffusion equations

The split diffusion equations (16) and (18) are discretized in space by a finite-element method, using linear basis functions, and the Crank–Nicolson finite-difference in time. The discretized system of equations can be expressed as

$$\{\tilde{c}^2_{i,h}(t_{n+1}, x_l)\} = [P_*]^{-1}[Q_*]\{\tilde{c}^2_{i,h}(t_n, x_l)\} + [P_*]^{-1}\{R\}\,, \quad \forall\, i\,, \tag{29}$$

where $\tilde{c}^2_{i,h}(t_{n+1}, x_l)$ is an approximation to $\hat{c}^2_{i,h}(t_{n+1}, x_l)$ and l is the global node number. In Eq. (29) $[P_*] = [P] + k[Q]/2$ and $[Q_*] = [P] - k[Q]/2$, where $[P]$ and $[Q]$ are the mass and the stiffness matrices, respectively, and $\{R\}$ represents boundary terms. Both $[P]$ and $[Q]$ are tri-diagonal matrices,

and the coefficients are function of cell Peclet number, $P_e = h^2/k\bar{D}$, where $\bar{D} = D/R_i$. Further details of the solution procedure can be found in Lapidus and Pinder [27].

The numerical procedure for the split diffusion equation is independent of the numerical procedure for the split advection equation. However, they share the same computational domain, and the dependent variables are related by the initial conditions specified in Eqs. (16) and (19). Similar observations are also applicable to the numerical schemes for integrating the reaction equation (17) and evaluating the integral in Eq. (11).

To proceed with the computations, it is necessary to determine the diffusion of concentration gradient [17, 18], $\widehat{cx}^{\,2}_{i,h}(t_{n+1}, x)$, corresponding to Eq. (28) in the advection step. In this study, instead of solving a diffusion type equation for $\widehat{cx}^{\,2}_{i,h}(t, x)$, numerical differentiation is used. The solution of diffusion equation provides information about $\tilde{c}^2_{i,h}(t_{n+1}, x)$. For the spatial interpolation of $\tilde{c}^2_{i,h}(t_{n+1}, x)$, a Hermite interpolation function is utilized. The interpolation function is similar to Eq. (26) with $\tilde{c}^1_{i,h}(t_{n+1}, x)$ replaced by $\tilde{c}^2_{i,h}(t_{n+1}, x)$ and the variables on the right-hand side are defined at t_{n+1}. The resulting interpolation function for spatial gradient of concentration can be expressed as

$$\widetilde{c\eta}^{\,2}_{i,h}(t_{n+1}, \eta) = \sum_{l=1}^{2} \left\{ \frac{\mathrm{d}H_l}{\mathrm{d}\eta}\, \tilde{c}^2_{i,h,l} + \frac{\mathrm{d}H^\eta_l}{\mathrm{d}\eta}\, \partial_\eta \tilde{c}^2_{i,h,l} \right\}, \quad \forall\, i\,, \tag{30}$$

where $\widetilde{c\eta}^{\,2}_{i,h}(t_{n+1}, \eta) = \partial_\eta \tilde{c}^2_{i,h}(t_{n+1}, \eta)$ and H_l and H^η_l are given by Eq. (27). The concentration gradient, $\partial_\eta \tilde{c}^2_{i,h,l}(t_{n+1}, x_l)$, in Eq. (30) is determined by assuming a quadratic variation of $\tilde{c}^2_{i,h}(t_{n+1}, x_l)$ over the adjacent elements. This method of determining $\widetilde{cx}^{\,2}_{i,h}(t_{n+1}, x_l)$ is similar to that described by Rasch and Williamson [28] for solving the advection equation using shape preserving interpolation functions. The use of Hermite interpolation functions with estimated derivatives is significantly more accurate than direct numerical differentiations [28]. The present solution procedure represents a modification of the splitting algorithms by Ding and Liu [17] and Holly and Preissmann [18]. The numerical scheme retains the accuracy of their algorithms, but avoids the need to solve diffusion equations for $\widetilde{cx}^{\,2}_{i,h}(t, x)$, $i = 1, 2, \ldots, I$.

3.3. Reaction equations

The reaction equation (17), which is an initial value problem, is integrated by an explicit second-order accurate Runge–Kutta method [29]. The resulting solution is a numerical approximation, $\tilde{c}^3_{i,h}(t_{n+1}, x_l)$, to $\hat{c}^3_{i,h}(t_{n+1}, x_l)$. The equation for concentration gradient $\widehat{cx}^{\,3}_{i,h}(t, x)$ is obtained by differentiating Eq. (17) with respect to x, and is solved by the same numerical procedure.

3.4. Evaluation of forcing terms

Using the trapezoidal integration rule, Eq. (11) can be approximated as

$$c_{i,p}(t_{n+1}, x) \approx \frac{k}{2}\, \mu_i [c_{i-1}(t_{n+1}, x) + \exp\{k\mathscr{L}\} c_{i-1}(t_n, x)]\,, \tag{31}$$

where μ_i is defined at $t_{n+1/2}$. Expanding the exponential in a Taylor series and retaining the first two terms, Eq. (31) can be expressed as

$$c_{i,p}(t_{n+1}, x) \approx \frac{k}{2}\, \mu_i \{c_{i-1}(t_{n+1}, x) + c_{i-1}(t_n, x)\} + \frac{k^2}{2}\, \mu_i \mathscr{L} c_{i-1}(t_n, x)\,, \tag{32}$$

which is an $\mathcal{O}(k^2)$ accurate approximation to the integral (11). The last expression in Eq. (32) is evaluated by assuming a quadratic variation of $c_{i-1}(t_n, x)$ in space, $x \in [x_{l-1}, x_{l+1}]$. Let $\tilde{c}_{i,p}(t_{n+1}, x_l)$ be the numerical approximation to $c_{i,p}(t_{n+1}, x)$, then Eq. (32) can be expressed as

$$\tilde{c}_{i,p}(t_{n+1}, x_l) = \frac{k}{2}\, \mu_i \{\tilde{c}_{i-1}(t_{n+1}, x_l) + \tilde{c}_{i-1}(t_n, x_l)\} + \frac{k^2}{2}\, \mu_i \mathscr{L} \tilde{c}^*_i(t_n, x_l; x)\,, \tag{33}$$

where

$$\tilde{c}_i^*(t_n, x_l; x) = \sum_{r=-1}^{1} \phi_r(x)\tilde{c}_{i-1}(t_n, x_{l+r}) . \tag{34}$$

The quadratic interpolation function $\phi_r(x)$ in local coordinate, η, is given by [27]

$$\phi_{-1}(\eta) = -\frac{1}{2}\eta(1-\eta) , \qquad \phi_0(\eta) = 1-\eta^2 , \qquad \phi_{+1}(\eta) = \frac{1}{2}\eta(1+\eta) , \tag{35}$$

where the global coordinates of the nodes corresponding to $\eta = -1, 0$ and $+1$ are x_{l-1}, x_l and x_{l+1}, respectively.

The equation for concentration gradient, $cx_i(t, x) = \partial_x c_i(t, x)$, can be obtained by differentiating Eq. (1) with respect to x. Let $cx_{i,p}(t, x)$ be the contribution of the inhomogeneous term to the time evolution of $cx_i(t, x)$, similar to Eq. (11). Then, $cx_{i,p}(t_{n+1}, x)$ is given by

$$cx_{i,p}(t_{n+1}, x) = \int_{t_0'}^{t'} \exp\{(t'-\tau)\mathscr{L}\}[\mathscr{L}_x c_i(\tau, x) + \partial_x \mu_i c_{i-1}(\tau, x) + \mu_i cx_{i-1}(\tau, x)]\, d\tau , \tag{36}$$

where $\mathscr{L}_x = \partial_x \mathscr{L}$. To $\mathbb{O}(k^2)$ accuracy, the integral is approximated as

$$\begin{aligned} cx_{i,p}(t_{n+1}, x) \approx{}& \frac{k}{2}[\mathscr{L}_x c_i(t_{n+1}, x) + \partial_x \mu_i c_{i-1}(t_{n+1}, x) + \mu_i cx_{i-1}(t_{n+1}, x)] \\ &+ \frac{k}{2}\exp\{k\mathscr{L}\}[\mathscr{L}_x c_i(t_n, x) + \partial_x \mu_i c_{i-1}(t_n, x) + \mu_i cx_{i-1}(t_n, x)] . \end{aligned} \tag{37}$$

The right-hand side of Eq. (37) is evaluated by a procedure similar to Eq. (33).

4. Boundary conditions

The boundary conditions (5) and (6) are defined for the governing equations (1). As $\hat{c}_{i,h}^j(t, x) \neq c_i(t, x)$, $j = 1, \ldots, J$, these boundary conditions are not directly applicable to the split equations (15)–(19). Several researchers have derived appropriate boundary conditions for operator splitting algorithms for hyperbolic [30, 31] and parabolic diffusion [32, 33] equations. However, no such attempt has been reported in the context of the advection–diffusion–reaction equations. Therefore, the intermediate Dirichlet boundary conditions applicable to the split equations are derived in this section.

Considering Eq. (15) as an initial value problem, the equation can be integrated over a time step $k/2$ to obtain

$$\hat{c}_{i,h}^1(t_{n+1/2}, x) = \exp\left\{\frac{k}{2}\mathscr{L}_1\right\}\hat{c}_{i,h}(t_n, x) , \tag{38}$$

and an $\mathbb{O}(k^2)$ accurate approximation to it is

$$\hat{c}_{i,h}^1(t_{n+1/2}, x) \approx \hat{c}_{i,h}(t_n, x) + \frac{k}{2}\mathscr{L}_1\hat{c}_{i,h}(t_n, x) + \frac{k^2}{8}\mathscr{L}_1^2\hat{c}_{i,h}(t_n, x) . \tag{39}$$

By Eqs. (9)–(11) $c_{i,p}(t_n, x) = 0$ at $t = t_n$, so that $\hat{c}_{i,h}(t_n, x) = c_i(t_n, x)$. Let $f_i^{1,1}(t_{n+1/2})$ be the appropriate boundary condition for Eq. (15) at $t_{n+1/2}$. As $f_i^{1,1}(t_{n+1/2}) = c_i(t_{n+1/2}, x_0)$, $x_0 \in \Gamma_1$, the previous equation can be expressed as

$$f_i^{1,1}(t_{n+1/2}) \approx c_i(t_n, x_0) + \frac{k}{2}\mathscr{L}_1 c_i(t_n, x)\Big|_{x_0} + \frac{k^2}{8}\mathscr{L}_1^2 c_i(t_n, x)\Big|_{x_0} . \tag{40}$$

Similarly, the boundary conditions for Eqs. (16) and (18) are

$$f_i^{1,2}(t_{n+1/2}) \approx \hat{c}_{i,h}^1(t_{n+1/2}, x_0) + \frac{k}{2}\mathscr{L}_2\hat{c}_{i,h}^1(t_{n+1/2}, x)\Big|_{x_0} + \frac{k^2}{8}\mathscr{L}_2^2\hat{c}_{i,h}^1(t_{n+1/2}, x)\Big|_{x_0} , \tag{41}$$

$$f_i^{1,2}(t_{n+1}) \approx \hat{c}_{i,h}^3(t_{n+1}, x_0) + \frac{k}{2}\mathscr{L}_2\hat{c}_{i,h}^3(t_{n+1}, x)\Big|_{x_0} + \frac{k^2}{8}\mathscr{L}_2^2\hat{c}_{i,h}^3(t_{n+1}, x)\Big|_{x_0} , \tag{42}$$

No boundary condition is necessary for the reaction equation (17). By the convergence of the operator splitting algorithm to the solution of Eq. (12) (see Section 5), $\hat{c}^{1}_{i,h}(t_{n+1}, x) \approx \hat{c}_{i,h}(t_{n+1}, x) \approx c_{i,h}(t_{n+1}, x)$. On the boundary Γ_1, $c_i(t_{n+1}, x_0) = f^1(t_{n+1})$. Therefore, Eq. (9) can be used to express the boundary condition for Eq. (19) as

$$f_i^{1,1}(t_{n+1}) \approx f_i^1(t_{n+1}) - c_{i,p}(t_{n+1}, x_0) . \tag{43}$$

In Eq. (40), $c_i(t_n, x_0) = f^1(t_n)$ and the second and third terms, containing $\partial_x c_i(t_n, x)$ and $\partial_{xx} c(t_n, x)$, are evaluated numerically using the Hermite interpolation function (26). As a C^1 continuous Hermite interpolation cannot provide fourth-order concentration derivative, a different approach is used for Eqs. (41) and (42). In Eq. (41), $\hat{c}^{1}_{i,h}$ and $\partial_x \hat{c}^{1}_{i,h} = \widehat{cx}^{1}_{i,h}$ are known from the solutions of Eqs. (23) and (28), respectively. Considering $\widehat{cx}^{1}_{i,h}$ as a dependent variable, a Hermite interpolation function, similar to Eq. (26), can be written as

$$\widetilde{c\eta}^{1}_{i,h}(t_{n+1}, \eta) = \sum_{l=1}^{2} \{H_l \widetilde{c\eta}^{1}_{i,h,l} + H_l^{\eta} \partial_{\eta} \widetilde{c\eta}^{1}_{i,h,l}\} , \quad \forall i , \tag{44}$$

where $\widetilde{c\eta}^{1}_{i,h}(t, \eta)$, corresponding to $\widetilde{cx}^{1}_{i,h}(t, x)$, is the concentration gradient in the local coordinate, and $\partial_{\eta} \widetilde{c\eta}^{1}_{i,h,l}$ is determined by assuming a quadratic variation of $\widetilde{cx}^{1}_{i,h}(t, x)$ over the elements adjacent to the boundary. The interpolation function (44) is differentiated to determine $\partial_{xxxx} \tilde{c}^{1}_{i,h}(t, x) = \partial_{xxx} \widetilde{c\eta}^{1}_{i,h}(t, x)$. The reaction step of computations provides information about $\hat{c}^{3}_{i,h}(t, x)$ and $\widehat{cx}^{3}_{i,h}(t, x)$, and the same numerical procedure is used for evaluating the second- and fourth-order concentration derivatives in Eq. (42). The right-hand side of Eq. (43) is a function of known quantities.

For the present algorithm, additional boundary conditions are necessary for the concentration gradient equations (28). Following Holly and Usseglio-Polatera [34], these boundary conditions can be expressed as

$$f_{i,x}^{1,1} = -\tilde{u}^{-1} \partial_t f_i^{1,1} , \quad \forall i , \tag{45}$$

where $f_{1,x}^{1,1} = \widehat{cx}^{1}_{i,h}(t, x_0)$, and $f_i^{1,1}$ is defined by Eqs. (40) and (43).

5. Error analysis

The total error in the proposed algorithm is the result of (i) spitting up the homogeneous differential equations, (ii) the approximations made in solving the split equations, and (iii) the numerical evaluations of the integrals. Let $\tilde{c}_i(t_{n+1}, x_l)$ be the numerical approximation obtained by the algorithm. Then, the error, $\epsilon_i(k, h)$, is given by

$$\epsilon_i(k, h) = c_i(t_{n+1}, x_l) - \tilde{c}_i(t_{n+1}, x_l) , \quad i = 1, \ldots, I . \tag{46}$$

Using Eq. (9) and its numerical approximation, $\epsilon_i(k, h)$ can be expressed as

$$\epsilon_i(k, h) = \{c_{i,h}(t_{n+1}, x_l) + c_{i,p}(t_{n+1}, x_l)\} - \{\tilde{c}_{i,h}(t_{n+1}, x_l) + \tilde{c}_{i,p}(t_{n+1}, x_l)\} , \tag{47}$$

which can be decomposed into

$$\epsilon_i(k, h) = \epsilon_i^{\mathrm{S}}(k) + \epsilon_i^{\mathrm{T}}(k, h) + \epsilon_i^{\mathrm{I}}(k, h) , \quad \forall i , \tag{48}$$

such that

$$\epsilon_i^{\mathrm{S}}(k) = c_{i,h}(t_{n+1}, x_l) - \hat{c}_{i,h}(t_{n+1}, x_l) = E_{\mathrm{S}}(k) c_{i,h}(t_n, x)|_{x=x_l} , \tag{49}$$

$$\epsilon_i^{\mathrm{T}}(k, h) = \hat{c}_{i,h}(t_{n+1}, x_l) - \tilde{c}_{i,h}(t_{n+1}, x_l) = E_{\mathrm{T}}(k, h) \hat{c}_{i,h}(t_n, x)|_{x=x_l} , \tag{50}$$

$$\epsilon_i^{\mathrm{I}}(k, h) = c_{i,p}(t_{n+1}, x_l) - \hat{c}_{i,p}(t_{n+1}, x_l) = E_{\mathrm{I}}(k, h) c_{i,p}(t_n, x)|_{x=x_l} , \tag{51}$$

for $i = 1, \ldots, I$, where $\epsilon_i^{\mathrm{S}}(k)$, $\epsilon_i^{\mathrm{T}}(k, h)$ and $\epsilon_i^{\mathrm{I}}(k, h)$ are the errors due to splitting the homogeneous

differential equation (12), the use of numerical procedures to solve Eqs. (15)–(19), and that resulting from the numerical evaluation of the integral (11), respectively. The associated error operators $E_S(k)$, $E_T(k,h)$ and $E_I(k,h)$ are defined below.

5.1. Splitting error operator

If $\mathscr{L}_j\mathscr{L}_l = \mathscr{L}_l\mathscr{L}_j$ for $j \neq l$, then $\epsilon_i^S(k)$ is zero [13]. Otherwise, the local splitting error is defined by Eq. (49). Using Eqs. (10) and (14), the splitting error operator, $E_S(k)$, can be expressed as [13]

$$E_S(k) = \exp\{k\mathscr{L}\} - \prod_{j=1}^{J-1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \exp\{k\mathscr{L}_J\} \prod_{i=J-1}^{1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\}. \tag{52}$$

The error operator can be determined by expanding the exponentials in a Taylor series. An explicit expression of $E_S(k)$ for $J=2$ can be found in LeVeque and Oliger [13]. For $J=3$, $E_S(k)$ consists of 24 terms, and therefore not shown. However, an analysis similar to LeVeque and Oliger [13] indicates that $\mathbb{O}(1)$, $\mathbb{O}(k)$ and $\mathbb{O}(k^2)$ terms are cancelled, and

$$\|E_S(k)\| = \left\|\exp\{k\mathscr{L}\} - \prod_{j=1}^{J-1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \exp\{k\mathscr{L}_J\} \prod_{j=J-1}^{1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\}\right\| = \mathbb{O}(k^3). \tag{53}$$

Therefore, the splitting error, $\epsilon_i^S(k)$, in algorithm (14) is of $\mathbb{O}(k^3)$ and the exact solution obtained by solving the split equations sequentially is $\mathbb{O}(k^2)$ accurate. Eq. (53) also implies that

$$\lim_{k\to 0} \|E_S(k)\| = 0, \tag{54}$$

indicating the convergence of the solution obtained by solving the split equations (15)–(19) to the solution of Eq. (12) for $i = 1, \ldots, I$.

5.2. Numerical composite and split operators

The numerical implementation of the operator splitting algorithm (14) for Eq. (12) introduces additional temporal and spatial errors. From Eq. (50), the exact composite solution for the homogeneous equation (12) can be expressed as

$$\hat{c}_{i,h}(t_{n+1}, x_l) = \tilde{c}_{i,h}(t_{n+1}, x_l) + E_T(k,h)\hat{c}_{i,h}(t_n, x)|_{x=x_l}, \quad \forall i, \tag{55}$$

where $E_T(k,h)$ is the truncation error operator of the composite algorithm. Let $\mathscr{S}(k,h)$ be the symbol of an equivalent numerical solution operator incorporating the three numerical procedures for Eqs. (15)–(19). Then, Eq. (55) can be written as

$$\hat{c}_{i,h}(t_{n+1}, x_l) = \{\mathscr{S}(k,h) + E_T(k,h)\}\hat{c}_{i,h}(t_n, x)|_{x=x_l}, \quad \forall i. \tag{56}$$

A comparison with Eq. (14) indicates the following equivalent relation

$$\prod_{j=1}^{J-1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} \exp\{k\mathscr{L}_J\} \prod_{i=J-1}^{1} \exp\left\{\frac{k}{2}\mathscr{L}_j\right\} = \mathscr{S}(k,h) + E_T(k,h). \tag{57}$$

Using Eqs. (49) and (50), the exact solution operator of Eq. (12), as expressed by Eq. (10), is also given by

$$\exp\{k\mathscr{L}\} = \mathscr{S}(k,h) + E_S(k) + E_T(k,h). \tag{58}$$

In a similar way, the following equivalent operators can be defined for the split equations (15)–(19)

$$\exp\{k\mathscr{L}_j\} = \mathscr{S}_j(k,h) + E_T^j(k,h), \quad \forall j, \tag{59}$$

where $\mathscr{S}_j(k,h)$ is the numerical solution operator of the jth split equation, and $E_T^j(k,h)$ is the

corresponding truncation or interpolation error operator. A comparison with Eqs. (26) and (29) for $j = 1$ and 2 (neglecting boundary terms) shows that

$$\mathcal{S}_1(k,h) = \sum_{l=1}^{2} \{H_l + H_l^x \partial_x\}\,, \qquad \mathcal{S}_2(k,h) = [P_*]^{-1}[Q_*]\,, \tag{60}$$

respectively. A similar equation can be obtained for $\mathcal{S}_3(k,h)$ representing the Runge–Kutta operator. A relationship between $E_{\mathrm{T}}^j(k,h)$'s and $E_{\mathrm{T}}(k,h)$ is established later in this section to define $E_{\mathrm{T}}(k,h)$.

5.3. Truncation error operators

The leading order truncation error operators, $E_{\mathrm{T}}^j(k,h)$, for the different numerical procedures are presented in this section. Assuming that the velocity field is a smooth function of time and space, Eq. (24) can be integrated exactly and the time discretization error during the advection step of computations is zero. An operator providing an upper bound on the spatial interpolation error for the one-dimensional Hermite interpolation function [35] can be expressed as

$$E_{\mathrm{T}}^1(k,h) = \frac{h^4}{384}\,\partial_{xxxx}\,. \tag{61}$$

Eq. (61) has been derived for local coordinate $\eta = 0$, which corresponds to a Courant number of half and maximum interpolation error [18]. This equation indicates that the backward method of characteristic, using the Hermite interpolation function, is $\mathcal{O}(h^3)$ accurate. The global accuracy [27] of the Crank–Nicolson finite-element method for the diffusion equation is $\mathcal{O}(k^2, h^2)$. The leading order truncation error operator, assuming $\bar{D} = D/R_i$ is a constant, is given by

$$E_{\mathrm{T}}^2(k,h) = k^3\partial_{ttt} + \frac{1}{2}\,\bar{D}kh^2\partial_{xxxx}\,. \tag{62}$$

Numerical analyses of the Runge–Kutta methods can be found in Hairer and Wanner [29]. Assuming that λ_i is constant, the truncation error operator associated with the explicit second-order Runge–Kutta method is

$$E_{\mathrm{T}}^3(k,h) = -\frac{\lambda_i k^3}{24}\,\{\partial_{tt} - 4\lambda_i\partial_t + \lambda_i^2\}\,. \tag{63}$$

If sub-cycling of the reaction equation is used, then k should be replaced by $\bar{k}$. Eqs. (61) to (63) indicate that

$$\lim_{k,h\to 0} \|E_{\mathrm{T}}^j(k,h)\| = 0\,, \quad \forall\, j\,. \tag{64}$$

Therefore, the numerical procedures for the split equations are consistent with the corresponding split differential equations. As the computational mesh is refined, the numerical solutions of Eqs. (15)–(19) converge to the exact solutions of the corresponding split differential equations.

5.4. Composite error operators

The numerical solution operator for the composite algorithm, introduced in Eq. (56), can be obtained by replacing the exponentials in Eq. (14) by the corresponding numerical solution operators

$$\mathcal{S}(k,h) = \prod_{j=1}^{J-1} \mathcal{S}_j\Big(\frac{k}{2},h\Big)\mathcal{S}_J(k,h) \prod_{j=J-1}^{1} \mathcal{S}_j\Big(\frac{k}{2},h\Big)\,. \tag{65}$$

Using Eq. (59), the composite numerical operator can be expressed as

$$\mathcal{S}(k,h) = \prod_{j=1}^{J-1} \Big[\exp\Big\{\frac{k}{2}\,\mathcal{L}_j\Big\} - E_{\mathrm{T}}^j\Big(\frac{k}{2},h\Big)\Big][\exp\{k\mathcal{L}_J\} - E_{\mathrm{T}}^J(k,h)] \prod_{j=J-1}^{1} \Big[\exp\Big\{\frac{k}{2}\,\mathcal{L}_j\Big\} - E_{\mathrm{T}}^j\Big(\frac{k}{2},h\Big)\Big]\,. \tag{66}$$

To the leading order, the composite error operator for algorithm (14) can be determined directly by adding Eqs. (49) and (50). This approach, however, does not indicate the relation between $E_T(k,h)$ and $E_T^j(k,h)$. Therefore, let $E_S^T(k,h)$ be the composite error operator for the homogeneous equation (12). Then, the total error operator for algorithm (14) is represented by the discrepancy [13] between Eq. (10) and the solution obtained by Eq. (65)

$$E_S^T(k,h) = \exp\{k\mathscr{L}\} - \prod_{j=1}^{J-1} \mathscr{S}_j\left(\frac{k}{2},h\right)\mathscr{S}_J(k,h) \prod_{j=J-1}^{1} \mathscr{S}_j\left(\frac{k}{2},h\right). \tag{67}$$

Using Eqs. (52), (57)–(59) and assuming that $\mathscr{S}_j(k,h)$'s are at least $\mathbb{O}(k^2)$ accurate, Eq. (67) can be expressed as

$$E_S^T(k,h) = E_S(k) + E_T(k,h) + \mathbb{O}(k^4), \tag{68}$$

where

$$E_T(k,h) = E_T^J(k,h) + 2\sum_{j=1}^{J-1} E_T^j\left(\frac{k}{2},h\right), \tag{69}$$

and $E_T^j(k,h)$'s are defined by Eqs. (61)–(63). If sub-cycling of the reaction equation is used, then $E_T^J(k,h)$ is given by

$$E_T^J(k,h) = mE_T^J(\bar{k},h) = mE_T^J\left(\frac{k}{m},h\right). \tag{70}$$

Eqs. (53), (61)–(63) can be combined to obtain

$$\|E_T^S(k,h)\| = \mathbb{O}(k^3, kh^2), \qquad \lim_{k,h\to 0} \|E_T^S(k,h)\| = 0, \quad \forall i, \tag{71}$$

indicating that the composite algorithm for Eq. (12) is second-order accurate in time and space.

5.5. Integration error operator

The error analyses presented until now are related to the solution of the homogeneous equation (12). The algorithm for Eq. (1) involves additional discretization error. Assuming μ_i is locally constant, the error operator associated with the numerical evaluation of the integral (11) can be expressed as

$$E_I(k,h) = \frac{k^3}{4}\mu_i\left[\frac{1}{2}\mathscr{L}^2 - \frac{1}{3}\exp\{k\mathscr{L}\}\,\partial_{tt}\right]. \tag{72}$$

The first term in the parenthesis is the leading order term in the Taylor series expansion of the exponential in Eq. (31), that has been neglected in arriving at Eq. (32). The second term is the leading order error term associated with the trapezoidal integration rule. By Eqs. (61)–(63), (68) and (69), the contribution from the first term is at least two order of magnitude smaller than the second term. Therefore, to the leading order

$$E_I(k,h) = -\frac{k^3}{12}\mu_i\partial_{tt}, \tag{73}$$

indicating

$$\lim_{k,h\to 0} \|E_I(k,h)\| = 0. \tag{74}$$

5.6. Accuracy and convergence

Based on Eqs. (48)–(51), (68) and (69), the leading order error in the numerical implementation of the operator splitting algorithm for Eq. (1) can be expressed as

$$\epsilon_i(k,h) = E_S^T(k,h)c_{i,h}(t_n,x) + E_I(k,h)c_{i,p}(t_n,x), \quad \forall i. \tag{75}$$

Combined with Eqs. (53), (61)–(63) and (73), Eq. (75) indicates that the splitting algorithm is $\mathcal{O}(k^2, h^2)$ accurate, and

$$|\epsilon_i(k,h)| \leq \|E_S^T(k,h)\| \, |c_{i,h}(t_n,x)| + \|E_1(k,h)\| \, |c_{i,p}(t_n,x)| \,, \quad \forall\, i \,. \tag{76}$$

By Eqs. (54), (71) and (74), Eq. (76) shows that

$$|\epsilon_i(k,h)| = \mathcal{O}(k^3, kh^2) \,, \qquad \lim_{k,h\to 0} |\epsilon_i(k,h)| = 0 \,, \quad \forall\, i \,. \tag{77}$$

Eq. (77) indicates the convergence of the operator splitting algorithm for Eq. (1) to its exact solution as the spatial and the temporal discretizations are refined.

6. Stability analysis

The stability analysis of the operator splitting algorithm for Eq. (1) is presented in this section. The analysis applies to a Cauchy problem, $x \in (-\infty, \infty)$, or a problem with periodic boundary conditions. In addition, u, D, R_i and λ_i are assumed constants. Since the well-posedness and the stability of an inhomogeneous differential equation is determined by the corresponding homogeneous equation [19, 20], it is sufficient to analyze the stability of the splitting algorithm for Eq. (12). Let $\mathcal{G}(k, \xi)$, where $\xi \in [0, \infty)$ is a Fourier wave number, be the amplification matrix of the composite algorithm. Then, the composite algorithm for Eq. (12) will be stable if

$$\|\mathcal{G}(k,\xi)\| \leq 1 + \mathcal{O}(k) \,, \quad \forall\, i \,. \tag{78}$$

From Eqs. (9) and (31), the amplification of the solution of Eq. (1) will be within a distance $\mathcal{O}(k)$ from that of Eq. (12). Therefore, the solution procedure for Eq. (1) will be stable provided the splitting algorithm for Eq. (12) is stable.

6.1. Composite algorithm

Let $\mathcal{G}_j(k, \xi)$, $j = 1, \ldots, J$, be the amplification matrices or factors for the split equations. From Eq. (13), the norms of these amplification matrices are related to $\mathcal{G}(k, \xi)$ in Eq. (78) by [15, 27]

$$\|\mathcal{G}(k,\xi)\| \leq \prod_{j=1}^{J} \|\mathcal{G}_j(k,\xi)\| \,. \tag{79}$$

The splitting algorithm (13), for each i, will be stable provided

$$|\rho(\mathcal{G}_j(k,\xi))| \leq 1 + \mathcal{O}(k) \,, \quad \forall\, j \,, \tag{80}$$

where $\rho(\mathcal{G}_j(k, \xi))$ be the spectral radius of $\mathcal{G}_j(k, \xi)$. The stability criterion for algorithm (13) provides the necessary condition for the stability of the splitting algorithm (14) [13]. This follows from the observation that if a consistent numerical procedure is stable for a time step k, then the algorithm will also be stable for a smaller time step, $k/2$.

The backward method of characteristics for the advection equation and the implicit finite element method for the diffusion equation are unconditionally stable [27]. The explicit Runge–Kutta method is conditionally stable [29]. Therefore, the splitting algorithm presented in this paper is conditionally stable. The primary objective of rest of this section is to characterize the numerical dispersion and dissipation associated with the algorithm. The discussions related to the Crank–Nicolson finite element method and the Runge–Kutta method are mainly for the completeness of analysis.

6.2. Advection of equations

For constant $\tilde{u} = u/R_i$, the right-hand side of Eq. (28) is zero. The Hermite interpolation function (26) couples Eqs. (23) and (28), and the amplification matrix, $\mathcal{G}_1(k, \xi)$, of the backward method of characteristics can be expressed as

$$\mathcal{G}_1(k, \xi) = \begin{bmatrix} \sum_l H_l \, e^{\hat{i}(\eta_l - 1)\xi} & \sum_l H_l^\eta \, e^{\hat{i}(\eta_l - 1)\xi} \\ \sum_l \dfrac{dH_l}{d\eta} e^{\hat{i}(\eta_l - 1)\xi} & \sum_l \dfrac{H_l^\eta}{d\eta} e^{\hat{i}(\eta_l - 1)\xi} \end{bmatrix}, \tag{81}$$

where $\hat{i} = \sqrt{-1}$. Let $|\rho^2| \leqslant |\rho^1|$ be the absolute eigenvalues of $\mathcal{G}_1(k, \xi)$ defined by $\det(\mathcal{G}_1 - \rho I) = 0$. Then ρ^1 and ρ^2 represent two different modes of transport as discussed by Holly and Preissmann [18]. The physical mode of transport corresponds to ρ^1, while ρ^2 is a parasitic mode. The analytic expressions of the eigenvalues are quite complicated and therefore not shown. However, as $\xi \to 0$, corresponding to the low frequency wave that are well resolved by the numerical procedure, the amplification matrix simplifies to

$$\mathcal{G}_1(k, 0) = \begin{bmatrix} 1 & \frac{1}{2}\eta(\eta^2 - 1) \\ 0 & \frac{1}{2}\eta(3\eta^2 - 1) \end{bmatrix}, \tag{82}$$

and the corresponding eigenvalues are

$$\rho^1 = 1\,, \qquad \rho^2 = \frac{1}{2}\eta(3\eta^2 - 1)\,. \tag{83}$$

As $\eta \in [-1, 1]$, both $|\rho^1|$ and $|\rho^2| \leqslant 1$ for $\xi \to 0$ or the wave length approaching infinity. In a similar way, as $\xi \to \infty$, waves poorly resolved by the numerical discretization, it can be shown [18] that

$$\rho^1 = \rho^2 = 0\,. \tag{84}$$

Eq. (84) also follows from the observation that $\xi = \infty$ implies zero wave length, and there is no information that is propagated along the characteristic path.

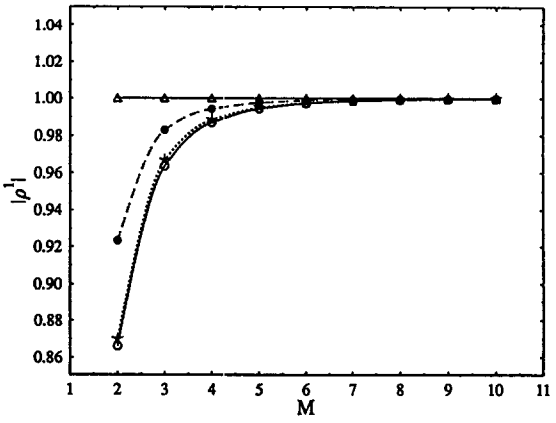
Let L be the wave length corresponding to ξ, and M be the number of elements per wave length. Then in the local coordinates $\xi = 2\pi/L = \pi/M$. The spectral radius of $\mathcal{G}_1(k, \xi)$, $|\rho(\mathcal{G}_1(k, \xi))| = |\rho^1|$, as function of $\tilde{M}\xi^{-1}$ for $\eta = 0, -1/2, -3/4$ and -1.0 are shown in Fig. 1. As $H_l(\eta, \eta_l)$ and $H_l^\eta(\eta, \eta_l)$ are symmetric about $\eta = 0$, the plots of $|\rho^1|$ are similar for $\eta \geqslant 0$. The figure indicates that $|\rho(\mathcal{G}_1(k, \xi))|$ is a monotonically increasing function of ξ^{-1}. By Eqs. (83) and (84) and Fig. 1, the maximum and minimum values of $|\rho(\mathcal{G}_1(k, \xi))|$ are 0 and 1, respectively. Therefore, the numerical procedure for the advection equation is unconditionally stable for all wave numbers.

As M is a measure of the spatial resolution of a wave by the numerical discretization, the analysis indicates that for a damping error of $1 - |\rho(\mathcal{G}_1(k, \xi))| \leqslant 0.001$ per time step a value of $M \geqslant 8$ should be used. For a given problem, this criterion provides a basis of selecting the maximum size of elements so that numerical damping is negligible. In addition, the solution will be propogated at the correct phase by satisfying the criterion. The numerical examples presented by Ding and Liu [17] and Holly and Preissmann [18] are consistent with these observations.

6.3. Diffusion equations

Let $\rho(\mathcal{G}_2(k, \xi))$ be the spectral radius of the amplification matrix $\mathcal{G}_2(k, \xi)$ for the Crank–Nicolson finite-element procedure for the diffusion equation. Then $\rho(\mathcal{G}_2(k, \xi))$ is given by [36]

$$\rho(\mathcal{G}_2(k, \xi)) = \frac{2 + \cos(\xi h) - 3P_e(1 - \cos(\xi h))}{2 + \cos(\xi h) + 3P_e(1 - \cos(\xi h))}\,, \tag{85}$$



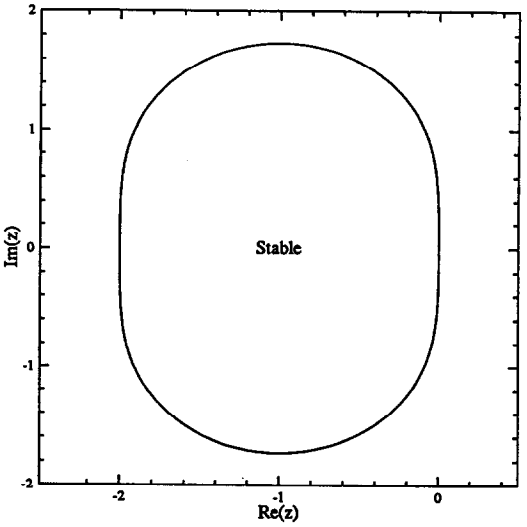


Fig. 1. Spectral radius, $|\rho^1|$, of the backward method of characteristics as a function of $M = \pi/\xi$.

Fig. 2. Stability domain of the explicit second-order accurate Runge–Kutta method.

where $P_e = \max_{1 \leq i \leq I} h^2 R_i / kD$. Eq. (85) is based on a linear stability analysis neglecting boundary conditions, and indicates that $|\rho(\mathcal{G}_2(k, \xi))|$ satisfies Eq. (80) for all ξ and P_e. Thus, the numerical procedure for the diffusion equation is unconditionally stable.

6.4. Reaction equations

The reaction equation (17) is an initial value problem. Therefore, ξ is not involved in the stability analysis. Let $\mathcal{G}_3(k, \xi) \equiv \mathcal{G}_3(z)$, then the stability polynomial of the two stage explicit second-order accurate Runge–Kutta method is given by [29]

$$\mathcal{G}_3(z) = 1 + z + \frac{z^2}{2}, \qquad z = -\bar{k} \max_{1 \leq i \leq I} \lambda_i . \tag{86}$$

Therefore, the algorithm will remain stable provided the roots of polynomial on the right-hand side of Eq. (86) satisfy $|\mathcal{G}_3(z)| \leq 1$. Fig. 2 shows the stability domain of the Runge–Kutta method, which is defined [29] as the set of complex numbers, z, such that $|\mathcal{G}_3(z)| \leq 1$. As the stability domain is a subset of the entire left-hand complex plane, the algorithm is conditionally stable.

The numerical procedures for the advection and the diffusion equations are unconditionally stable, while that for the reaction equation is conditionally stable. As a result, the composite algorithm (14) is conditionally stable. However, a sufficiently large time step, consistent with the accuracy of the algorithms for the advection and diffusion equations, can be used provided the reaction equation is sub-cycled.

7. Numerical examples

Four numerical examples are described in this section to demonstrate the simulation characteristics of the splitting algorithm, and to verify the numerical analyses presented in the previous sections. The numerical experiments are described in Subsections 7.1–7.4, and the simulation accuracy is analyzed in Subsection 7.5. The Courant and the Peclet numbers in these examples are representative of groundwater transport problems. The simulation characteristics of similar algorithms ($I = 1$ and $\mu_1 = 0$) for pure advection and highly advection dominated problems can be found in Ding and Liu [17] and Holly and Preissmann [18].

7.1. Example 1

In the first example, the advection–diffusion–reaction equation for $I = 1$ is considered. Thus, the problem does not involve inhomogeneous forcing term. A large number of analytical solutions for constant velocity, diffusion and decay coefficients is available in van Genuchten [37]. For the following initial condition

$$f_1^0(x) = 0\,, \qquad x \in [0, \infty)\,, \tag{87}$$

and boundary conditions

$$f_1^1(t) = 1, t \in (0, t_0]\,, \qquad f_1^1(t) = 0, t \in (t_0, \infty)\,, \qquad x = 0\,, \tag{88}$$

$$\partial_x c_1 = 0\,, \qquad x \to \infty\,, \tag{89}$$

the analytical solution can be expressed as [37]

$$\begin{aligned} c_1(t,x) &= \frac{1}{2} A(t,x)\,, \qquad t \in (0, t_0]\,, \\ &= \frac{1}{2}\left[A(t,x) - A(t - t_0, x)\right], \qquad t \in (t_0, \infty)\,, \end{aligned} \tag{90}$$

where

$$A(t,x) = \exp\left[\frac{(u-\alpha)x}{2D}\right] \operatorname{erfc}\left[\frac{R_1 x - \alpha t}{2\sqrt{DR_1 t}}\right] + \exp\left[\frac{(u+\alpha)x}{2D}\right] \operatorname{erfc}\left[\frac{R_1 x + \alpha t}{2\sqrt{DR_1 t}}\right],$$

$$\alpha = u\left[1 + \frac{4\lambda_1 D}{u^2}\right]^{1/2}. \tag{91}$$

Fig. 3 compares the numerical solutions with the analytical solutions at $t = 5.0, 7.5, 10.0$ and 12.5 days for $t_0 = 5.0$ days and the following model parameters

$$R_1 = 3.0\,, \qquad \lambda_1 = 7.235 \cdot 10^{-7}\,\mathrm{s}^{-1}\,,$$

$$u = 2.894 \cdot 10^{-6}\,\mathrm{m\,s}^{-1}\,, \qquad D = 4.340 \cdot 10^{-8}\,\mathrm{m}^2\,\mathrm{s}^{-1}\,.$$

The computational time step, k, and the nodal spacing, h, are selected such that the Courant and the Peclet numbers are 0.5 and 13.33, respectively. A Courant number of 0.5 is used as the corresponding

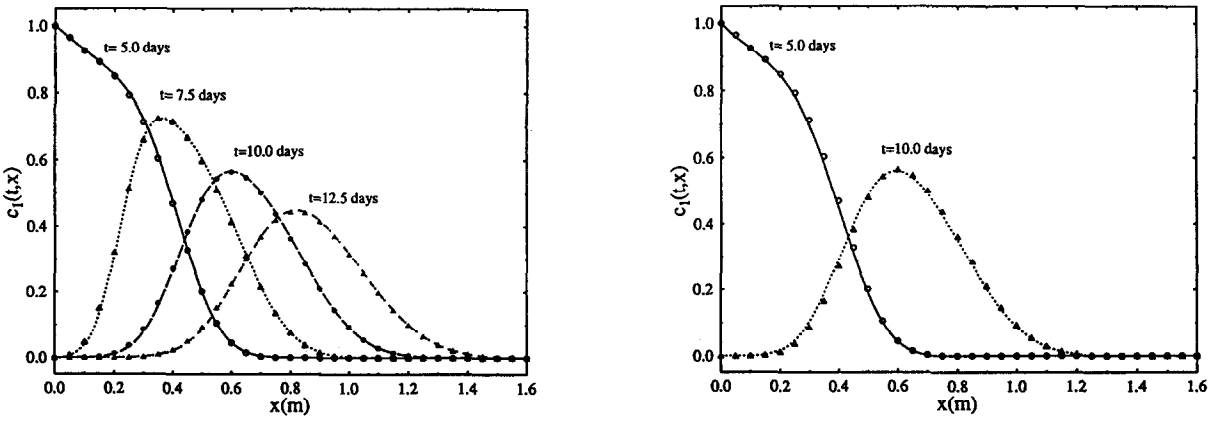


Fig. 3. A comparison of the numerical and analytical solutions for Example 1; lines are analytical solutions and symbols are numerical solutions.

Fig. 4. A comparison of numerical solutions obtained by first-order accurate algorithm (13) (symbols) and second-order accurate algorithm (14) (lines) for constant u, D and R_1.

interpolation error and numerical dissipation are maximum. The computed results shown in Fig. 3 are in good agreement with the analytical solutions.

In this example the coefficients u, D, R_1 and λ_1 are constants. Therefore, the splitting error, $\epsilon_1^S(k)$, is zero. Under these conditions algorithms (13) and (14) are equivalent. Fig. 4 compares the solutions obtained by the two splitting algorithms at $t = 5.0$ and 10.0 days. The discrepancy between the two solutions is indistinguishable, which is consistent with the theoretical analyses presented in Sections 2 and 4.

7.2. Example 2

The second example considers the transport of ammonia in a porous media, which is accompanied by a nitrification process—an example of biochemical reactions induced by the subsurface microorganisms [2]. The transformation incorporating three species, $I = 3$, can be represented by

$$\mathrm{NH}_4^+ \xrightarrow{\lambda_1} \mathrm{NO}_2^- \xrightarrow{\lambda_2} \mathrm{NO}_3^- . \tag{92}$$

Cho [38] and van Genuchten [39] have developed analytical solutions of Eqs. (1) for constant Dirichlet boundary conditions. The analytical solutions are not shown because they are quite complicated. Fig. 5 compares the numerical and the analytical solutions for the following initial conditions

$$f_i^0(x) = 0 , \qquad x \in [0, \infty) , \quad \forall i , \tag{93}$$

and Dirichlet boundary conditions

$$f_1^1(t) = 1 , \qquad f_2^1(t) = f_3^1(t) = 0 , \qquad x = 0 , \tag{94}$$

$$f_i^1(t) = 0 , \qquad x \to \infty , \quad \forall i . \tag{95}$$

The following flow and transport parameters from van Genuchten [39] are used in the computations

$$u = 2.778 \cdot 10^{-6}\ \mathrm{m\,s^{-1}} , \qquad D = 5.0 \cdot 10^{-9}\ \mathrm{m^2\,s^{-1}} ,$$
$$R_1 = 2 , \qquad R_2 = R_3 = 1 ,$$
$$\lambda_1 = 1.389 \cdot 10^{-6}\ \mathrm{s^{-1}} , \qquad \lambda_2 = 2.778 \cdot 10^{-5}\ \mathrm{s^{-1}} , \qquad \lambda_3 = 0 ,$$

and the results are compared at $t = 8.333$ days. The Courant numbers in the computations are 0.5 (for $i = 1$) and 0.25 (for $i = 2$ and 3), while the Peclet number is 27.87. The discrepancy between the two solution is indistinguishable, except near the inflection points in the spatial distribution of NH_4^+.

7.3. Example 3

In the third example, the migration of a four-member, $I = 4$, radionuclide decay chain

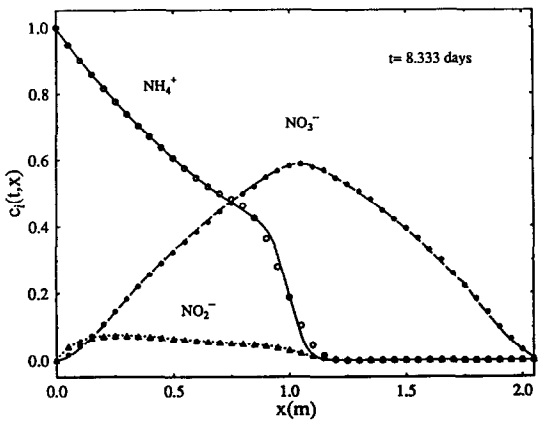
$$^{238}\mathrm{Pu} \xrightarrow{\lambda_1} {}^{234}\mathrm{U} \xrightarrow{\lambda_2} {}^{230}\mathrm{Th} \xrightarrow{\lambda_3} {}^{226}\mathrm{Ra} , \tag{96}$$

is simulated. The decay constants in Eq. (96) are given by $\lambda_i = \ln 2 / T_{i,1/2}$, where $T_{i,1/2}$ is the half-life of the ith radionuclide. The Dirichlet boundary conditions [39, 40] at $x = 0$ can be expressed as

$$f_i^1(t) = \sum_{l=1}^{i} a_{i,l} \exp\{-(\lambda_l + \beta_l)t\} , \quad i = 1, \ldots, I , \tag{97}$$

where β_l is the rate of release of the lth radionuclide at $x = 0$, and $a_{i,l}$'s are constants depending on the mechanism by which the radionuclides are released. Eq. (97) represents an exponential release mode. The details of these boundary conditions, and the values of the coefficients, $a_{i,l}$, for the radionuclide decay chain (96) can be found in van Genuchten [39] and Higashi and Pigford [40]. The initial conditions are given by Eqs. (93), while the boundary conditions as $x \to \infty$ are given by Eq. (95).

The model parameters used in the computations are as follows [39]



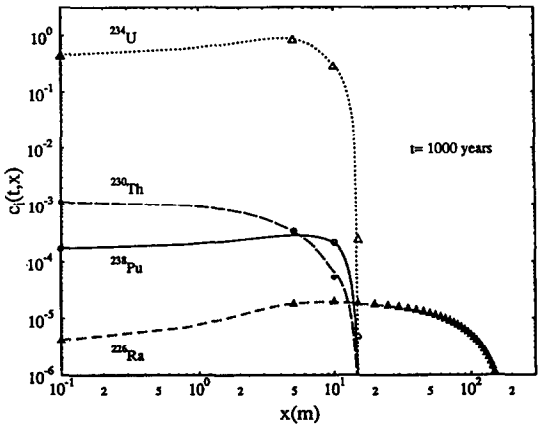


Fig. 5. A comparison of the numerical and analytical solutions for Example 2; lines are analytical solutions and symbols are numerical solutions.

Fig. 6. A comparison of the numerical and analytical solutions for Example 3; lines are analytical solutions and symbols are numerical solutions.

$$\lambda_1 = 2.506 \cdot 10^{-10}, \quad \lambda_2 = 8.971 \cdot 10^{-14}, \quad \lambda_3 = 2.747 \cdot 10^{-13}, \quad \lambda_4 = 1.374 \cdot 10^{-11}\ \mathrm{s}^{-1},$$
$$\beta_1 = \beta_2 = \beta_3 = \beta_4 = 3.171 \cdot 10^{-11}\ \mathrm{s}^{-1},$$
$$u = 3.171 \cdot 10^{-6}\ \mathrm{m\,s}^{-1}, \quad D = 3.171 \cdot 10^{-7}\ \mathrm{m}^2\,\mathrm{s}^{-1},$$
$$R_1 = 1.0 \cdot 10^4, \quad R_2 = 1.4 \cdot 10^4, \quad R_3 = 5.0 \cdot 10^4, \quad R_4 = 5.0 \cdot 10^2.$$

The results of numerical computations at $t = 1000$ years are compared with the analytical solutions presented by van Genuchten [39] in Fig. 6. In this example, the effective transport velocities, u/R_i, and the diffusion coefficients, D/R_i, vary over two order of magnitude, and the computational time step is determined by the model parameters for ^{226}Ra. As in the previous examples, these results agree very well with the analytical solutions.

7.4. Example 4

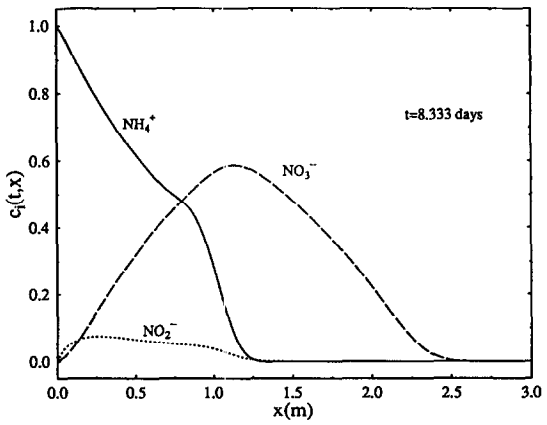
In the previous examples, splitting errors, $\epsilon_i^S(k)$, are zero. To introduce splitting error in the computations, it is necessary to assume at least one of the parameters u, D, R_i or λ_i is a function of x and/or $c_i(t, x)$. In this context, Example 2 is considered with the following changes

$$u = 2.778 \cdot 10^{-6}\left(1 + \frac{x}{8}\right), \quad D = \alpha u, \tag{98}$$

where α is an empirical coefficient known as longitudinal dispersivity [1]. A value of $\alpha = 0.0018$ m is used in this example. The nonuniform velocity distribution in Eq. (98) can be realized in a soil column experiment with a steady flow in a converging cylinder. The previous examples thus correspond to steady flow in a cylinder of constant diameter. The computed longitudinal distributions of concentrations for the three species at $t = 8.333$ days are shown in Fig. 7. As a result of higher velocity and diffusion coefficient, compared with Example 2, the computed concentrations are generally higher and the species have migrated further into the porous media.

7.5. Accuracy and convergence

The operator splitting algorithm presented in this study is second-order accurate in time and space. This is illustrated by analyzing the computational errors in Examples 2 and 4. In Example 2, the maximum difference (absolute value) between the analytical and numerical solutions in the computa-



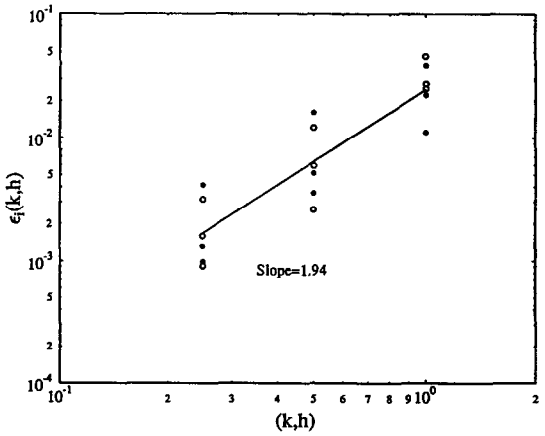


Fig. 7. Computed results for Example 4 with non-uniform velocity distribution and velocity dependent diffusion coefficient.

Fig. 8. Plot of $\epsilon_i(k, h)$ as functions of (k, h) for Example 2 (full circle) and Example 4 (open circle). Slope of the best fit straight line is approximately 1.94.

tional domain is taken as error in the computation. As analytical solution is not available for Example 4, the exact solution is approximated by refining (k, h) in the following sequence

$$(k, h) = \left(\frac{1}{2^m}, \frac{1}{2^m}\right), \quad m = 0, \ldots, 4. \tag{99}$$

Let $\tilde{c}_{i,[2^m]}$ be the computed result corresponding to (k, h) defined by Eq. (99) for $m \leqslant 4$. Then the pointwise error in the computations is approximated by

$$\epsilon_i(k, h) \approx \max_{x_l \in \Omega} |\tilde{c}_{i,[2^m]}(T, x_l) - \tilde{c}_{i,[2^4]}(T, x_l)|, \quad m \leqslant 2, \quad \forall i, \tag{100}$$

where T is the simulation period. Fig. 8 shows the plot of $\epsilon_i(k, h)$ as functions of (k, h). Considering all the points as a set of data, the best fit line (in log–log graph) is also known in the figure. The slope of the line is 1.94, approximately indicating that the operator splitting algorithm is $\mathcal{O}(k^2, h^2)$ accurate. The scatter in the plotted data is mainly due to different Courant and Peclet numbers associated with each species. The magnitude of interpolation error during the advection step of computations is quite sensitive to the Courant number. Eq. (61) provides an estimate of the maximum interpolation error corresponding to a Courant number of half or its multiple. As the Courant number approaches either zero or one, the interpolation error goes to zero [18].

8. Summary and conclusions

In this paper an operator splitting algorithm for the solution of a system of coupled inhomogeneous equations, representing the transport and the transform of non-conservative pollutants, has been presented. The solution of each equation is expressed by the superposition of the solution of the associated homogeneous equation, and the contribution from the inhomogeneous term. The homogeneous equation for each species is split into an advection equation, a diffusion equation and a reaction equation. The split advection equation is solved by a backward method of characteristics. A C^1 continuous Hermite function is used for the spatial interpolations of concentration. The diffusion equation is solved by the Crank–Nicolson finite-element method. The reaction equation is integrated in time by an explicit Runge–Kutta method. The contribution from the inhomogeneous term, coupling the system of equations, is approximated by the trapezoidal integration rule.

The error analysis of the algorithm indicates that the numerical procedure is second-order accurate in both time and space. The composite solution is made second-order accurate by adopting Strang's

splitting procedure, and using at least second-order accurate numerical algorithms for the split equations. The numerical procedures for the split advection and diffusion equations are unconditionally stable, while that for the reaction equations is conditionally stable. Consequently, the composite algorithm is also conditionally stable. However, by sub-cycling the reaction equations, a large time step consistent with the accuracy of the numerical procedures for the split advection and diffusion equations can be used. The numerical characteristics of the algorithm have been demonstrated by several numerical examples.

Acknowledgments

The research reported in this paper was in part supported by a research grant from the Xerox Corporation to Cornell University. During the course of study LAK has also been supported by a Fellowship from the DeFrees Foundation. This research was conducted using the Cornell National Supercomputer Facility, which receives major funding from the National Science Foundation and IBM Corporation, and with additional support from New York State and members of the Corporate Research Institute.

References

[1] J. Bear and A. Verruijt, Modeling Groundwater Flow and Pollution (D. Reidel, Dordrecht, 1987).
[2] C.N. Sawyer and P.L. McCarty, Chemistry for Environmental Engineers (McGraw-Hill, New York, 1978).
[3] M.F. Wheeler and C.N. Dawson, An operator-splitting method for advection–diffusion–reaction problems, in: J.A. Whiteman, ed., The Mathematics of Finite Elements and Applications (Academic Press, New York, 1988) 463–82.
[4] C.N. Dawson and M.F. Wheeler, Time-splitting methods for advection–diffusion–reaction equations arising in contaminant transport, in: R.E. O'Malley Jr., ed., ICIAM 91, SIAM Phil. (1992) 71–82.
[5] W.W. McNab Jr. and T.N. Narasimhan, A multiple species transport model with sequential decay chain interactions in heterogeneous subsurface environments, Water Resour. Res. 29(8) (1993) 2737–46.
[6] M.F. Wheeler, C.N. Dawson, P.B. Bedient, C.Y. Chiang, R.C. Borden and H.S. Rifai, Numerical simulation of microbial biodegradation of hydrocarbons in ground water, Proc. NWWA Conf., Dublin, Ohio (1987) 92–109.
[7] H.S. Rifai and P.B. Bedient, Comparison of biodegradation kinetics with an instantaneous reaction model for groundwater, Water Resour. Res. 26(4) (1990) 637–45.
[8] W. Kinzelbach, W. Schafer and J. Herzer, Numerical modeling of natural and enhanced denitrification processes in aquifers, Water Resour. Res. 27(6) (1991) 1123–35.
[9] C.T. Miller and A.J. Rabideau, Development of split-operator Petrov–Galerkin methods to simulate transport and diffusion problems, Water Resour. Res. 29(7) (1993) 2227–40.
[10] A.F.B. Tompson and D.E. Dougherty, Particle-grid methods for reacting flows in porous media: application to Fischer's equation, in: G. Gambolati et al., ed., Computational Methods in Subsurface Hydrology (Computational Mechanics Publ., 1990) 393–98.
[11] A.J. Valocchi and M. Malmstead, Accuracy of operator splitting for advection–dispersion–reaction problems, Water Resour. Res. 28(5) (1992) 1471–76.
[12] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (1968) 506–17.
[13] R.J. LeVeque and J. Oliger, Numerical methods based on additive splittings for hyperbolic partial differential equations, Math. Comp. 40(162) (1983) 469–97.
[14] N.N. Yanenko, The Method of Fractional Steps (Springer-Verlag, Berlin, 1971).
[15] G.I. Marchuk, Splitting and alternating direction methods, in: P.G. Ciarlet and J.L. Lions, eds., Handbook of Numerical Analysis, Vol. I, Finite Difference Methods (Part 1)—Solution of Equations in R^n (Part 1) (North-Holland, Amsterdam, 1990) 197–462.
[16] D. Ding and P.L.-F. Liu, An operator-splitting algorithm for two-dimensional convection–dispersion–reaction problems, in: H. Niki and M. Kawahara, eds., Computational Methods in Flow Analysis (Okayama Univ. Sci., 1988) 1101–08.
[17] D. Ding and P.L.-F. Liu, An operator-splitting algorithm for two-dimensional convection–dispersion–reaction problems, Int. J. Numer. Methods Engrg. 28 (1989) 1023–40.
[18] F.M. Holly Jr. and A. Preissmann, Accurate calculation of transport in two-dimensions, Proc. ASCE J. Hyd. Div. 103 (1977) 1259–78.
[19] H.-O. Kreiss and J. Lorenz, Initial-Boundary Value Problems and the Navier–Stokes Equations (Academic Press, New York, 1989).
[20] J.C. Strikwerda, Finite Difference Schemes and Partial Differential Equations (Wadsworth & Brook/Cole, Belmont, CA, 1989).

[21] H. Fujita and T. Suzuki, Evolution problems, in: P.G. Ciarlet and J.L. Lions, eds., Handbook of Numerical Analysis, Vol. II, Finite Element Methods (Part 1), (North-Holland, Amsterdam, 1991) 791–926.
[22] L. Demkowicz, T.J. Oden and W. Rachowicz, A new finite element method for solving compressible Navier–Stokes equations based on an operator splitting method and *h-p* adaptivity, Comput. Methods Appl. Mech. Engrg. 84 (1990) 275–326.
[23] J.B. Perot, An analysis of the fractional step method, J. Comput. Phys. 108 (1993) 51–58.
[24] J. Glass and W. Rodi, A higher order numerical scheme for scalar transport, Comput. Methods Appl. Mech. Engrg. 31 (1982) 337–58.
[25] G. Yang, P. Belleudy and A. Temperville, A higher-order Eulerian scheme for coupled advection–diffusion transport, Int. J. Numer. Methods Fluids 12 (1991) 43–59.
[26] J.-C. Yang, K.-N. Chen and H.-Y. Lee, Investigation of use of reach-back characteristic method for 2-D dispersion equation, Int. J. Numer. Methods Fluids 13 (1991) 841–55.
[27] L. Lapidus and G.F. Pinder, Numerical Solution of Partial Differential Equations in Science and Engineering (John Wiley and Sons, New York, 1982).
[28] P.J.Rasch and D.L. Williamson, On shape preserving interpolation and semi-Lagrangian transport, SIAM J. Sci. Stat. Comput. 11(4) (1990) 656–87.
[29] E. Hairer and G. Wanner, Solving Ordinary Differential Equations—II: Stiff and Differential-Algebraic Problems (Springer-Verlag, Berlin, 1991).
[30] R.J. Le Veque, Intermediate boundary conditions for time-split methods applied to hyperbolic partial differential equations, Math. Comp. 47(175) (1986) 37–54.
[31] G.R. McGuire and J. Ll. Morris, Boundary techniques for the multistep formulation of the optimized Lax–Wendroff method for non-linear hyperbolic systems in two space dimensions, J. Inst. Maths. Applics. 10 (1972) 150–65.
[32] A.R. Gourlay and A.R. Mitchell, Intermediate boundary corrections for split operator methods in three dimensions, BIT 7 (1967) 31–38.
[33] B.P. Sommeijer, P.J. van der Houwen and J.G. Verwer, On the treatment of time-dependent boundary conditions in splitting methods for parabolic differential equations, Int. J. Numer. Methods Engrg. 17 (1981) 335–46.
[34] F.M. Holly Jr. and J.M. Usseglio-Polatera, Dispersion simulation in two-dimensional tidal flow, Proc. ASCE J. Hyd. Div. 103(HY11) (1984) 905–26.
[35] J.T. Oden and G.F. Carey, Finite Elements: Mathematical Aspects, IV (Prentice-Hall, Englewood Cliffs, NJ, 1983).
[36] W.G. Gray and G.F. Pinder, An analysis of the numerical solution of the transport equation, Water Resour. Res. 12(3) (1976) 547–55.
[37] M. Th. van Genuchten, Analytical solutions for chemical transport with simultaneous adsorption, zero-order production and first-order decay, J. Hydrology 49 (1981) 213–33.
[38] C.M. Cho, Convective transport of ammonia with nitrification in soil, Can. J. Soil Sci. 51 (1971) 339–50.
[39] M. Th. van Genuchten, Convective-dispersive transport of solutes involved in sequential first-order decay reactions, Comput. Geosci. 11(2) (1985) 129–47.
[40] K. Higashi and T.H. Pigford, Analytical models for migration of radionuclides in geologic sorbing media, J. Nucl. Sci. Tech. 15(9) (1980) 700–09.