JOURNAL OF COMPUTATIONAL PHYSICS 45, 1–42 (1982)

Review Article

Numerical Solution of the Atmospheric Diffusion Equation for Chemically Reacting Flows

GREGORY J. MCRAE, WILLIAM R. GOODIN,* AND JOHN H. SEINFELD†

Environmental Quality Laboratory, California Institute of Technology, Pasadena, California 91125

Received May 15, 1981

A comprehensive study of numerical techniques for solving the atmospheric diffusion equation is reported. Operator splitting methods are examined in which the three-dimensional problem is converted into a sequence of one-dimensional problems. A Galerkin, linear finite element scheme with a nonlinear filter is found to be computationally superior to the other methods tested for the advection–diffusion components. The chemical reaction dynamics component, treated within the splitting scheme, is generally highly stiff. A second-order predictor, iterated corrector technique, in combination with an asymptotic treatment of the stiff components, is found to be computationally superior for the chemical kinetics. The validity of the pseudo steady state approximation for certain reactive species is also investigated.

1. INTRODUCTION

Many disciplines in engineering and science depend on the availability of predictive models of chemically reacting fluid flows. One area of considerable practical interest and a source of many challenging problems in numerical analysis is the construction of mathematical descriptions of the formation and transport of urban-scale air pollution. A complete treatment of atmospheric concentration dynamics and chemical interactions involves the full, three-dimensional turbulent planetary boundary layer equations for conservation of mass, momentum and energy. Unfortunately the routine solution of such a system is an enormous undertaking and not feasible on the present generation of computers since a typical calculation might involve $O(10^4)$ grid points, 20–50 chemical species and $O(10^6)$ computer storage locations. A somewhat more limited approach, and the focus of this work, is based on decoupling the mass conservation equations from the equations of motion of the air. This simplification results in a set of coupled parabolic partial differential equations that describe the combined

* *Present address*: Advanced Technology Group, Dames and Moore, 1100 Glendon Avenue, Los Angeles, CA 90024.

† Department of Chemical Engineering.

0021-9991/82/010001-42$02.00/0

influences of advection, turbulent diffusion and chemistry within a prescribed flow field. The presence of nonlinearities and the existence of widely disparate temporal and spatial scales considerably complicate the selection and implementation of suitable solution techniques. In addition the availability and utilization of computational resources are practical considerations equally as important as the requirement for numerical accuracy.

This paper begins with a general statement of the atmospheric diffusion equation and proceeds to describe the use of coordinate transformations and operator splitting techniques for numerical solution. Once the equations have been decomposed into component parts by operator splitting, specially suited procedures for the components (advection, diffusion, and chemical reaction) can be applied. We then describe the choice and testing of appropriate techniques for solving the transport or advection–diffusion components of the equation. The final element involves numerical solution of the chemical kinetics. Although the numerical techniques described in this work have been specifically developed to solve the atmospheric diffusion equation, much of the material is applicable to other problems, particularly those that involve chemically reacting fluid flows.

2. Governing Differential Equations

Consider an arbitrary, time-varying, spatial domain Ω_t located in the Euclidean space E^3 and bounded by $\partial\Omega_t$. In this region, a spatial point is denoted $\mathbf{X} = \{X, Y, Z\} \in \Omega_t$. Within Ω_t the conservation of mass for each of p chemical species $c_i(\mathbf{X}, t)$; $i = 1,..., p$, can be expressed by the following set of coupled, nonlinear, parabolic, partial differential equations,

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (\mathbf{u}c_i) = \nabla \cdot (\mathbf{K} \cdot \nabla c_i) + f_i(c_1,..., c_p), \tag{1}$$

with $(\mathbf{X}, t) \in \Omega_t \times [0, T]$. For this system $\mathbf{u}$ is the prescribed advective velocity field $\mathbf{u}(\mathbf{X}, t) = (u, v, w)$, $\mathbf{K}$ is a second-order, diagonal, eddy diffusivity tensor and f_i a temperature dependent chemical formation (or depletion) rate of species i. In meteorological applications (1) is frequently called the atmospheric diffusion equation [1].

To complete the problem formulation both the initial and boundary conditions need to be specified. For the system (1) the initial conditions $c_i(\mathbf{X}, 0)$, are given by

$$c_i(\mathbf{X}, 0) = c_i^0(\mathbf{X}); \qquad i = 1,..., p; \qquad \mathbf{X} \in \Omega_0. \tag{2}$$

The measured concentration data, from which the initial conditions are normally specified, are sparse, irregularly spaced, and generally limited to ground level values. Under these conditions, a representative initial field can be obtained by interpolation using the techniques described in Goodin *et al.* [2–4]. Boundary conditions simply represent statements of mass continuity across the enclosing surface $\partial\Omega_t$. For this

system most practical cases are described by the inhomogeneous mixed Neumann and Dirichlet boundary conditions

$$a(\mathbf{X}, t)\, c_i + b(\mathbf{X}, t) \frac{\partial c_i}{\partial \sigma} = g_i(\mathbf{X}, t); \qquad (\mathbf{X}, t) \in \partial\Omega_t \times [0, T]. \tag{3}$$

In this equation σ indicates the normal direction to $\partial\Omega_t$, and the functions $a(\mathbf{X}, t)$, $b(\mathbf{X}, t)$ and $g_i(\mathbf{X}, t)$ describe particular cases, the explicit forms of which are presented in Reynolds *et al.* [5].

The difficulties that arise during numerical solution of (1)–(3) stem from the radically different character of the advection, $\nabla \cdot (\mathbf{u}c_i)$, turbulent diffusion, $\nabla \cdot (\mathbf{K} \cdot \nabla c_i)$, and chemical reaction, f_i, operators. Even though (1) is formally parabolic in most atmospheric flows, transport in the horizontal plane is dominated by advection, leading to hyperbolic like characteristics. One of the major sources of difficulty arises during numerical solution of the chemical reaction terms f_i. While complicating the numerical solution, the presence of the nonlinearities in f_i is not as much a problem as the potential for eigenvalues that span a wide range of time scales. In typical photochemical reaction mechanisms of the type described by Falls and Seinfeld [6], it is possible to encounter situations in which individual reaction times differ by $O(10^8$ seconds). That, in turn, virtually dictates an implicit solution procedure for the chemical kinetics.

3. Coordinate Transformations

In typical applications the airshed domain Ω_t is defined by three bounding surfaces; the topography $Z = h(X, Y)$, vertical sides at the horizontal extremes, and a time varying upper boundary, $Z = H(X, Y, t)$. The presence of topographic relief can considerably complicate the numerical implementation of boundary conditions of the form (3). The problem can be avoided to a certain extent by transforming the spatial domain into one of simpler geometry. This can be accomplished by a mapping $\mathbf{F}: \Omega_t \to \Omega_c$, that transforms points in the physical domain Ω_t into the more convenient computational domain Ω_c. Points in Ω_c are denoted by $\mathbf{x} = (x, y, z, t)$.

A variety of functional forms for $\mathbf{F}$ are used in practice; a common one in atmospheric modeling application is the terrain-following coordinate transformation [5, 7, 8],

$$\mathbf{x} = F(\mathbf{X}) = \begin{bmatrix} X \\ Y \\ \dfrac{Z - h(X, Y)}{H(X, Y, t) - h(X, Y)} \end{bmatrix}, \tag{4}$$

that scales the vertical extent of the modeling region into the new domain $z \in [0, 1]$. So long as the time varying upper boundary H, does not intersect the terrain defined

by h, then a unique inverse for (4) exists. The general requirement for a nonzero Jacobian usually precludes the simultaneous use of these transformations in all three coordinate directions.

Once the form of the transformation has been established, the next step is to apply it to the atmospheric diffusion equation. An important characteristic of this equation is that it represents a differential statement of the conservation of mass for each species c_i. Roache [9] indicates that, with few exceptions, the most accurate numerical results are obtained using numerical approximations that are based on the flux or conservative form of the governing equations. With this in mind, it is desirable to preserve the conservative structure of (1) during the coordinate transformation. If this is done, then it is possible to consider each computational cell as a control volume and develop difference expressions that satisfy the physical conservation laws on a macroscopic level, not at the limit of small time steps and spatial dimensions. Methods for manipulating first and second-order partial differential equations that preserve the conservative properties are described in Anderson *et al.* [10], Oberkampf [11], and Vinokur [12]. Lapidus [13], in particular, has shown that with a nonsingular space transformation, the conservative form of the governing differential equations can be maintained. Using these procedures it is possible to develop the following conservative form of the atmospheric diffusion equation,

$$\frac{\partial \Delta H c_i}{\partial t} + \nabla \cdot (\mathbf{V} \Delta H c_i) = \nabla \cdot (\Delta H \mathbf{K}_c \cdot \nabla c_i) + \Delta H f_i, \qquad (\mathbf{x}, t) \in \Omega_c \times [0, T], \quad (5)$$

where Ω_c is now the transformed domain and $\Delta H = H(x, y, t) - h(x, y)$. The components of the transformed velocity field are now $\mathbf{V} = (u, v, W)$, where the new vertical velocity W, is given by

$$W = \frac{1}{\Delta H}\left[w - u\left(\frac{\partial h}{\partial x} + z\frac{\partial \Delta H}{\partial x}\right) - v\left(\frac{\partial h}{\partial y} + z\frac{\partial \Delta H}{\partial y}\right) - z\frac{\partial \Delta H}{\partial t}\right]. \quad (6)$$

One problem arises as a result of the transformation. Initially the eddy diffusivity tensor $\mathbf{K}$ was diagonal, however, the transformed form is given by

$$\mathbf{K}_c = \begin{bmatrix} K_{xx} & 0 & -\dfrac{K_{xx}}{\Delta H}\left(\dfrac{\partial h}{\partial x} + z\dfrac{\partial \Delta H}{\partial x}\right) \\ 0 & K_{yy} & -\dfrac{K_{yy}}{\Delta H}\left(\dfrac{\partial h}{\partial y} + z\dfrac{\partial \Delta H}{\partial y}\right) \\ -\dfrac{K_{xx}}{\Delta H}\left(\dfrac{\partial h}{\partial x} + z\dfrac{\partial \Delta H}{\partial x}\right) & -\dfrac{K_{yy}}{\Delta H}\left(\dfrac{\partial h}{\partial y} + z\dfrac{\partial \Delta H}{\partial y}\right) & \dfrac{K_{xx}}{\Delta H^2}\left(\dfrac{\partial h}{\partial x} + z\dfrac{\partial \Delta H}{\partial x}\right)^2 \\ & & + \dfrac{K_{yy}}{\Delta H^2}\left(\dfrac{\partial h}{\partial y} + z\dfrac{\partial \Delta H}{\partial y}\right)^2 \\ & & + \dfrac{K_{zz}}{\Delta H^2} \end{bmatrix}. \quad (7)$$

While the presence of off-diagonal terms can complicate the numerical solution, their contribution to turbulent transport in most urban scale flows is negligible since it is possible to show, that for all but the most rugged terrain,

$$\frac{1}{\Delta H}\left[\frac{\partial h}{\partial x}+z\frac{\partial \Delta H}{\partial x}\right]\ll 1; \qquad \frac{1}{\Delta H}\left[\frac{\partial h}{\partial y}+z\frac{\partial \Delta H}{\partial y}\right]\ll 1. \tag{8}$$

4. General Approach to the Numerical Solution of the Problem

Once the equations have been transformed, the next step is to formulate an approach for obtaining numerical solutions of the model system. Although the focus of the present work is the atmospheric diffusion equation, the problem can be stated in the more general form,

$$\frac{\partial c_i}{\partial t}=L(\mathbf{x},t)\,c_i(\mathbf{x},t)+f_i(c_1,\ldots,c_p); \qquad (\mathbf{x},t)\in\Omega_c\times[0,T], \tag{9}$$

$$B(\mathbf{x},t)\,c_i(\mathbf{x},t)=g_i(\mathbf{x},t); \qquad (\mathbf{x},t)\in\partial\Omega_c\times[0,T], \tag{10}$$

$$c_i(\mathbf{x},0)=c_i^0(\mathbf{x}); \qquad (\mathbf{x})\in\Omega_0, \tag{11}$$

where L is a multi-dimensional, semi-linear, elliptic differential operator containing first and second-order derivatives, with respect to x, y, and z, but no mixed derivatives, and B is a linear operator.

While there is an extensive literature relevant to obtaining solutions of the homogeneous system there are relatively few numerical treatments of problems that involve *both* chemical reactions and transport in three dimensions. Even though much of what is available is confined to one- and two-dimensional systems, many different techniques have been applied in practice. For example, Margolis [14] used the method of lines to examine the multi-component chemical dynamics of a premixed laminar flame. Chin and Braum [15] employed a discrete analog of the invariant embedding algorithm to solve the two-point boundary value problem associated with a model of oil shale retorting. A variety of schemes were used by Engquist *et al.* [16] to predict the performance of a catalytic converter; a fourth-order dissipative leap-frog difference method for the hyperbolic components, a Dufort–Frankel procedure for the parabolic elements and Newton iteration for the remaining nonlinear equations. Douglas *et al.* [17] utilized an extrapolated Crank–Nicholson–Galerkin procedure when solving a quasilinear parabolic problem. Kansa [18] used a block implicit scheme, modified to include the basic strategies of stiff ordinary differential equation solution algorithms, to model the combustion process in an axisymmetric wick.

There are two basic steps that must be undertaken as part of most approaches to obtaining numerical solutions of systems of the form (9)–(11). One is to identify the means for approximating the spatial derivatives and the other is to select the

procedure for time integration. Spatial discretization techniques are used to convert the system of time-varying partial differential equations into a set of ordinary differential equations. In most cases this can be accomplished by using either classical finite difference or finite element techniques to produce semi-discrete systems of the form

$$\mathbf{M}\frac{d\mathbf{c}_i}{dt} + \mathbf{S}\mathbf{c}_i = \mathbf{f}_i(c_1, ..., c_p, t), \tag{12}$$

where the matrices $\mathbf{M}$ and $\mathbf{S}$ are typically large and sparse, especially for multi-dimensional problems, and $\mathbf{c}_i$ is a vector-valued function representing the concentration distribution at r points in the computational domain. If $\mathbf{M}$ is the identity matrix, as is often the case when finite difference techniques are used, then the system (12) can be solved using the method of lines. Further details of different parameterizations of the elements of $\mathbf{M}$ and $\mathbf{S}$ are discussed subsequently.

One of the major difficulties associated with a solution of (12) is that the set of equations is usually quite stiff. Consider for example, the case of $\mathbf{f} = \mathbf{f}(t)$ only and constant $\mathbf{M}$. Then the analytic solution of (12) is given by

$$\mathbf{c}_i(t) = \exp\{-\mathbf{M}^{-1}\mathbf{S}t\}\,\mathbf{c}_i(0) + \int_0^t \exp\{-(t-\tau)\,\mathbf{M}^{-1}\mathbf{S}\}\,\mathbf{S}^{-1}\mathbf{f}_i(\tau)\,d\tau. \tag{13}$$

If δ is the discretization parameter, either the computational cell size or finite element, then the condition number of $\mathbf{M}^{-1}\mathbf{S}$ is $O(\delta^{-2})$ [19, 20]. In fact, because of the unboundedness of the eigenvalue spectrum as $\delta \to 0$, increasing demands for accuracy simply exacerbate the stiffness problem. What is not often recognized is that the stiffness of the ordinary differential equations may be an artifact of the spatial discretization and, apart from the character of $\mathbf{f}$, is not a property of the governing differential equations. Because the equations are stiff this usually dictates that an implicit solution procedure must be used for the time integration. While not a major restriction for one-dimensional systems, this can create major computational and operational problems when extended to higher dimensions.

In many situations the practical aspects of implementing the computational procedures can impose another set of limitations. Often the number of previous results that can be held in fast core storage, during one solution step, constrains the choice of a time integration procedure. In addition, careful consideration must be given to the way in which arrays are indexed on computers that employ virtual memory systems otherwise the paging overheads can become very large. These issues, and the theoretical considerations discussed above, are some of the major motivations for the use of operator splitting techniques.

5. Operator Splitting and the Method of Fractional Steps

If the spatial discretization procedures are directly applied to the three-dimensional operator L, the resulting matrices, while sparse, usually cannot be economically decomposed or inverted. One way to reduce the magnitude of the computational task is to employ operator splitting and reduce the multidimensional problem to a sequence of one-dimensional equations. If this is done then successive solutions to each component part can be combined to produce a "weak" approximation to the original operator. There are a number of significant advantages to be gained from this approach. Because the matrices arising from the one-dimensional spatial discretizations are usually tridiagonal, the cost of using stable implicit procedures is small. Perhaps the most important benefit is that the numerical solution techniques can be tailored to the physical behavior of each element, a feature that is particularly attractive in applications involving chemically reacting flows. For example, Rizzi and Bailey [21] used the space-marching procedure of Rizzi and Inouye [22], in combination with operator splitting, to examine the chemical dynamics occurring in supersonic flows over complex geometric shapes. Similar approaches were adopted by Kee and Miller [23] in a study of laminar diffusion flames and by Thomas and Wilson [24] for chemically reacting turbulent jets. In each case isolating the reaction kinetics removed the numerical time step restrictions on the transport operators imposed by the chemistry.

The initial step in operator splitting is to decompose L into a sum of simpler terms

$$L = \sum_{j=1}^{3} L_j. \tag{14}$$

Although it is not strictly necessary, each L_j is usually associated with a particular coordinate direction. As an example (9) can be written in the form ($L_1 \equiv L_x$; $L_2 \equiv L_y$; $L_3 \equiv L_z$)

$$\frac{\partial c_i}{\partial t} = (L_x + L_y + L_z)\, c_i(\mathbf{x}, t) + f_i(c_1, ..., c_p, \mathbf{x}, t). \tag{15}$$

Once the elemental components L_j have been identified, the next step is to determine their equivalent discrete representation in the computational domain. First, the continuously varying concentration field must be approximated at the r computational points by the values $\mathbf{c}_i = (c_i(\mathbf{x}_j, t); j = 1, 2, ..., r)$. At each of the grid points, the spatial derivative L_j must be replaced by its discrete approximation. The net result is the replacement of the scalar operation, $L_j c_i$, distributed over the physical domain, by the matrix product $\mathbf{A}_j \mathbf{c}_i$. In this formulation the elements of $\mathbf{A}_j$ depend on the particular discretization scheme and its coupling of adjacent grid point values. Given the numerical equivalents of each L_j, they then must be combined in a sequence that approximates the system as a whole. There are two common ways to accomplish this; one is to use Alternating Direction Implicit (ADI) schemes and the

other employs Locally One-Dimensional (LOD) or fractional step methods. The most widely known splitting procedure is the ADI algorithm which advances the concentration field a single time step Δt from the level n to time level $n+1$ using the sequence [25–27],

$$\mathbf{c}_i^* - \mathbf{c}_i^n = \frac{\Delta t}{2}\mathbf{A}_x\mathbf{c}_i^* + \Delta t[\tfrac{1}{2}\mathbf{A}_x + \mathbf{A}_y + \mathbf{A}_z]\,\mathbf{c}_i^n + \Delta t\,\mathbf{f}_i^n, \tag{16}$$

$$\mathbf{c}_i^{**} - \mathbf{c}_i^n = \frac{\Delta t}{2}[\mathbf{A}_x\mathbf{c}_i^* + \mathbf{A}_y\mathbf{c}_i^{**}] + \Delta t[\tfrac{1}{2}(\mathbf{A}_x + \mathbf{A}_y) + \mathbf{A}_z]\,\mathbf{c}_i^n, \tag{17}$$

$$\mathbf{c}_i^{***} - \mathbf{c}_i^n = \frac{\Delta t}{2}[\mathbf{A}_x\mathbf{c}_i^* + \mathbf{A}_y\mathbf{c}_i^{**} + \mathbf{A}_z\mathbf{c}_i^{***}] + \frac{\Delta t}{2}[\mathbf{A}_x + \mathbf{A}_y + \mathbf{A}_z]\,\mathbf{c}_i^n, \tag{18}$$

where $\mathbf{c}_i^*$, $\mathbf{c}_i^{**}$ are the intermediate results and $\mathbf{c}_i^{***}$ is an $O(\Delta t^3)$ approximation to $\mathbf{c}_i^{n+1}$. An alternate, but equivalent representation, that is more suited to practical problems, especially those involving steady state applications, is to solve for the concentration increment using

$$\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_x\right](\mathbf{c}_i^* - \mathbf{c}_i^n) = \Delta t[\mathbf{A}_x + \mathbf{A}_y + \mathbf{A}_z]\,\mathbf{c}_i^n + \Delta t\,\mathbf{f}_i^n, \tag{19}$$

$$\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_y\right](\mathbf{c}_i^{**} - \mathbf{c}_i^n) = \mathbf{c}_i^* - \mathbf{c}_i^n, \tag{20}$$

$$\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_z\right](\mathbf{c}_i^{***} - \mathbf{c}_i^n) = \mathbf{c}_i^{**} - \mathbf{c}_i^n. \tag{21}$$

By eliminating the intermediate results from (19)–(21) the ADI solution sequence can be written in the factored form [26]

$$\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_x\right]\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_y\right]\left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_z\right](\mathbf{c}_i^{n+1} - \mathbf{c}_i^n) = \Delta t[\mathbf{A}_x + \mathbf{A}_y + \mathbf{A}_z]\,\mathbf{c}_i^n + \Delta t\,\mathbf{f}_i^n. \tag{22}$$

Briley and McDonald [27] discuss the computational implementation of these techniques and in particular the use of linearization procedures for solving nonlinear partial differential equations. Apart from accuracy considerations, implicit discretization procedures usually allow arbitrarily large integration steps. When splitting techniques are used to solve transient problems, the maximum allowable time step is often constrained by the treatment of intermediate level boundary conditions. Weare [28] and Briley and McDonald [27] present analyses of the errors arising from different formulations of the boundary conditions. Unfortunately, ADI procedures are not ideally suited to solving the atmospheric diffusion equation because the coupling between the chemistry and transport in (16) imposes unreasonable time step limitations. In addition, since each term of the governing

differential equation is represented in each fractional step the memory paging overheads can become excessive.

An alternative approach is to use the method of fractional steps introduced by Yanenko [29] and described in Marchuk [30, 31] and Yanenko *et al.* [32]. Only the homogeneous Cauchy problem will be considered here. We discuss how nonlinear reactions can be included later. For the transport alone, the locally one-dimensional approximations, using Crank–Nicholson time integration, are given by

$$\mathbf{c}_i^{n+1} = \prod_{j=1}^{3} \left[\mathbf{I} - \frac{\Delta t}{2}\mathbf{A}_j\right]^{-1} \left[\mathbf{I} + \frac{\Delta t}{2}\mathbf{A}_j\right] \mathbf{c}_i^n \equiv \prod_{j=1}^{3} \mathbf{T}_j^n \mathbf{c}_i^n \equiv \mathbf{T}^n \mathbf{c}_i^n. \tag{23}$$

The principal difference between this formulation and the ADI scheme (20)–(22) is that the solution is advanced only in one coordinate direction at a time. Detailed discussions of the relationships between the two approaches are presented in Morris [33], Gourlay and Mitchell [34], Gourlay [35], and Gottlieb [36]. One important observation that can be made is that there are two sources of error in the fully discrete fractional step algorithm—the intrinsic error involved with operator splitting and the discretization errors associated with the operator approximations. In general these errors interact in a complex fashion. Crandall and Majda [37] have analyzed the stability, accuracy, and convergence of the basic fractional step algorithm when used to obtain discontinuous solutions of scalar conservation equations.

The formal order of the temporal approximation (23) can be developed by expanding the operators $\mathbf{T}_j^n$ in powers of Δt to give [31]

$$\mathbf{T}_j^n = \mathbf{I} - \Delta t\, \mathbf{A}_j^n + \frac{\Delta t^2}{2!}(\mathbf{A}_j^n)^2 - \frac{\Delta t^3}{3!}(\mathbf{A}_j^n)^3 + \cdots. \tag{24}$$

If $\mathbf{A}^n = \mathbf{A}_x^n + \mathbf{A}_y^n + \mathbf{A}_z^n$ then $\mathbf{T}^n$ is given by

$$\mathbf{T}^n = \mathbf{I} - \Delta t\, \mathbf{A}^n + \frac{\Delta t^2}{2!}\left[(\mathbf{A}^n)^2 + \sum_{\alpha=1}^{3}\sum_{\beta=\alpha+1}^{3} (\mathbf{A}_\alpha^n \mathbf{A}_\beta^n - \mathbf{A}_\beta^n \mathbf{A}_\alpha^n) + \cdots\right] + O(\Delta t^3). \tag{25}$$

Thus the split operator difference scheme will be second order accurate only if the split operators $\mathbf{A}_\alpha^n$ and $\mathbf{A}_\beta^n$ commute; otherwise, it is only of first order. To obtain second order accuracy, it is necessary to reverse the order of the operators each alternate step to cancel the two noncommuting terms. The consecutive cycles are then

$$\mathbf{c}_i^n = \prod_{j=1}^{3} \mathbf{T}_j^n \mathbf{c}_i^{n-1} \tag{26}$$

and

$$\mathbf{c}_i^{n+1} = \prod_{j=3}^{1} \mathbf{T}_j^n \mathbf{c}_i^n. \tag{27}$$

The proof of the stability of these approximations is considerably simplified using the following lemmas [31].

LEMMA 1. *Consider a positive semi-definite matrix* $\mathbf{A}$, *i.e.*, $\langle \mathbf{A}\mathbf{c}_i, \mathbf{c}_i \rangle \geqslant 0$, *on the Euclidean space, then for any value of the parameter* $\lambda \geqslant 0$,

$$\|(\mathbf{I} + \lambda \mathbf{A})^{-1}\| \leqslant 1, \tag{28}$$

where $\mathbf{I}$ *is the identity matrix and* $\|\cdot\|$ *is the appropriate norm.*

LEMMA 2. *For any positive semi-definite matrix* $\mathbf{A}$ *and* $\lambda \geqslant 0$ *then*

$$\|(\mathbf{I} - \lambda \mathbf{A})(\mathbf{I} + \lambda \mathbf{A})^{-1}\| \leqslant 1. \tag{29}$$

Using (28) *and* (29) *it is possible to show that*

$$\|\mathbf{c}_i^{n+1}\| \leqslant \|\mathbf{c}_i^n\| \leqslant \cdots \leqslant \|\mathbf{c}_i^0\|. \tag{30}$$

These results ensure absolute stability and boundness of the solution provided that the discrete operator approximation $\mathbf{A}$ *is also positive semi-definite.*

Implementation of operator splitting for the atmospheric diffusion equation (neglecting chemistry) can be accomplished by further separation of the material transport into advection $(\mathbf{T})_a$, and diffusive components $(\mathbf{T})_d$. If this is done, then a second-order accurate solution is given by

$$\mathbf{c}_i^{n+1} = (\mathbf{T}_x)_a(\mathbf{T}_x)_d(\mathbf{T}_y)_a(\mathbf{T}_y)_d(\mathbf{T}_z)_a(\mathbf{T}_z)_d(\mathbf{T}_z)_d(\mathbf{T}_z)_a(\mathbf{T}_y)_d(\mathbf{T}_y)_a(\mathbf{T}_x)_d(\mathbf{T}_x)_a\, \mathbf{c}_i^{n-1}. \tag{31}$$

Depending on the numerical scheme chosen, it is possible to combine the advection and diffusion into one transport step in each direction. For the remainder of this section and Sections 6 and 7 we focus on the atmospheric diffusion equation in the absence of chemistry to develop the solution procedure for the advection and diffusion components. Equation (31) indicates that the atmospheric diffusion equation can be decomposed, by operator splitting, into a series of simpler one-dimensional problems. Consequently, primary attention will be given to the one-dimensional transport problem (the subscript i denoting species i is dropped for convenience),

$$\frac{\partial c}{\partial t} = Lc = \frac{\partial}{\partial x}\left(K_{xx}\frac{\partial c}{\partial x} - uc\right) \tag{32}$$

and its component parts over the same domain

$$\text{Advection:} \qquad \frac{\partial c}{\partial t} = L_a c = -\frac{\partial uc}{\partial x} \tag{33}$$

and

$$\text{Diffusion:} \qquad \frac{\partial c}{\partial t} = L_d c = \frac{\partial}{\partial x} K_{xx} \frac{\partial c}{\partial x}. \tag{34}$$

The basic objective of the remainder of this work is to identify numerical solution techniques that are compatible with the characteristics of the physical problem, computationally efficient, stable, and accurate. In addition it is important, from a practical point of view, that the methods can be easily implemented and minimize core storage requirements.

6. Formulation of the Numerical Solution

A wide class of numerical approximations to the spatial derivatives in (32) can be expressed in the form $\mathbf{H}(\partial v/\partial x) = \mathbf{B}v$, where $\mathbf{v}$ is the material flux at the r computational grid points. The matrices $\mathbf{H}$ and $\mathbf{B}$ are of dimensions $r \times r$ with elements set by the particular discretization scheme. For example, the standard second-order, centered difference formula would have $\mathbf{H} = \mathbf{I}$ and $\mathbf{B}$ the tridiagonal form $[-1\ 0\ 1]$. Given the material flux

$$\mathbf{v} = \mathbf{K}_{xx}\frac{\partial \mathbf{c}}{\partial x} - \mathbf{U}\mathbf{c} \tag{35}$$

then (32) can be written as an equivalent set of first-order problems

$$\begin{aligned} \mathbf{H}\frac{\partial \mathbf{c}}{\partial x} &= \mathbf{B}\mathbf{c}, \\ \mathbf{P}\frac{\partial \mathbf{c}}{\partial t} &= \mathbf{Q}\mathbf{v}, \end{aligned} \tag{36}$$

where $\mathbf{B}$, $\mathbf{H}$, $\mathbf{P}$, $\mathbf{Q}$ are large sparse matrices resulting from the particular discretization formulation and $\mathbf{K}_{xx}$ and $\mathbf{U}$ are diagonal matrices corresponding to the turbulent diffusion coefficients and advective velocity components at each grid point. Eliminating $\mathbf{v}$ the system can be expressed in the partitioned matrix form,

$$\left(\begin{array}{c|c} \mathbf{H} & \mathbf{0} \\ \hline \mathbf{Q}\mathbf{K}_{xx} & -\mathbf{P} \end{array}\right) \left\{\begin{array}{c} \dfrac{\partial \mathbf{c}}{\partial x} \\ \hline \dfrac{\partial \mathbf{c}}{\partial t} \end{array}\right\} = \left\{\begin{array}{c} \mathbf{B}\mathbf{c} \\ \hline \mathbf{Q}\mathbf{U}\mathbf{c} \end{array}\right\}. \tag{37}$$

The relationship between this formulation and the operator splitting approach introduced in Section 5 can be seen in the explicit representations

$$\text{Advection:} \qquad \frac{\partial \mathbf{c}}{\partial t} = -\mathbf{P}^{-1}\mathbf{Q}\mathbf{U}\mathbf{c} \equiv (\mathbf{T}_x)_a \mathbf{c}, \tag{38}$$

$$\text{Diffusion:} \qquad \frac{\partial \mathbf{c}}{\partial t} = \mathbf{P}^{-1}\mathbf{Q}\mathbf{K}_{xx}\mathbf{H}^{-1}\mathbf{B}\mathbf{c} \equiv (\mathbf{T}_x)_d \mathbf{c}. \tag{39}$$

These two results can be combined to give the complete numerical approximation for $\partial \mathbf{c}/\partial t$,

$$\frac{\partial \mathbf{c}}{\partial t} = \mathbf{P}^{-1}\mathbf{Q}\{\mathbf{K}_{xx}\mathbf{H}^{-1}\mathbf{B} - \mathbf{U}\}\mathbf{c} \equiv \mathbf{T}_x\mathbf{c}. \tag{40}$$

While easy to implement, a direct solution of (40) has a number of drawbacks, the most serious of which is the need to evaluate $\mathbf{H}^{-1}$ and $\mathbf{P}^{-1}$. Normally both $\mathbf{H}$ and $\mathbf{B}$ are tridiagonal, unfortunately there is no guarantee that this property is preserved under the inverse transformation. If $\mathbf{H}^{-1}$ and $\mathbf{P}^{-1}$ are full matrices, then the operation count for evaluating the matrix products becomes quite large. The choice of whether to use a direct solution or a block tridiagonal LU decomposition depends to a large extent on the number of right-hand sides. A single evaluation of $\mathbf{T}_x$ followed by many products of the form $\mathbf{T}_x\mathbf{c}_i$, $i = 1, 2, \ldots, p$, may be more economical. The decision as to which is the more appropriate approach depends on the number of grid points, chemical species and a detailed operation count for each procedure. For the tests to be described in this paper block tridiagonal solution procedures were applied to the system (37). The resulting set of equations, subject to the appropriate boundary conditions, can be solved by standard methods. In subsequent sections the vector notation for $\mathbf{c}$, indicating the numerical approximations to $c(x, t)$ at the r grid points, will be omitted for clarity.

7. Solution of the Advective Transport Step T_a

There is an extensive literature that describes techniques suitable for solving the hyperbolic problem (33) [9, 38–41]. Most of the approaches fall into five basic categories: finite difference, variational, particle-in-cell, spectral and special purpose procedures. On the basis of a preliminary survey, seven methods were identified for detailed evaluation. These schemes were: the flux corrected transport algorithm (SHASTA) [42–44], compact differencing methods [45–50], finite element methods [51–53], the zero-average phase-error technique [54], upwind differencing [9], the Crowley [55] technique and finally the scheme of Price *et al.* [56]. These methods were used as described in the literature except for the finite element scheme that was applied to the conservative formulation of the advection equation.

The particular finite element model used in this study employs a Galerkin formulation and linear basis functions. With this technique, approximations to the concentration and velocity fields are expressed in terms of time-varying coefficients $\alpha_j(t)$, $\beta_j(t)$ and piecewise continuous basis function $\phi_j(x)$,

$$c(x, t) = \sum_{j=1}^{r} \alpha_j(t)\,\phi_j(x), \tag{41}$$

$$u(x, t) = \sum_{j=1}^{r} \beta_j(t)\,\phi_j(x), \tag{42}$$

where

$$\begin{aligned}\phi_j(x) &= \frac{x - x_{j-1}}{x_j - x_{j-1}}; \qquad x_{j-1} \leqslant x \leqslant x_j, \\ &= \frac{x_{j+1} - x}{x_{j+1} - x_j}; \qquad x_j \leqslant x \leqslant x_{j+1}, \\ &= 0; \qquad x < x_{j-1} \text{ or } x > x_{j+1}.\end{aligned} \tag{43}$$

Equation (43) describes a set of linear basis function that vanish outside the interval $[x_{j-1}, x_{j+1}]$. Using these functions, the Galerkin method requires that for all ϕ_j,

$$\left\langle \phi_j, \left\{ \frac{\partial \alpha_q \phi_q}{\partial t} + \frac{\partial}{\partial x}(\beta_s \phi_s \alpha_q \phi_q) \right\} \right\rangle = 0. \tag{44}$$

By expanding the inner product (44), the following set of ordinary differential equations in the dependent variable $\alpha_q(t)$ can be derived

$$M_{jq} \frac{d\alpha_q(t)}{dt} + \beta_s(t)\, N_{jqs}\, \alpha_q(t) = 0, \tag{45}$$

where

$$M_{jq} = \int \phi_j(x)\, \phi_q(x)\, dx, \tag{46}$$

$$N_{jqs} = \int \left[\phi_j(x)\, \phi_q(x) \frac{\partial \phi_s(x)}{\partial x} + \phi_j(x)\, \phi_s(x) \frac{\partial \phi_q(x)}{\partial x} \right] dx. \tag{47}$$

To compare the solution schemes, some idealized test problems with known solutions were selected. Particular attention was given to the harmonic content of each test case. A concentration distribution containing components with wavelengths shorter than the characteristic grid spacing represents a difficult test for any advection scheme. If little numerical or physical diffusion is present, an initial profile with sharp corners and steep sides should remain intact as it is transported by the velocity field. Test problems were also chosen to allow simultaneous and individual solutions of both transport components. In addition to the accuracy considerations judged by the important attributes of mass conservation, minimal dispersion and minimal pseudo-diffusion, additional constraints in choosing a numerical method arise as a result of the availability of computational resources. Execution time, storage requirements, ease of understanding, and implementation must also be considered since the most accurate scheme may also be the least efficient.

A series of test problems, listed in Table I was used to evaluate the schemes. The velocity was constant at 5 km/hr, the time step at 0.1 hours. The Courant number, C,

TABLE I

Test Problems for Advection Equation

Wave form	Function	Fourier spectrum
Square	$c(x,0)=\begin{cases}1 & \lvert x\rvert \leqslant \frac{\theta}{2} \\ 0 & \lvert x\rvert > \frac{\theta}{2}\end{cases}$	$\dfrac{\sin\frac{\omega\theta}{2}}{\frac{\omega\theta}{2}}$
Triangle	$c(x,0)=\begin{cases}1-\lvert x\rvert/\theta & \lvert x\rvert \leqslant \theta \\ 0 & \lvert x\rvert > \theta\end{cases}$	$\dfrac{\theta \sin\left(\frac{\omega\theta}{2}\right)}{\left(\frac{\omega\theta}{2}\right)^2}$
Gaussian	$c(x,0)=\exp\left[-\Pi\left(\frac{x}{\theta}\right)^2\right]$	$\theta\sqrt{\Pi}\exp\left[-\frac{1}{2}(\theta\omega)^2\right]$

θ—Volume/unit width ratio for the wave form.

was 0.25, which is less than the stability limit for all schemes. These parameters were chosen to be representative of meteorological conditions over a typical urban airshed. The results of the tests are summarized in Table II and shown in Figs. 1–3. Further detailed testing with a range of sample problems narrowed the solution methods to the SHASTA technique and a class of techniques that use linear finite elements or compact differences together with Crank–Nicholson time integration.

7.1. *Preservation of Positive Quantities and Filtering Schemes*

During the course of the testing it became obvious that in order to develop a scheme that preserves peaks, retains positive quantities, and does not severely diffuse sharp gradients, an additional step must be performed to minimize the effect of dispersive waves. As noted by Kreiss and Oliger [57], the basic problem with conventional Galerkin formulations is that they result in nondissipative, discrete approximations when applied to hyperbolic equations. What is required is a procedure for damping out the small scale perturbations before they can corrupt the basic solution. There are several different filtering procedures that can be applied: (1) adding diffusions terms to the basic equation [58], (2) expansion of the concentration field in orthogonal functions with a recombination that omits high wave numbers [59], (3) numerical filtering where the grid point value is replaced by an average formed from surrounding values, (4) inclusion of a dissipative term in the problem formulation [60, 61].

At the simplest level, one approach is to set any negative concentration to zero or a very small positive number following each advection step. This procedure is demonstrated using the finite element method with a square wave in Fig. 4a. While trivial to implement, this correction scheme can induce violations of mass conser-

TABLE II

Summary of Results of Advection Tests for Different Initial Distributions*

Numerical scheme	Test Square (S) Gaussian (G) Triangle (T)	Extreme value		Change in mass (%)	Relative computational time
		Maximum	Minimum		
Upwind [9]	S	0.755	0.0	0.0	1.0
	T	0.693	0.0	−0.03	
	G	0.635	0.0	−0.01	
Price [56]	S	1.463	−0.390	−0.93	1.2
	T	0.971	−0.086	−0.50	
	G	1.108	−0.216	0.26	
Fromm [54]	S	1.084	−0.067	−0.05	1.8
	T	0.918	−0.015	0.28	
	G	0.964	−0.006	0.07	
Crowley [55]	S	1.219	−0.222	−2.02	2.0
	T	0.932	−0.017	0.28	
	G	0.990	−0.001	0.07	
Finite element [Text]	S	1.218	−0.382	−10.27	2.0
	T	0.953	−0.025	0.17	
	G	0.999	−0.001	0.16	
SHASTA [42–44]	S	0.997	0.0	0.0	5.2
	T	0.875	0.0	0.20	
	G	0.900	0.0	0.04	

* Results are at the end of 80 time steps.

vation. Mahlman and Sinclair [62] attempted to correct this problem by using a method called "downstream borrowing." In this scheme, implemented at the end of each time step, negative values are reset to zero by borrowing material from the downstream grid cell so that mass is conserved. In the event that the downstream cell does not contain an adequate amount of material to prevent both cell concentrations from becoming negative, the deficit is borrowed from the upstream cell. With higher-order schemes it is occasionally necessary to borrow mass from the second cells away from the one containing negative c. Although this filling procedure always acts to preserve the total mass in the system, it systematically acts to reduce the mean square concentration. Filling is thus equivalent to adding a nonlinear diffusion term. An example of the application of this procedure is shown in Fig. 4b again using the finite element method with a square wave initial condition.

Boris and Book [42, 43] and van Leer [63] have introduced different approaches to the design of filtered second-order schemes. Their algorithms substantially inhibit

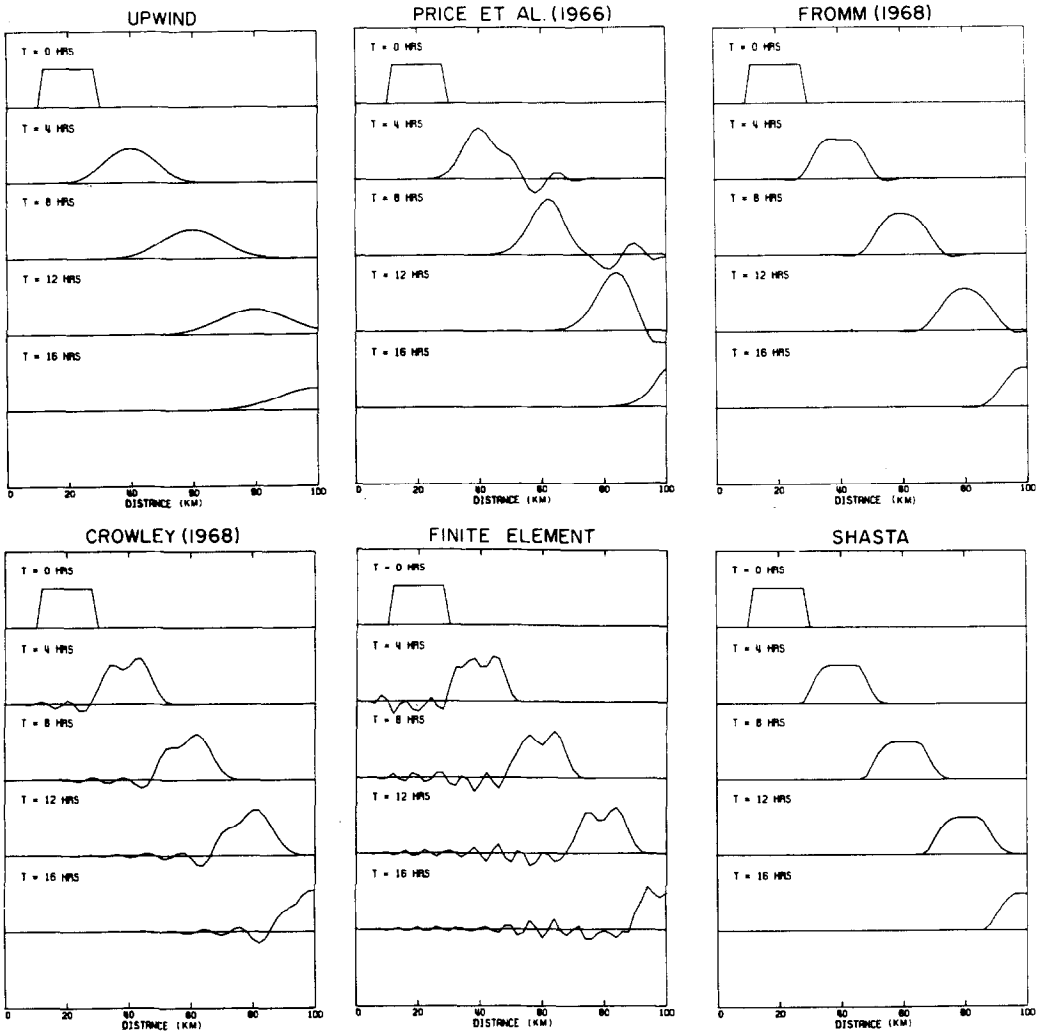

FIG. 1. Results of advection tests using square wave form.

or eliminate computational noise in regions of sharp gradients by using nonlinear smoothing techniques. The principal disadvantage of both methods is that there are substantial amplitude penalties associated with sharply peaked waves. When the SHASTA scheme of Boris and Book is used to advect a triangle, after a few steps the apex is typically severely truncated. However, once this has occurred, the distribution is transported with little change.

Recently Forester [64] introduced a very simple nonlinear filter designed to be used in conjunction with second and higher-order methods. Computational noise is

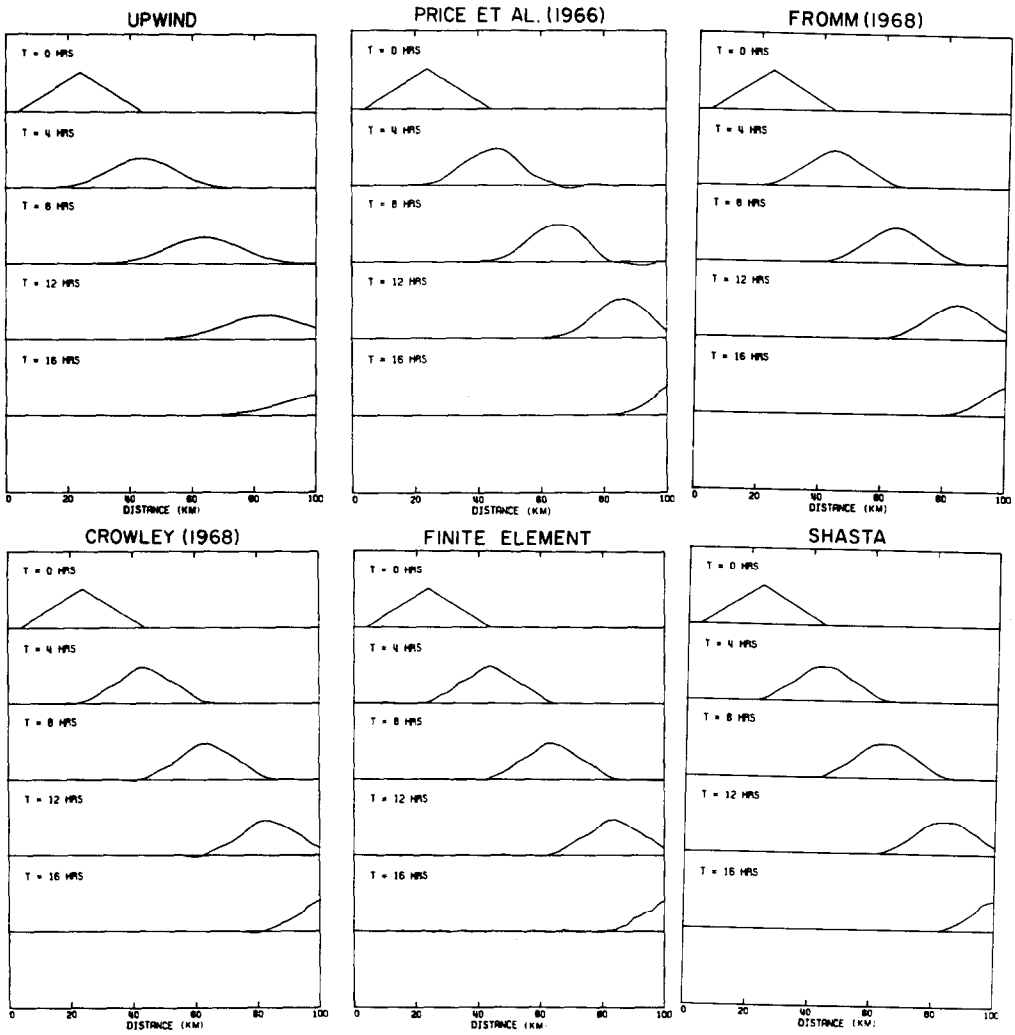

FIG. 2. Results of advection tests using triangular wave form.

minimized without incurring the amplitude penalty of either the SHASTA or van Leer techniques. When coupled with high-order schemes, the Forester method requires less than one-third of the mesh points of the SHASTA scheme to treat the extremes of sharply peaked waves. Positive concentrations are also preserved. The noise generated by the finite difference approximations of (33) is suppressed in the Forester method by a nonlinear filter that locally transforms (33) into

$$\frac{\partial c}{\partial t} + \frac{\partial uc}{\partial x} = \frac{\partial}{\partial x} K_n \frac{\partial c}{\partial x}, \tag{48}$$

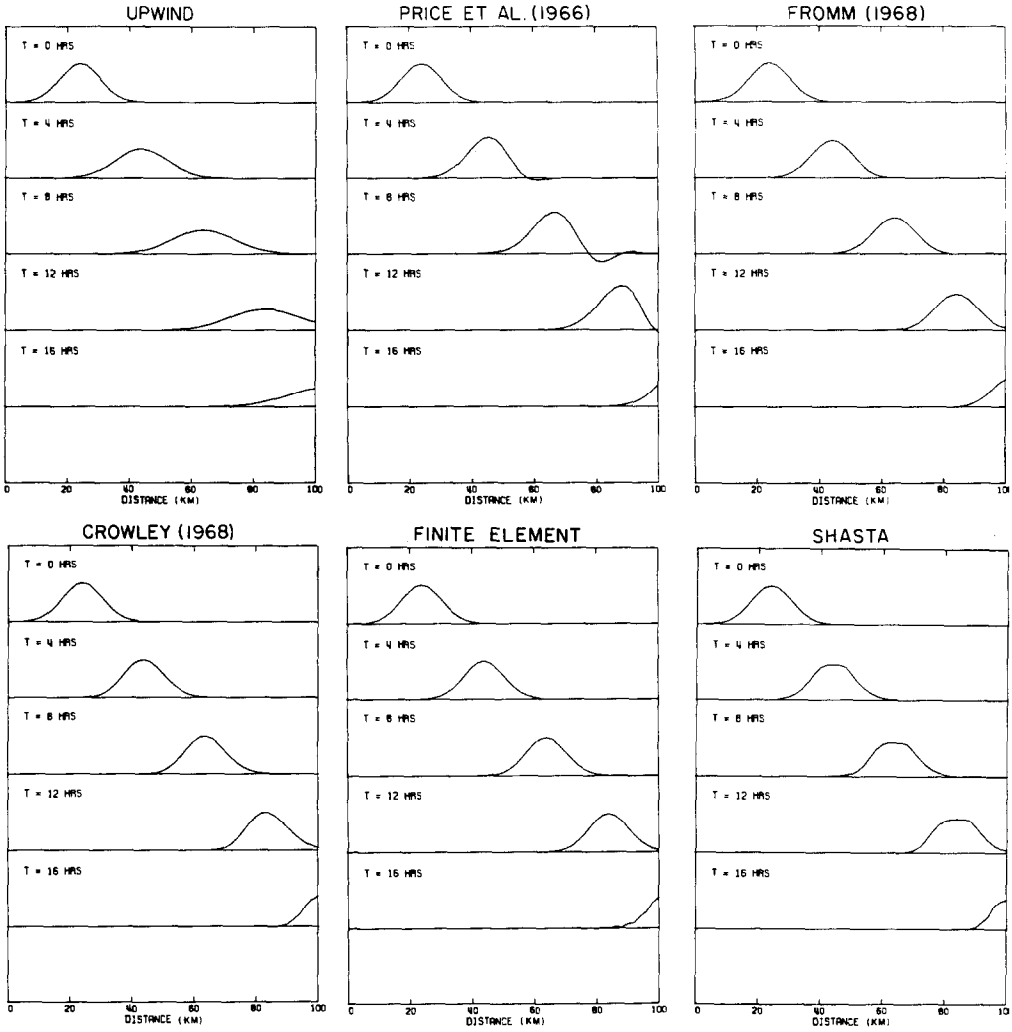

FIG. 3. Results of advection tests using Gaussian wave form.

where K_n is the diffusion coefficient associated with the filtering process. After the solution is advanced a time step, a set of empirically based criteria is used to decide if the term should remain or be removed. The filter for (33) is given by

$$c_j^{k+1} = c_j^k + \frac{K_f}{2}[(c_{j+1} - c_j)(\psi_j + \psi_{j+1}) - (c_j - c_{j-1})(\psi_j + \psi_{j-1})]^k, \qquad (49)$$

where c_j^{k+1} is the value of c_j after k applications of the filter and K_f is the weighting coefficient associated with the filtering process. The ψ_j's can only assume a value of 0

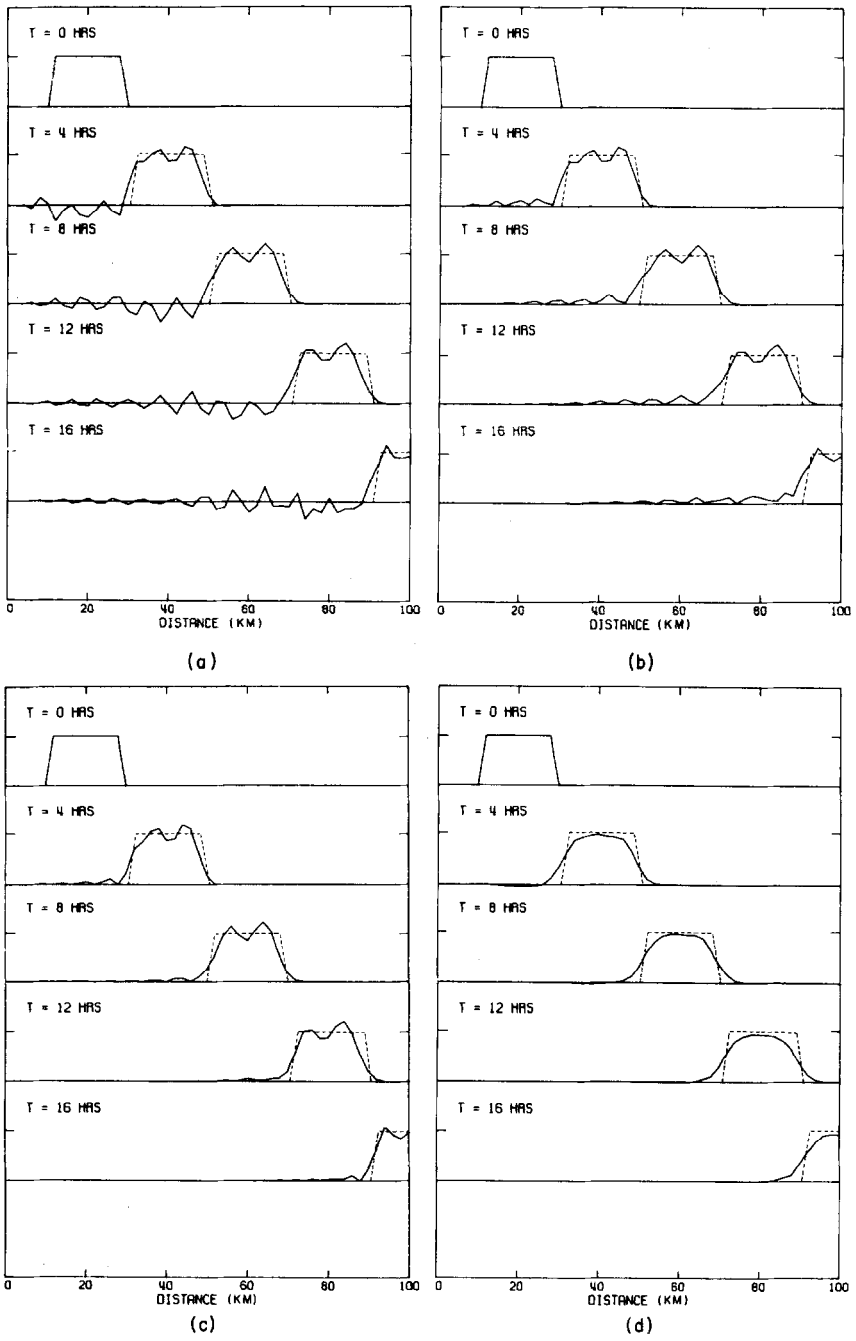

FIG. 4. Application of difference schemes to maintain concentration positivity. (a) Original linear finite element solution. (b) Absolute value $|c|$. (c) Downstream borrowing. (d) Nonlinear filter.

or 1 and determine the points at which smoothing occurs. Clearly if all are zero, no filtering takes place. For the condition $\psi_j = 1$, (51) takes a form that is analogous to the three-point difference expression for the diffusion term,

$$c_j^{k+1} = c_j^k + K_f[c_{j+1} - 2c_j + c_{j-1}]^k. \tag{50}$$

A key element of the filter application is the selection of the points in the grid mesh at which to set $\psi = 1$. Initially, all ψ are set to zero. Consider a point j and an interval $[j-m, j+m+1]$. On this interval the function S_e is evaluated using

$$S_e = \text{sgn}[c_e - c_{e-1}]; \qquad e = j-m, j-m-1, \ldots, j, j+1, \ldots, j+m+1, \tag{51}$$

where

$$\begin{aligned} \text{sgn}(c) &= +1, \qquad \frac{c}{|c|} \geqslant 0, \\ &= -1, \qquad \frac{c}{|c|} < 0. \end{aligned} \tag{52}$$

At mesh point j there is an extremum of c_j if S_j and S_{j+1} are of opposite sign. The distribution of c on the interval $[j-m, j+m+1]$ is considered to be smooth if $S_{j+1}, \ldots, S_{j+m+1}$ have the same sign and all $S_{j-1}, \ldots, S_{j-m}$ are of opposite sign to S_{j+1}. If this occurs, the values of ψ are left unchanged and no smoothing is applied to c_j. No tests for sign continuity of $S_j, \ldots, S_{j-m-1}$ are performed unless c_j is an extremum. These cases are illustrated in Fig. 5. If the slope or sign continuity does not hold for the m values of S on each side of the extremum in c_j, ψ is reset to 1 for the range of i from $i-l$ to $i+l$. To ensure that the mesh points at which ψ is nonzero in fact denote regions that contain computational noise, it is necessary to select the proper magnitudes for l and m. The value of m is chosen to be representative of one-half the wavelength of the lowest-frequency noise waves; l simply must be large enough to permit nonzero c values to be continuous.

For many high-order advective schemes nonlinear effects tend to drive the wavelength of the computational noise toward the limit of two mesh intervals, this can be seen in the results shown in Figs. 1–4. In general, the structure of the dispersive waves depends on the advection algorithm, its performance for different Courant numbers, and the nature of the concentration gradients. Values of m, l, K_f and the number of iterations required to satisfy the error tolerance must be determined empirically. For the above fourth-order schemes the values chosen were $m = 4$, $l = 2$, $K_f = 0.2$, and the number of iterations set to 2 and 3 for local Courant numbers less than 0.5 and greater than 0.5, respectively. An application of the filter, together with the finite element scheme, to the square-wave propagation problem is shown in Fig. 4d. There is clearly a significant improvement over the results displayed in Fig. 1.

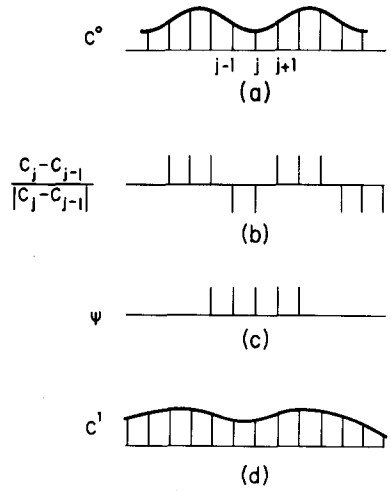

FIG. 5. Steps in the application of the discrete noise filter. (a) Initial distribution c^0. (b) Evaluation of the normalized derivatives. (c) Establishment of ψ function. (d) Resulting distribution after one filter application c^1.

7.2. *Conservation Properties of Different Advection Methods*

With the addition of the nonlinear filter, the performance of the finite element scheme improved to the point where it was useful to perform a quantitative comparison between it and the SHASTA method. In particular, it was important to assess the ability of each scheme to preserve mass, concentration gradients etc. A variety of initial distribution and velocity fields were used to test the techniques. The triangle test problem used in previous sections of this work has the property that

$$\frac{\partial}{\partial t}\int c\,dx = 0, \tag{53}$$

$$\frac{\partial}{\partial t}\int c^2\,dx = 0, \tag{54}$$

$$\frac{\partial}{\partial t}\int c^4\,dx = 0, \tag{55}$$

$$\frac{\partial}{\partial t}\int \left(\frac{\partial c}{\partial x}\right)^2 dx = 0, \tag{56}$$

$$\frac{\partial}{\partial t}\int \left(\frac{\partial^2 c}{\partial x^2}\right)^2 dx = 0. \tag{57}$$

Each of these integrals was evaluated numerically using, in the case of (56) and (57), standard finite difference approximations to the derivatives. While a numerical scheme should ideally conserve both mass (53) and mean square mass (54), diffusive

TABLE III

Results of Advection of Triangular Wave Form after 80 Time Steps

Numerical scheme	ERROR (%) = 100 (calculated/exact − 1)				
	$\int c\,dx$	$\int c^2\,dx$	$\int c^4\,dx$	$\int \left(\frac{\partial c}{\partial x}\right)^2 dx$	$\int \left(\frac{\partial^2 c}{\partial x^2}\right)^2 dx$
Fourth order	0.20	0.00	−0.44	−3.15	−28.17
SHASTA	0.20	−0.92	−5.51	−12.40	−97.75

effects tend to damp the latter quantity. The ability of a numerical scheme to maintain peak values is measured by (55), growth or decay of local gradients by (56) and change of profile curvature by (57). In a more general context, it should be noted that integrals (53) and (54) are analytically conserved in more complex source-free and nondiffusive flows. If gradient reducing diffusion terms are not included in calculations with more complicated flows, (56) and (57) tend to increase with time from either numerical distortion or from a physically real cascade to smaller spatial scales [62]. In practice, it is often difficult to establish which of these two effects is dominant. Since (56) and (57) are conserved in the test problem, any increase in their magnitude with time must be attributed to numerical errors. If this occurs, extra damping would be required to suppress the growth of the errors.

Errors in preserving the conservation properties for the SHASTA and fourth-order schemes are displayed in Table III. The SHASTA scheme performs poorly at maintaining peak values and, in addition, has the most diffusive effect on the profile. By comparison, the finite element method exhibits little diffusion.

A test of the capability of each scheme to handle variable velocity fields was also devised for the system

$$\frac{\partial c}{\partial t} + \frac{\partial uc}{\partial x} = 0; \qquad x \in [0, 100], \tag{58}$$

where the velocity field $u(x)$ is given by

$$u(x) = \frac{x+1}{20}. \tag{59}$$

The exact solution, $c_e(x, t)$ of this system is

$$c_e(x, t) = 0.1(x+1)\exp\left[-\frac{t}{10}\right]. \tag{60}$$

Initial and boundary condition for the problem are $c(x, 0) = c_e(x, 0)$ and $c(0, t) = c_e(0, t)$. Each numerical scheme used a grid size $\Delta x = 2$ km, and a time step $\Delta t = 0.2$ hours. Under these conditions, the maximum Courant number is 0.5. After

TABLE IV

Errors in Concentration Predictions after 120 Time Steps for a Spatially Varying Velocity Field

Numerical scheme	Error (%) at			
	$x = 24$	$x = 50$	$x = 76$	$x = 100$
Fourth order	0.87	−0.08	0.01	0.03
SHASTA	1.18	1.20	1.20	1.87
Exact solution	0.0338	0.0690	0.1042	0.1367

120 time steps (24 hours), the errors were calculated and the results are shown in Table IV. While each scheme performed reasonably well, the finite element method produced better predictions at all spatial locations.

A rather difficult advection calculation, in two dimensions, is the rotating cone problem introduced by Crowley [55] and Molenkamp [65]. The test consists of solving the axisymmetric advection problem

$$\frac{\partial c}{\partial t} + \omega \frac{\partial c}{\partial \theta} = 0, \tag{61}$$

where θ is the angular coordinate, and ω the angular velocity around the axis of rotation. The exact solution of (61) is given by $c(r, \theta, t) = c^0(r, \theta - \omega t)$, where c^0 is the initial distribution of c. Since there is no physical diffusion, the shape c^0 should remain unchanged upon rotation. The Crowley problem consists of solving (61) in rectangular coordinates where the rotation is anticlockwise about the origin.

TABLE V

Summary of Results of Two-Dimensional Cone in a Circular Wind Field ($C_x = C_y = 0.5$)

Numerical scheme	1/4 Revolution		1 Revolution	
	Maximum value	Minimum value	Maximum value	Minimum value
Fromm	0.7400	−0.0218	0.5466	−0.0288
Crowley	0.8478	−0.0586	0.7283	−0.1279
Finite element	0.8731	−0.0335	0.8645	−0.0545
SHASTA	0.6670	0.0	0.5118	0.0
Exact solution	1.0000	0.0	1.000	0.0

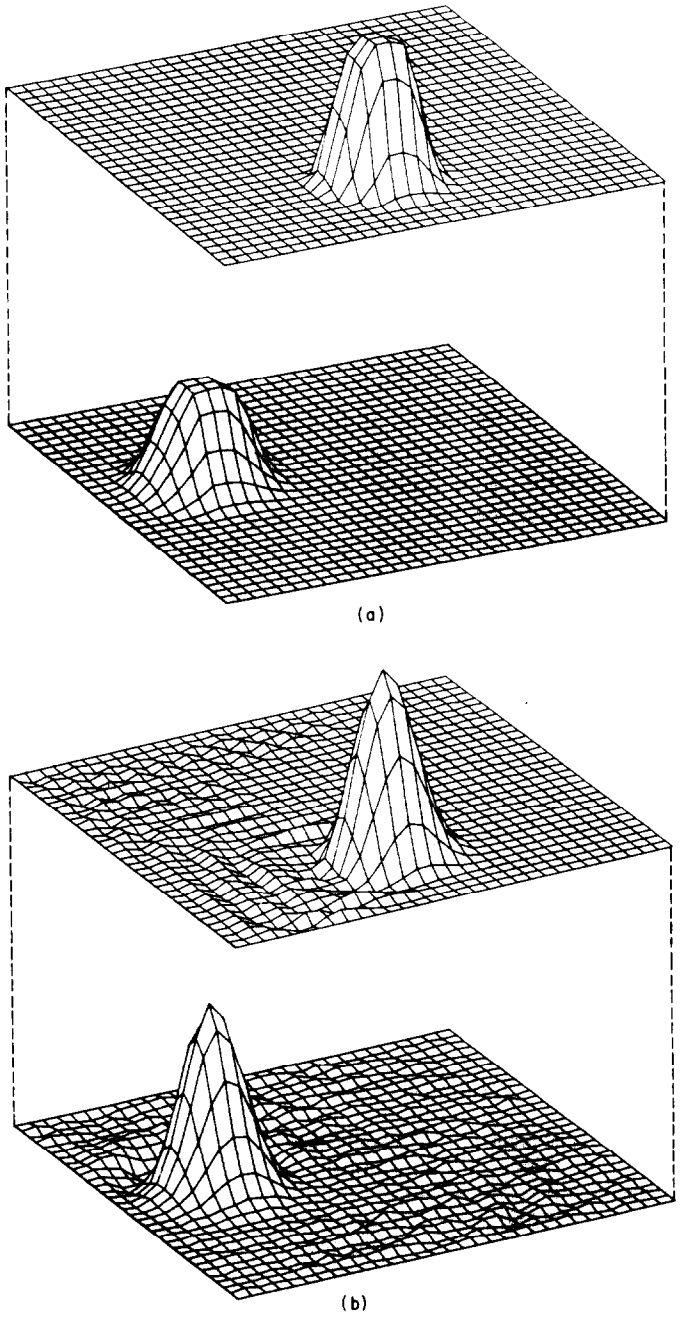

FIG. 6. Results of Crowley test problem for a quarter and complete revolution of a cone using (a) SHASTA method and (b) linear finite element scheme (without filtering step).

Under these conditions, the velocity components are given by $u = -y\omega$, $v = x\omega$ and

$$\frac{\partial c}{\partial t} - \frac{\partial \omega y c}{\partial x} + \frac{\partial \omega x c}{\partial y} = 0. \tag{62}$$

The method of fractional steps was used to solve the problem on a 32×32 grid with $\Delta x = \Delta y = 1$ km, $\Delta t = 0.5$ hrs and $\omega = 0.0626$ rad/hr. A conical distribution, centered initially at $(-8, 0)$, of base radius 4 and with $c_{\max} = 1$, $c_{\min} = 0$ was used to describe c^0. The results of the experiment, summarized in Table V, are displayed in Fig. 6; the conclusions are similar to the last test case. The peak truncation problem, characteristic of SHASTA, is particularly apparent. From a practical point of view, it is encouraging to note that the amplitudes of the dispersive waves associated with unfiltered finite element scheme are quite small.

8. Solution of the Diffusive Transport Step T_d and Boundary Condition Treatment

Previous sections were devoted to the implementation and testing of a suitable scheme for the advection equation. The contribution to species transport from turbulent diffusion depends on the coordinate direction. In the horizontal plane, transport is dominated by advection and so a simple, explicit three-point finite difference form [9] can be adopted for $(\mathbf{T}_x)_d$ and $(\mathbf{T}_y)_d$. A linear finite element scheme, with Crank–Nicholson time differencing, was used for $(\mathbf{T}_z)_d$. This removed the time step limitation of an explicit method and enabled the use of variable mesh spacing to resolve vertical concentration gradients.

The boundary of the grid is usually placed at the limits of the available data or far from the main calculation area. Boundary conditions are termed either inflow or outflow, depending on the direction of flow relative to the grid region. Often in fluid flow problems, the concentration at the inflow boundary is known and can be specified as a function of time. The outflow boundary is generally not known and therefore must be calculated. This boundary condition is sometimes called a "computational boundary condition" for this reason. Some helpful discussions of boundary conditions exist in the literature [66–73]. The boundary conditions used with (32) are

$$\text{inflow:} \qquad uc - K_{xx}\frac{\partial c}{\partial x} = uc_{\text{in}}, \tag{63}$$

$$\text{outflow:} \qquad -K_{xx}\frac{\partial c}{\partial x} = 0, \tag{64}$$

where c_{in} is the known concentration just outside of the inflow boundary. If it is assumed that advection is the dominant transport mechanism at the outflow

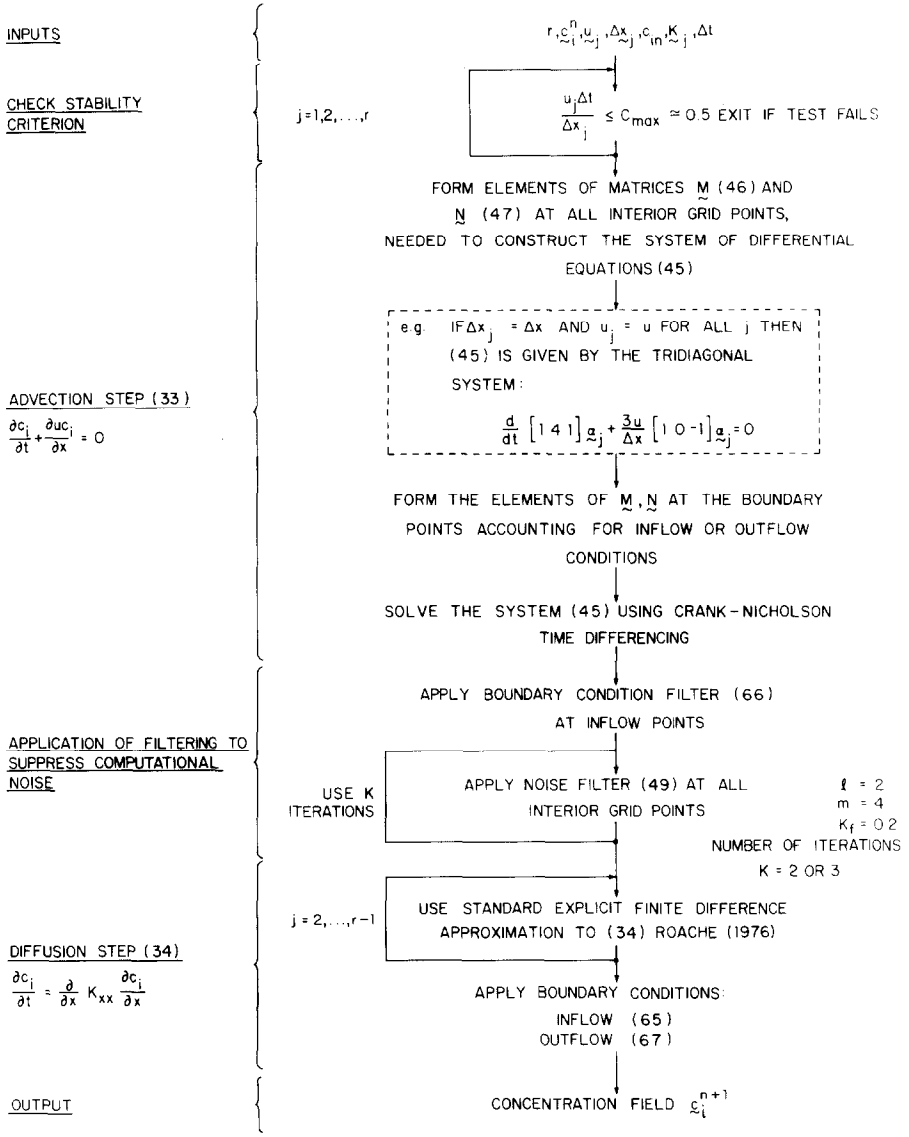

FIG. 7. Structure of the algorithm for solving the advection–diffusion equation for species transport in the x-direction.

boundary, then diffusive transport can be neglected. If the left end of the grid is an inflow boundary, then (63) can be represented as

$$u_1 c_1^{n+1} - \tfrac{1}{2}[(K_{xx})_1 + (K_{xx})_2] \frac{c_2^{n+1} - c_1^{n+1}}{\Delta x} = u_1 c_{\text{in}}^{n+1}, \tag{65}$$

which, in turn, can be solved explicitly for c_1^{n+1} since all other quantities are known. When using a multiple-step, advection–diffusion algorithm, (65) is used following the second (diffusion) step. A boundary value must also be set following the first (advection) step. The single condition $u_1 c_1 = u_1 c_{\text{in}}$ is used for this step in conjunction with a smoothing procedure at the point adjacent to the boundary point. This smoothing damps any waves that may be generated by the inflow boundary point. The simplest smoothing algorithm is

$$c_2^* = \tfrac{1}{2} c_2^{n+1} + \tfrac{1}{4}(c_1^{n+1} + c_3^{n+1}), \tag{66}$$

where c_2^* is the smoothed value of c^{n+1} at $j = 2$. A procedure analogous to the above can be applied to the right boundary. The concentration at an outflow boundary is influenced by information from the interior of the grid. Concentration gradients that are advected to the boundary must be preserved as they pass out of the grid. The simple choice of representing (64) by a zero gradient, i.e., $c_1 = c_2$ or $c_r = c_{r-1}$, where r is the right boundary point, was discarded due to its inability to preserve gradients. The approach adopted was to solve the advection equation (with zero diffusion) using a one-sided difference at the boundary:

$$\frac{c_r^{n+1} - c_r^n}{\Delta t} + \frac{u_r c_r^{n+1} - u_{r-1} c_{r-1}^{n+1}}{\Delta x} = 0. \tag{67}$$

This procedure preserves concentration gradients as they move out of the grid system as can be seen in the previous figures for the one dimensional test problems.

Figure 7 shows a flow diagram of the numerical solution of the advection and diffusion components of the atmospheric diffusion equation.

9. Numerical Solution of the Chemical Kinetics

In the previous two sections primary emphasis was placed on the transport components of the atmospheric diffusion equation. Equation (1) contains terms, f_i, $i = 1, 2, \ldots, p$, that describe the contributions to the rates of change of the p chemical species concentrations, $c_1, c_2, \ldots, c_p$, due to chemical reactions. At any one spatial point the rate of change of each species concentration resulting only from the

chemical kinetics can be described by a set of coupled, nonlinear ordinary differential equations,

$$\frac{dc_i}{dt} = f_i(c_1, c_2, ..., c_p, t); \qquad i = 1, 2, ..., p, \tag{68}$$

and associated initial conditions $c_i(0) = c_i^0$, $i = 1, 2, ..., p$.

There are two sources of difficulty that arise during the numerical solution of (68). One is minor and caused by the nonlinearities resulting from the polynomial form of the mass action rate laws. The more serious problem, however, arises as a result of the fact that in atmospheric systems there are reactions whose characteristic time scales differ by orders of magnitude. Such systems are often referred to as being "stiff." There are various definitions of what constitutes stiffness, the most common is of the form:

DEFINITION. The system (68) is said to be *stiff* if

(a) $\mathrm{Re}(\lambda_i) < 0; \qquad i = 1, 2, ..., p,$

and

(b) $(\max_i |\mathrm{Re}\, \lambda_i|)/(\min_i |\mathrm{Re}\, \lambda_i|) = R \gg 1,$

where R is the stiffness ratio and λ_i are the eigenvalues of the Jacobian matrix $\mathbf{J} = \partial \mathbf{f}/\partial \mathbf{c}$. A way to view the problem of stiffness is to write (68) in the form

$$\frac{dc_i}{dt} = a_i - b_i c_i, \tag{69}$$

where a_i is the production rate for species c_i and $b_i c_i$ is the loss rate. The reciprocal of b_i can be interpreted as the characteristic time for decay of species i. If a_i and b_i are constants then (69) can be solved to give

$$c_i(t) = \frac{a_i}{b_i} + \left[c_i(0) - \frac{a_i}{b_i} \right] \exp(-b_i t). \tag{70}$$

Expressed in this way, it can be seen that $1/b_i$ describes how quickly species c_i reaches its equilibrium value. Figure 8 presents a typical eigenvalue spectrum for atmospheric reaction mechanisms together with the characteristic reaction times $1/b_i$. Two features are readily apparent: one is the close correspondence, for many species, between the eigenvalues and the characteristic reaction times and the other is the extreme range $O(10^{12}$ min) of the spectrum.

In passing it is worthwhile to comment on the reason why some of the eigenvalues are so closely matched to the corresponding reaction times. Consider atomic oxygen (O), which has the fastest reaction time of any species in the system. An examination

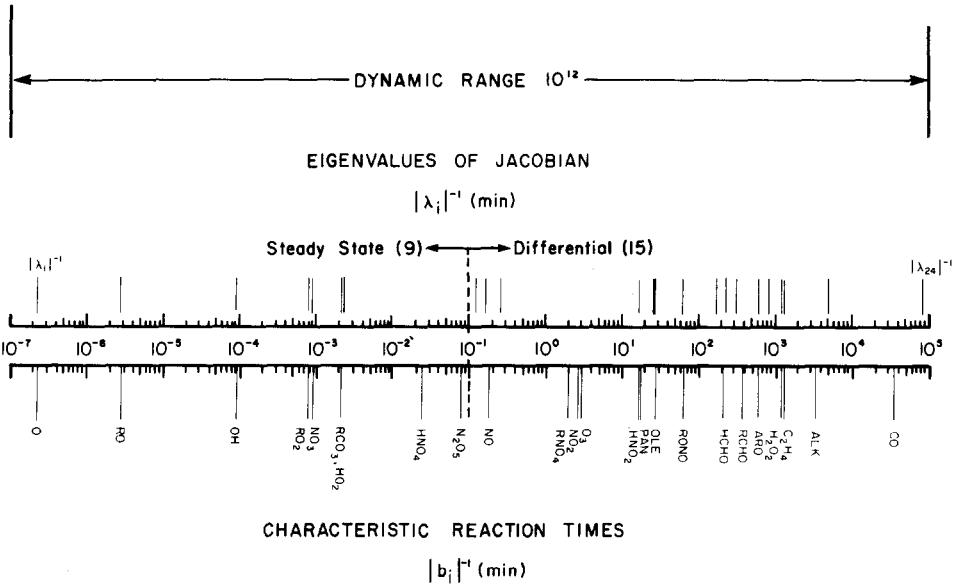

FIG. 8. Typical eigenvalue spectrum and characteristic reaction times for the photochemical mechanism of McRae *et al.* [89].

of O atom production and decay rates under typical conditions indicates that the predominant removal step (by four orders of magnitude) is reaction with molecular oxygen

$$O + O_2 + M \rightarrow O_3 + M. \tag{71}$$

Since the concentration of both molecular oxygen (O_2) and the third body (M), are fixed, the kinetics of O are described to a very good approximation by (69) with a_i and b_i constant. Under these conditions the eigenvalues and characteristic reaction times can be expected to be similar. This behavior was also observed for most of the free radicals: RO, OH, RO_2, NO_3, RCO_3, and HO_2. When there is coupling between species, and the rate terms are of comparable magnitude, the a_i's and b_i's are no longer constant and the analytic solution (70) is inappropriate.

9.1. *Selection of a Suitable Solution Scheme*

In the last few years considerable effort has been devoted to developing general purpose algorithms for solving stiff ordinary differential equations [74–79].

In applications involving simultaneous transport and chemistry such as that of interest here, the reaction rate equations must be integrated at a large number of grid points for relatively short periods of time between transport steps. As a consequence, self starting methods with low overheads are highly desirable. As mentioned earlier, the large size of the computational grid usually precludes storing more than the

results of the previous time step. From a pragmatic point of view it is important to recognize that errors associated with the transport steps are rarely smaller than a few percent so in general there is little to be gained by requiring highly accurate solutions of the kinetics. Summarizing, the desirable requirements of a solution scheme for the chemical kinetics are low start up costs, minimal computer memory requirements, and extreme computational speed.

Given the above considerations, two different solution schemes were sought; one capable of providing highly accurate benchmark standards of predictions and the other, an extremely fast algorithm for use in the airshed model. Since the factors influencing the choice of the method use in the model are discussed in Section 9.3 they will not be discussed here. The method chosen to establish the standard of accuracy for judging other methods was the implementation of the Gear technique by Hindmarsh and Byrne [80] and Byrne *et al.* [81]. Their program, called EPISODE, is extremely well documented and has been subjected to extensive testing by a number of different investigators [79, 81]. Unlike the original Gear method, the program employs a true variable step, variable order approximation that is ideally suited to problems with time varying parameters. Another reason for choosing this particular code was the ease with which different treatments of the Jacobian could be tested. In the version of EPISODE used in this study the Jacobian could be evaluated in either of four ways: functional iteration, analytic evaluation, finite differences, or diagonal approximations. The ability to exercise easily these options considerably simplified the task of identifying the most efficient means for solving the chemical kinetics.

9.2. *Pseudo Steady State Approximation*

Even with fast integration schemes the computational cost of solving the atmospheric diffusion equation is extremely high. There is a need to reduce both the number of active chemical species, to minimize storage requirements, and the stiffness, to lower the computational cost. One approach, commonly used in chemical kinetics, is to alleviate some of these difficulties by employing the pseudo steady state approximation [82, 83]. The basic idea behind this approximation is that the transients associated with the stiff variables decay very rapidly to their equilibrium values. If the concentrations are partitioned into two components, one associated with the nonstiff components $\mathbf{c}_d$ and the other comprising the stiff species, $\mathbf{c}_s$, then if the pseudo steady state approximation is used, (68) is replaced by the systems

$$\dot{\mathbf{c}}_d = \mathbf{f}_d(\mathbf{c}_d, \mathbf{c}_s) \tag{72}$$

and

$$\mathbf{0} = \mathbf{f}_s(\mathbf{c}_d, \mathbf{c}_s). \tag{73}$$

The two main difficulties associated with the valid use of pseudo steady state approximations are the identification of those species that can be treated in this way and the determination of the time after which the approximation is valid. For simple systems there is an extensive literature that utilizes singular perturbation theory to establish

TABLE VI

Comparison between the Exact Calculation and the Pseudo Steady State Approximation for Different Chemical Species

Time (min)	% Error								
	OH	O	RCO_3	RO_2	RO	HO_2	HNO_4	NO_3	N_2O_5
60	-2.5×10^{-3}	-4.3×10^{-6}	$+8.3 \times 10^{-3}$	-3.4×10^{-3}	-1.2×10^{-2}	$+8.6 \times 10^{-3}$	$+8.5 \times 10^{-2}$	-3.5×10^{-2}	-3.7×10^{-1}
120	-2.7×10^{-3}	$+2.1 \times 10^{-6}$	-2.0×10^{-3}	-4.2×10^{-3}	$+8.6 \times 10^{-3}$	$+4.1 \times 10^{-3}$	$+3.8 \times 10^{-2}$	$+2.6 \times 10^{-2}$	$+3.3 \times 10^{-1}$
180	$+2.6 \times 10^{-2}$	-5.6×10^{-6}	$+7.3 \times 10^{-3}$	-1.0×10^{-2}	$+1.5 \times 10^{-2}$	-1.1×10^{-2}	$+2.1 \times 10^{-2}$	-1.6×10^{-1}	$+1.6 \times 10^{-1}$
240	$+7.0 \times 10^{-2}$	$+4.7 \times 10^{-6}$	-2.8×10^{-3}	-1.5×10^{-2}	$+1.8 \times 10^{-1}$	-4.5×10^{-2}	$+3.0 \times 10^{-3}$	$+9.1 \times 10^{-2}$	$+1.4 \times 10^{-1}$
300	$+1.2 \times 10^{-1}$	$+3.2 \times 10^{-6}$	-2.0×10^{-2}	$+2.2 \times 10^{-3}$	$+4.3 \times 10^{-1}$	-7.3×10^{-2}	-3.0×10^{-2}	-2.1×10^{-1}	$+7.0 \times 10^{-2}$
360	-8.8×10^{-2}	-1.9×10^{-5}	$+1.9 \times 10^{-3}$	-1.8×10^{-2}	-5.6×10^{-1}	$+7.9 \times 10^{-2}$	$+3.7 \times 10^{-2}$	$+1.7 \times 10^{-2}$	$+2.9 \times 10^{-2}$

* Percentage Error = 100 [PSSA/EXACT − 1].

the appropriate bounds [78, 82–84]. Unfortunately, there is as yet no well-developed theory for systems as complex as the photochemical reaction mechanism utilized in this study. Thus, an approximate way to identify candidate species was developed.

The particular approach adopted in this study was to analyze the behavior of the kinetic equations by performing an eigenvalue–eigenvector analysis of the mechanism Jacobian under a wide variety of test conditions. The reason for doing this is that the eigenvalues all have negative real parts that can be ranked into two distinct subsets. The first set of largest negative eigenvalues generally have eigenvectors containing only one or two components. These elements as noted above usually correspond to those species that have very fast reaction times. These λ's typically have magnitudes as large as 10^7, corresponding to species half-lives as short as 10^{-6} seconds. The second set of eigenvalues has corresponding eigenvectors that each involve many, if not most, of the species in the reaction set. These represent the relatively slowly reacting species.

Using the eigenvalue analysis procedure, nine species were identified as candidates for the steady state approximation: O, RO, OH, RO_2, NO_3, RCO_3, HO_2, HNO_4, and N_2O_5. The solutions using the steady state approximation and one where all species were treated by differential equations were compared over a wide range of conditions. Typical examples of the results of these tests are shown in Tables VI and VII. Table VI is an assessment of the validity of each approximation. An inspection

TABLE VII

Comparison between Predictions of Complete System and Kinetics Using Pseudo Steady State Approximations

Time (min)	Species	Concentration (parts-per-million by volume) Complete system	Kinetics with 9 PSSA species	% Difference*
30	NO	0.0566	0.0567	0.18
	NO_2	0.4034	0.4070	0.89
	O_3	0.0830	0.0834	0.48
60	NO	0.0202	0.0202	0.00
	NO_2	0.3869	0.3889	0.51
	O_3	0.2189	0.2191	0.09
90	NO	0.0110	0.0110	0.00
	NO_2	0.3338	0.3329	−0.27
	O_3	0.3379	0.3383	0.12
120	NO	0.0066	0.0066	0.00
	NO_2	0.2628	0.2652	0.91
	O_3	0.4358	0.4391	0.75

* Percentage difference = 100 [PSSA/complete − 1].

of the results indicates that there are negligible differences between the species being treated by differential or algebraic equations. The most important comparison, however, is the influence of the use of the approximation on the predicted concentrations, $\mathbf{c}_d$. Even after 120 minutes the maximum error shown in Table VII is less than 0.5%. The conclusion reached from an analysis of these and other test cases was that the species identified from the eigenvalue analysis could be treated in steady state with minimal effects on the predicted concentrations of the primary species, $\mathbf{c}_d$.

Once the concentration vector has been partitioned into stiff and nonstiff components, there are a variety of algorithms that can take advantage of the problem structure. For example, Robertson [85] utilized the division in the iterations involved with the use of implicit multistep formulas. During any single step, by fixing the part of the iteration matrix corresponding to the nonstiff components and only updating the elements arising from the transients, significant computational economies were achieved. Techniques that achieve these efficiencies without prior knowledge about the problem structure are relatively rare. Enright and Kamel [86] have developed a general purpose computer code for systems where the stiffness is due to a few components of a large system.

One other approach for minimizing the influence of stiffness is to choose the initial conditions for $\mathbf{c}_s$ so that the complete system does not have the initial transient behavior. While it is extremely difficult to develop a general theory some initial steps in this direction have been made by Watkins [87] and Lambert [88]. The approach of Watkins [87] is particularly relevant because his algorithm has been developed to set initial conditions for transport problems. Unfortunately the cost of the proposed iteration scheme, when applied to systems of the size encountered in this study, is likely to be prohibitive. Kreiss [78] has addressed a similar situation in an attempt to set the initial conditions in a way that would eliminate the rapidly oscillating terms associated with large, purely imaginary eigenvalues. At this time there is no satisfactory means for a priori specification of the initial values for $\mathbf{c}_s$ that will remove or reduce the stiffness of systems of the type considered here.

9.3. *Asymptotic Integration Scheme*

In the previous section the size and stiffness of the reaction mechanism was reduced by employing the pseudo steady state approximation. Even with these changes it was still not feasible to economically use the EPISODE program in the solution of the full atmospheric diffusion equation. A variety of other alternatives were investigated in an attempt to significantly lower the computational cost but without substantially compromising the solution accuracy. The trapezoidal rule was rejected because of the overheads associated with the matrix decompositions. Even with the use of sparse matrix packages and infrequent Jacobian updating, the cost of Newton-type schemes was still excessive. The particular approach finally decided upon with the asymptotic integration method of Young and Boris [89, 90]. Designed to solve the reaction kinetics embedded in very large hydrodynamic problems, the method is self starting, extremely fast and requires minimal storage; as such, it satisfies most of the selection criteria discussed above.

A particularly attractive feature of the method is that it has a very low start up overhead because all that is required to begin a new integration step are the current values of the variables and the derivatives. A second-order predictor–corrector scheme that takes special notice of those equations determined at the beginning of the step to be stiff is employed to continue the integration process. When applied to stiff equations, the method is suited to situations where the solution is slowly changing or nearly asymptotic yet the time constants are prohibitively small. This occurs when the formation and loss rates are large, nearly equal, and there is strong coupling among the equations. Thus, the stiff equations are treated with a very stable method that damps out the small oscillations caused by the very small time constants.

The predictor–corrector algorithm provides enough information to choose the subsequent timestep size once convergence has been achieved. For efficiency, an initial timestep is chosen that approximates the timestep that will be determined after convergence of the predictor–corrector scheme. This initial trial timestep is chosen independently of the stiffness criterion and is determined such that none of the variables will change by more than a prescribed amount. If the formation rate is much larger than the loss rate, it is reasonable to assume that a_i and b_i will remain relatively constant for large changes in c_i. Often the initial change in c_i may be large enough to equilibrate the formation and loss rates. Thus the initial trial timestep $\Delta\tau$, is chosen in two ways:

$$\Delta\tau = \varepsilon \min_i \left[\frac{c_i}{f_i}\right] \tag{74}$$

or if $a_i \gg b_i c_i$ then

$$\Delta\tau = \varepsilon \min_i \left[\frac{1}{b_i}\right]. \tag{75}$$

The second criterion is needed when the initial conditions, for some species, are unknown or set to zero. Here ε is a scale factor, the selection of which is discussed shortly. The timestep dictated by (74) may be larger than some or all of the equilibrium times, in which case the corresponding equations would be classified as stiff. Nevertheless, when solved by the asymptotic method, this timestep ensures that accuracy can be maintained. When a stiff equation is close to equilibrium, the changes in the functional values over the timestep will be small even though the adjustment rate toward equilibrium can be very much shorter than the timestep. When the stiff equation is far from a dynamic equilibrium, the timestep should be scaled down proportionally to the equilibrium time to ensure that the transition to equilibrium will be followed accurately. This readjustment, because of the very fast rate, generally takes place rapidly after which much longer timesteps may be taken.

After a timestep has been chosen, all of the equations are separated into two classes, stiff and nonstiff, according to the values of the b_i. The two types of equations are then integrated by separate predictor–corrector schemes. A simple asymptotic formula is used for those equations determined to be stiff.

The predictor part of the step is performed as follows:

$$\text{Nonstiff:} \qquad c_i(1) = c_i(0) + \Delta\tau f_i(0), \tag{76}$$

$$\text{Stiff:} \qquad c_i(1) = c_i(0) + \frac{\Delta\tau f_i(0)}{1 + \Delta\tau f_i(0)}, \tag{77}$$

where $f_i(0) = f_i[t(0), c_i(0)]$ and $c_i(k)$ is the kth iterated value of c_i, or an approximation to $c_i[t(0) + \Delta\tau]$. The zeroth iteration, $c_i(0)$, is the initial value at $t(0)$ and $c_i(1)$ is the result of the predictor step. Also note that $f_i(k) = f_i[t(0) + \Delta\tau, c_i(k)]$. The corrector formulas are:

$$\text{Nonstiff:} \qquad c_i(k+1) = c_i(0) + \frac{\Delta\tau}{2}[f_i(0) + f_i(k)], \tag{78}$$

$$\text{Stiff:} \qquad c_i(k+1) = c_i(0) + \frac{2\Delta\tau[a_i(k) - b_i(0)\, c_i(0) + f_i(0)]}{4 + \Delta\tau[b_i(k) + b_i(0)]}. \tag{79}$$

By comparing $c_i(k+1)$ with $c_i(k)$ on successive iterations using the relative error criterion ε to satisfy

$$\max_i \left[\frac{|c_i(k+1) - c_i(k)|}{c_i(k+1)}\right] \leqslant \varepsilon \tag{80}$$

the convergence of each of the individual equations can be determined. As applied in the present application, ε is typically $O(10^{-3})$ and if the formation and loss rates are nearly equal ε is scaled down slightly, to allow quicker convergence for equations that are nearly in equilibrium.

In practice, c_i is constrained by a minimum value when c_i is decaying exponentially toward zero. This lower bound must be selected to insure that its value in no way affects the physics but yet decouples the equation from accurate integration. Decoupling is accomplished by avoiding applying (80) to all equations that have decayed to values corresponding to their lower bounds. Convergence for these equations is then trivial and the function no longer affects the size of the timestep. For equations that are decaying exponentially to zero, with time constants that are small enough to control the timestep, it is important for efficiency reasons to decouple these equations at the largest lower bound possible.

In practical application the maximum solution speed is realized by keeping the allowed number of corrector iterations small, typically one or two. If satisfactory convergence of all equations has not been obtained before or during the last iteration, the step is started over with a smaller timestep. By keeping the maximum number of iterations small, a minimum amount of time is wasted on an unstable or nonconvergent step. When nonconvergence is encountered, it is more efficient to reduce the timestep sharply (a factor of 2 or 3). On the other hand, when increasing the timestep, as, for example, when convergence is achieved on the first or second

iteration, it is best to increase only by 5–10% each step. The asymptotic integration scheme was compared against the program EPISODE [80, 81] to evaluate the characteristics of the algorithm when applied to the photochemical reaction mechanism. For all EPISODE calculations semi-relative error control was used with a convergence tolerance of 0.0001. The starting and maximum step sizes were set to 10^{-5} and 10 minutes, respectively.

Both programs were exercised over a wide range of initial conditions, pseudo steady state approximations, photolysis rates and diurnal cycles. Two features were apparent in all the tests, and they are illustrated in Table VIII. First, and perhaps most important, is that there were negligible differences in the predictions of both schemes over solution steps comparable to the maximum expected transport times. For example, after 30 minutes the maximum discrepancy between the two schemes for the species NO, NO_2, and O_3 was $O(0.2\%)$.

The most striking difference between the two schemes is the high start up costs associated with the EPISODE algorithm. During the initial 30 minutes there is a factor of 7 difference in the computation time. Once started, however, the incremental cost, per time interval, of using EPISODE becomes successively smaller. From a

TABLE VIII

Comparison of Start Up Times for EPISODE and Hybrid Solution Scheme for Typical Smog Chamber Experiment

Time (min)	Species	Concentration (parts-per-million by volume) Episode	Hybrid solver	Computer time (ms) per 30 minute step Episode	Hybrid
30	NO	0.0567	0.0567 (0.00)*	1014	152
	NO_2	0.4070	0.4077 (0.17)		
	O_3	0.0834	0.0832 (−0.24)		
60	NO	0.0202	0.0203 (0.50)	175	104
	NO_2	0.3889	0.3914 (0.64)		
	O_3	0.2191	0.2194 (0.14)		
90	NO	0.0110	0.0107 (−2.73)	79	81
	NO_2	0.3329	0.3290 (−1.17)		
	O_3	0.3383	0.3450 (1.98)		
120	NO	0.0066	0.0062 (−6.06)	47	70
	NO_2	0.2652	0.2557 (−3.58)		
	O_3	0.4391	0.4497 (2.41)		
				1315 ms	407 ms

* Percentage difference between EPISODE and Hybrid solution technique = 100 [Hybrid/EPISODE − 1].

practical point of view, considering the short integration intervals in an operator splitting solution, the asymptotic scheme is clearly preferable to the EPISODE algorithm for the present application.

9.4. *Implementation of Asymptotic Integration Scheme*

Using the operator splitting procedures described earlier, (9) can be written in the form

$$\text{Transport} \qquad \frac{\partial c_i}{\partial t} = L(\mathbf{x}, t)\, c_i, \tag{81}$$

$$\text{Chemistry} \qquad \frac{\partial c_i}{\partial t} = f_i(c_1, \ldots, c_p, t). \tag{82}$$

If $\mathbf{T}_x$, $\mathbf{T}_y$, $\mathbf{T}_z$ and $\mathbf{C}_c$ are the numerical approximations to the transport and chemistry operators then a complete solution can be obtained from the sequence

$$\mathbf{c}_i^{n+1} = \mathbf{T}_x \mathbf{T}_y \mathbf{T}_z \mathbf{C}_c(2\Delta t)\, \mathbf{T}_z \mathbf{T}_y \mathbf{T}_x \mathbf{c}_i^{n-1}, \tag{83}$$

where $\mathbf{C}_c$ symbolically denotes the means of solving (82) at each of the grid points given a set of initial conditions. Most of the computer time required for each cycle (83) is consumed by the chemical solution $\mathbf{C}_c$. Two advantages of operator splitting are apparent, the chemistry is decoupled from the transport and it can be solved for a period $2\Delta t$. This latter feature is particularly important because most of the overhead associated with solving (82) occurs at the start of each initial value problem; subsequent time increments can be obtained at minimal expense.

The actual sequence of operations used to obtain a solution of (83) is as follows. Solve

$$\frac{\partial c_i^*}{\partial t} = L_x c_i^*, \tag{84}$$

$$\frac{\partial c_i^{**}}{\partial t} = L_y c_i^{**}, \tag{85}$$

$$\frac{\partial c_i^{***}}{\partial t} = L_z c_i^{***}, \tag{86}$$

on the interval $t^{n-1} \leqslant t \leqslant t^n$,

$$\frac{\partial c_i}{\partial t} = f_i(c_1, \ldots, c_p, t) \tag{87}$$

on the interval $t^{n-1} \leqslant t \leqslant t^{n+1}$, and then solve the system (84)–(86) in the reverse order, i.e., in z, y, and x directions. The initial conditions for each of the problems (84)–(86) are: $c_i^*(t^{n-1}) = c_i(t^{n-1})$, $c_i^{**}(t^{n-1}) = c_i^*(t^n)$, $c_i^{***}(t^{n-1}) = c_i^{**}(t^n)$ and for (87) $c_i(t^{n-1}) = c_i^{***}(t^n)$.

Unfortunately, there is little guidance in the literature relevant to establishing a priori bounds on the maximum value of Δt. Within the airshed model it has been observed that the convergence of the sequence (83), during the photochemically active daylight hours, is controlled more by the rate of vertical turbulent mixing than by the Courant limit of the horizontal advection schemes. As a result of considerable experimentation with successively smaller time steps it was found that if $2\Delta t$ was limited to be less than 10 minutes, the predicted results were comparable to cases in which the two-dimensional coupled problem (9) was solved directly. At night when there is little or no chemical activity, the chemical time steps are controlled by the stability limits of the advection schemes. The total computer time required to simulate the concentration dynamics of 15 species at 3000 grid points for a 24-hour period is $O(50$ minutes) on an IBM 370/168. The interested reader is referred to McRae *et al.* [91] for a description of the air pollution model.

10. Conclusions

In this paper, a variety of numerical methods were studied in order to identify a solution scheme for the atmospheric diffusion equation. As a result of this investigation, a composite technique was developed in which operator-splitting was first used to segment the three-dimensional system of equations into a sequence of one-dimensional problems. Each transport step was further simplified to three basic components: an advection step, application of a nonlinear filter and finally a diffusion step. A Galerkin, linear finite element scheme was adopted for the critical advection step. The results of numerous numerical experiments indicate that this algorithm, together with the filter step, preserves extreme values, gradients, total mass and mean square concentration. The solution of the chemical kinetics component is carried out by a second-order predictor, iterated corrector technique, in combination with an asymptotic treatment of the stiff components of the problem [90, 91]. Computational economies are achieved by implementation of the pseudo steady state approximation.

APPENDIX: Notation

$a(\mathbf{X}, t), b(\mathbf{X}, t)$	Coefficients associated with boundary conditions (3)
a_i	Production rate for species i, $i = 1, 2,..., p$
$\mathbf{A}$	An $r \times r$ matrix representing the discrete approximation to L at r computational grid points. ($\mathbf{A}_j$ is the discrete representation of L_j)
b_i	First-order coefficient for removal rate of species i, $i = 1, 2,..., p$
B	Linear boundary operator
$\mathbf{B}, \mathbf{H}, \mathbf{M}, \mathbf{P}, \mathbf{Q}, \mathbf{S}$	Matrices of dimension $r \times r$ associated with different spatial discretization techniques
$\mathbf{c}_d$	Concentration vector of nonstiff components
$c_i(k)$	kth iterate of c_i

$c_i(\mathbf{X}, t), c_i(\mathbf{x}, t)$	Concentrations of species i in the physical and computational domains; $i = 1, 2,..., p$
$\mathbf{c}_i(\mathbf{x}, t)$	Concentration vector of species i at r computational points $\mathbf{c}_i(\mathbf{x}, t) = (c_i(\mathbf{x}_j, t); j = 1, 2,..., r)$
$\mathbf{c}_s$	Concentration vector of stiff components
C	Courant number
$\mathbf{C}_c$	Symbol representing solution of the chemical kinetics
f_i	Chemical formation (or depletion) rate of species i $\mathbf{f}_i = (f_i(c_1(\mathbf{x}_j, t),..., c_p(\mathbf{x}_j, t)); j = 1, 2,..., r)$
$\mathbf{F}$	Mapping function that transforms points from $\mathbf{X}$ into $\mathbf{x}$
$g_i(\mathbf{X}, t)$	Species specific boundary condition coefficient
$h(X, Y)$	Topographic surface (lower boundary of region)
$H(X, Y, t)$	Time varying upper surface of region
$\mathbf{I}$	Unit matrix of dimension $r \times r$
$\mathbf{J}$	Jacobian matrix with elements $\partial f_i/\partial c_j$, $i,j = 1, 2,..., p$
k_l	Rate constant for reaction l
K_f	Coefficient in noise filter
K_n	Diffusion coefficient associated with noise filter
$\mathbf{K}$	Second-order turbulent eddy diffusion tensor (usually a diagonal matrix with elements K_{xx}, K_{yy}, K_{zz}). In the computational domain $\mathbf{K}_{xx}$ are the values of K_{xx} at each of the r grid points.
L	Three-dimensional, semi-linear, elliptic differential operator (L_x, L_y, L_z are the components in x, y and z directions).
R	Stiffness ratio
r	Radial coordinate for Crowley problem
t	Time
T	Extent of time interval for solution
$\mathbf{T}$	Composite transport operator (T_j is the transport operator for the jth direction)
$\mathbf{u}(\mathbf{X}, t)$	Velocity field in physical domain $\mathbf{u} = (u, v, w)$
$\mathbf{U}$	Velocity field in computational domain $\mathbf{U} = (u_j; j = 1, 2,..., r)$
$\mathbf{V}(\mathbf{x}, t)$	Velocity field in transformed domain $\mathbf{V} = (u, v, W)$
$\mathbf{x}$	Point in computational domain $\mathbf{x} = (x, y, z) \in \Omega_c$
$\mathbf{X}$	Point in physical domain $\mathbf{X} = (X, Y, Z) \in \Omega_t$

Greek Symbols

α, β	Time varying coefficients associated with the concentration and velocity distributions employed in the Galerkin formulation
δ	Discretization unit (either finite element or grid size)
ε	Relative error criterion
Δt	Basic time step of atmospheric diffusion equation
$\Delta\tau$	Time step for solution of the chemical kinetics
Δx	Size of computational grid element
ΔH	$= H(X, Y, t) - h(X, Y)$

θ Volume to width ratio for test wave forms or angular coordinate
λ An arbitrary parameter with $\lambda \geqslant 0$
λ_i Eigenvalue of Jacobian matrix $\mathbf{J}$, $i = 1, 2, ..., p$
σ Normal direction to $\partial\Omega$
$\mathbf{v}$ Material flux $= \mathbf{K}_{xx}(\partial \mathbf{c}/\partial x) - \mathbf{U}\mathbf{c}$
ϕ_j Basis functions for Galerkin formulation
ψ_j Filter function variable (0, 1)
ω Fourier frequency for test wave forms and angular velocity for Crowley problem
Ω_c Time invariant computational domain
Ω_t Time varying physical domain (Ω_0 initial extent)
$\partial\Omega$ Domain boundary

Sub- and Superscripts

a Advective transport step
c Indicates computational domain
d Diffusive transport step or nonstiff component of concentration vector
e Grid point subscript for testing sign changes during filter application
i Species index
j Index to denote coordinate direction ($x = 1$, $y = 2$, $z = 3$) or computational grid point ($j = 1, 2, ..., r$)
k Iteration counter during one time step
l Domain of final filter application (number of grid points)
m Half width of enveloping interval for testing slope change in filtering scheme
n Time level
o Initial conditions
p Number of chemical species
q Spatial integration index for Galerkin formulation
r Number of computational grid points
s Spatial integration index for Galerkin formulation or stiff component of concentration vector

ACKNOWLEDGMENTS

This work was supported by the State of California Air Resources Board under Contracts A5-046-87 and A7-187-30. Additional funding was provided by a Department of Energy Institutional Grant EY-76-G-03-1305. The assistance of Dr. T. R. Young of the Naval Research Laboratory, and Dr. J. W. Tilden of the Environmental Quality Laboratory is appreciated.

REFERENCES

1. J. H. SEINFELD, "Air Pollution: Physical and Chemical Fundamentals," McGraw–Hill, New York, 1975.
2. W. R. GOODIN, G. J. MCRAE, AND J. H. SEINFELD, *J. Appl. Meteorol.* 18 (1979a), 761.

3. W. R. GOODIN, G. J. MCRAE, AND J. H. SEINFELD, *J. Appl. Meteorol.* **19** (1979b), 98.
4. W. R. GOODIN, G. J. MCRAE, AND J. H. SEINFELD, *J. Appl. Meteorol.* **20** (1981), 92.
5. S. D. REYNOLDS, P. M. ROTH, AND J. H. SEINFELD, *Atmospheric Environment* **7** (1973), 1033.
6. A. H. FALLS AND J. H. SEINFELD, *Environ. Sci. Technol.* **12** (1978), 1398.
7. T. GAL-CHEN AND R. C. SOMERVILLE, *J. Comput. Phys.* **17** (1975), 209.
8. T. L. CLARK, *J. Comput. Phys.* **24** (1977), 186.
9. P. J. ROACHE, "Computational Fluid Dynamics," Second ed., Hermosa Publications, Albuquerque, 1976.
10. J. L. ANDERSON, S. PREISER, AND E. L. RUBIN, *J. Comput. Phys.* **2** (1968), 279.
11. W. L. OBERKAMPF, *Int. J. Numer. Methods Engrg.* **10** (1976), 211.
12. M. VINOKUR, *J. Comput. Phys.* **14** (1974), 105.
13. A. LAPIDUS, *J. Comput. Phys.* **2** (1967), 154.
14. S. B. MARGOLIS, *J. Comput. Phys.* **27** (1978), 410.
15. R. C. Y. CHIN AND R. L. BRAUN, *J. Comput. Phys.* **34** (1980), 74.
16. B. ENGQUIST, B. GUSTAFSSON, AND J. VREEBURG, *J. Comput. Phys.* **27** (1978), 295.
17. J. DOUGLAS, T. DUPONT, AND R. EWING, *SIAM J. Numer. Anal.* **16** (1979), 503.
18. E. J. KANSA, *J. Comput. Phys.* **42** (1981), 152.
19. J. DESCLOUX, *SIAM J. Numer. Anal.* **9** (1972), 260.
20. A. ISERLES, *SIAM J. Numer. Anal.* **18** (1981), 1.
21. A. W. RIZZI AND H. E. BAILEY, *AIAA J.* **14** (1976), 621.
22. A. W. RIZZI AND M. INOUYE, *AIAA J.* **11** (1973), 1478.
23. R. J. KEE AND J. A. MILLER, *AIAA J.* **16** (1978), 169.
24. P. D. THOMAS AND K. H. WILSON, *AIAA J.* **14** (1976), 629.
25. J. DOUGLAS AND J. E. GUNN, *Numer. Math.* **6** (1964), 428.
26. J. E. DENDY, *SIAM J. Numer. Anal.* **14** (1977), 313.
27. W. R. BRILEY AND H. MCDONALD, *J. Comput. Phys.* **34** (1980), 54.
28. T. J. WEARE, *Int. J. Numer. Methods Engrg.* **14** (1979), 921.
29. N. N. YANENKO, "The Method of Fractional Steps," Springer-Verlag, New York, 1971.
30. G. I. MARCHUK, *in* "Numerical Solution of Partial Differential Equations, II," Academic Press, New York, 1971.
31. G. I. MARCHUK, "Methods of Numerical Mathematics," Springer-Verlag, New York, 1975.
32. N. N. YANENKO, V. M. KOVENYA, V. D. LISEJKIN, V. M. FOMIN, AND E. V. VOROZHTSOV, *Comput. Methods Appl. Mech. Engrg.* **17** (1979), 659.
33. J. L. MORRIS, *J. Comput. Phys.* **5** (1970), 219.
34. A. R. GOURLAY AND A. R. MITCHELL, *SIAM J. Numer. Anal.* **6** (1969), 37.
35. A. R. GOURLAY, *Proc. Roy. Soc. London Ser. A* **323** (1971), 219.
36. D. GOTTLIEB, *SIAM J. Numer. Anal.* **9** (1972), 650.
37. M. CRANDALL AND A. MAJDA, *Numer. Math.* **34** (1980), 285.
38. E. S. ORAN AND J. P. BORIS, *Prog. Energy Combustion Sci.* **7** (1981), 1.
39. D. ANDERSON AND B. FATTAHI, *J. Atmospheric Sci.* **31** (1974), 1500.
40. C. Y. LIU, W. R. GOODIN, AND C. M. LAM, *Comput. Methods Appl. Mech. Engrg.* **9** (1976), 281.
41. G. R. CARMICHAEL, T. KITADA, AND L. K. PETERS, *Computers and Fluids* **8** (1980), 155.
42. J. P. BORIS AND D. L. BOOK, *J. Comput. Phys.* **11** (1973), 38.
43. J. P. BORIS AND D. L. BOOK, *J. Comput. Phys.* **20** (1976), 397.
44. D. L. BOOK, J. P. BORIS, AND K. HAIN, *J. Comput. Phys.* **18** (1975), 248.
45. Y. ADAM, *Comput. Math. Appl.* **1** (1975), 393.
46. Y. ADAM, *J. Comput. Phys.* **24** (1977), 24.
47. R. S. HIRSH, *J. Comput. Phys.* **19** (1975), 90.
48. F. THIELE, *J. Comput. Phys.* **27** (1978), 1.
49. M. CIMENT AND S. H. LEVENTHAL, *Math. Comp.* **32** (1978), 143.
50. M. CIMENT, S. H. LEVENTHAL, AND B. C. WEINBERG, *J. Comput. Phys.* **28** (1978), 135.
51. J. J. CONNOR AND C. A. BREBBIA, "Finite Element Techniques for Fluid Flow," Butterworth, London, 1976.

52. G. STRANG AND G. J. FIX, "An Analysis of the Finite Element Method," Prentice-Hall, Engelwood Cliffs, N.J.
53. K. W. MORTON AND A. K. PARROTT, *J. Comput. Phys.* **36** (1980), 249.
54. J. E. FROMM, *J. Comput. Phys.* **3** (1968), 176.
55. W. P. CROWLEY, *Monthly Weather Rev.* **96** (1968), 1.
56. H. S. PRICE, R. S. VARGA, AND J. E. WARREN, *J. Math. Phys.* **45** (1966), 301.
57. H. KREISS AND J. OLIGER, "Methods for the Approximation Solution of Time Dependent Problems," GARP Publication Series No. 10, World Meteorological Organization.
58. H. J. P. CULLEN, *Quart. J. Roy. Meteorol. Soc.* **102** (1976), 77.
59. H. STORCH, *Beiträge Phys. Atmosphare* **51** (1978), 189.
60. W. H. RAYMOND AND A. GARDNER, *Monthly Weather Rev.* **104** (1976), 1583.
61. L. B. WAHLBIN, *in* "Mathematical Aspects of Finite Elements in Partial Differential Equations," Academic Press, New York, 1974.
62. J. D. MAHLMAN AND R. W. SINCLAIR, "Advances in Environmental Science and Technology," Vol. 8, Wiley, New York, 1977.
63. B. VAN LEER, *J. Comput. Phys.* **14** (1974), 361.
64. C. K. FORESTER, *J. Comput. Phys.* **23** (1977), 1.
65. C. R. MOLENKAMP, *J. Appl. Meteorol.* **7** (1968), 160.
66. T. NITTA, *J. Meteorol. Soc. Japan* **40** (1962), 13.
67. J. M. VARAH, *SIAM J. Numer. Anal.* **8** (1971), 569.
68. J. M. VARAH, *SIAM J. Numer. Anal.* **8** (1971), 598.
69. J. H. CHEN, *J. Comput. Phys.* **13** (1973), 522.
70. B. GUSTAFSSON, H. O. KREISS, AND A. SUNDSTROM, *Math. Comp.* **26** (1972), 649.
71. B. GUSTAFSSON, *J. Comput. Phys.* **34** (1980), 108.
72. D. GOTTLIEB AND E. TURKEL, *J. Comput. Phys.* **26** (1978), 181.
73. D. M. SLOAN, *Int. J. Numer. Methods Engrg.* **15** (1980), 1113.
74. R. A. WILLOUGHBY, "Stiff Differential Systems," Plenum, New York, 1974.
75. J. D. LAMBERT, *in* "Computational Techniques for Ordinary Differential Equations," Academic Press, London, 1976.
76. D. D. WARNER, *J. Phys. Chem.* **81** (1977), 2329.
77. A. R. CURTIS, *in* "Numerical Software—Needs and Availability," Academic Press, London, 1978.
78. H. O. KREISS, *SIAM J. Numer. Anal.* **16** (1979), 980.
79. L. F. SHAMPINE AND C. W. GEAR, *SIAM Rev.* **21** (1979), 1.
80. A. C. HINDMARSH AND G. D. BYRNE, "EPISODE: An Experimental Program for the Integration of Systems of Ordinary Differential Equation Systems," Lawrence Livermore Laboratory Report UCID-30112, 1975.
81. G. D. BYRNE, A. C. HINDMARSH, K. R. JACKSON, AND H. G. BROWN, *Comput. Chem. Engrng.* **1** (1977), 133.
82. R. C. AIKEN AND L. LAPIDUS, *AIChE J.* **21** (1975a), 817.
83. R. C. AIKEN AND L. LAPIDUS, *AIChE J.* **21** (1975b), 1227.
84. P. V. KOKOTOVIC, J. J. ALLEMONG, J. R. WINKELMAN, AND J. H. CHOW, *Automatica* **16** (1980), 23.
85. H. H. ROBERTSON, *J. Inst. Math. Appl.* **18** (1976), 249.
86. W. H. ENRIGHT AND M. S. KAMEL, *ACM Trans. Math. Software* **5** (1979), 374.
87. D. S. WATKINS, *SIAM J. Numer. Anal.* **18** (1981), 13.
88. J. D. LAMBERT, *SIAM J. Numer. Anal.* **18** (1981), 83.
89. T. R. YOUNG, "CHEMEQ—A Subroutine for Solving Stiff Ordinary Differential Equations," NRL Memorandum Report 4091, Naval Research Laboratory, Washington, D.C. (1980).
90. T. R. YOUNG AND J. P. BORIS, *J. Phys. Chem.* **81** (1977), 2424.
91. G. J. MCRAE, W. R. GOODIN, AND J. H. SEINFELD, *Atmospheric Environment*, in press.