JOURNAL OF COMPUTATIONAL PHYSICS 17, 358–383 (1975)

Numerical Representations Which Model Properties of the Solution to the Diffusion Equation

BRIAN MARTIN

School of Physics, Sydney University, Sydney, New South Wales, Australia 2006

Received June 20, 1974; revised December 7, 1974

A particular way of basing numerical representations on the integral form of the diffusion equation is presented. By imposing certain restrictions on these representations, it is shown how to guarantee that the numerical solution will conserve mass, produce no negative masses, be stable, and have the same low-order moments as the analytical solution to the diffusion equation. Selected representations obtained using the method are found to be highly successful on sample tests, according to the various criteria used.

1. INTRODUCTION

Numerical representations of the diffusion equation have an important place in the solution of many problems in different branches of science. Usually such representations are based on the differential form of the diffusion equation; one works with concentrations at grid points. Here we develop numerical representations based on the integral form of the diffusion equation, in which one works with masses in boxes. In most numerical representations, whether based on the differential or integral form of the diffusion equation, finite difference approximations are used for at least some of the time and space derivatives involved. Here finite differences are replaced by sets of restrictions; the numerical solution generated by a representation satisfying particular sets of restrictions will be guaranteed to exactly model certain properties of the solution to the diffusion equation.

In Section 2 we introduce the particular form of the numerical representations based on the integral form of the diffusion equation which will be used as the basis for further development in the paper. In Section 3 some criterion for measuring the success of numerical representations are presented and discussed: ease of generation and computation, goodness of fit, stability, nonnegativity of the masses, conservation of mass and correct center of mass motion. In Sections 4 and 5 it is shown how to formulate restrictions on numerical representations to directly or indirectly make the numerical solution satisfy these criteria. In Section 4 restric-

tions for ensuring conservation of mass and nonnegativity of the masses are considered; under certain circumstances these restrictions are sufficient to guarantee stability. In Section 5 a method is presented for formulating restrictions on the numerical representation sufficient to make the numerical solution possess low-order moments equal to corresponding moments of the analytical solution. This method, called "moment-fitting," in most cases leads to improved goodness of fit, and as a by-product guarantees conservation of mass and correct center of mass motion. In the final section a number of selected representations are compared, using the various criteria, on a series of test problems. The representations guaranteed to model the properties of the solution to the diffusion equation considered here are found to be highly successful in nearly every respect.

2. Numerical Representations Based on the Integral Form of the Diffusion Equation

The differential form of the diffusion equation may be written

$$\partial c(\mathbf{x}, t)/\partial t = -\nabla \cdot \mathbf{J}(c, \mathbf{x}, t) + \mathscr{S}(c, \mathbf{x}, t), \tag{1}$$

where $c(\mathbf{x}, t)$ is the mass density or concentration, $\mathbf{J}(c, \mathbf{x}, t)$ is the mass current density, and $\mathscr{S}(c, \mathbf{x}, t)$ the rate of change of concentration due to sources and sinks. $\mathbf{J}$ and $\mathscr{S}$ depend in general on the position $\mathbf{x}$, time t, and the concentration $c(\mathbf{x}, t)$ and its derivatives. In this paper we restrict our attention to only those forms of (1) which are linear in c. A common and standard form for the mass current density $\mathbf{J}$, linear in c, is the phenomenological relationship

$$\mathbf{J}(c, \mathbf{x}, t) = -\mathsf{D}(\mathbf{x}, t) \cdot (\partial c/\partial \mathbf{x}) + \mathbf{V}(\mathbf{x}, t)\, c. \tag{2}$$

Here $\mathsf{D}(\mathbf{x}, t)$ is a tensor of diffusion coefficients and $\mathbf{V}(\mathbf{x}, t)$ is the velocity of the medium. Henceforth D and $\mathbf{V}$ and their analogs in other equations will be referred to as the diffusion parameters.

To solve the diffusion equation numerically, it is usual to select a set of points distributed throughout the volume of the medium, and represent the time and space derivatives in (1) in finite difference form. Numerical representations of this type may be described as being based on the differential form of the diffusion equation. Often there are difficulties with such representations, ranging from nonconservation of total mass to spurious diffusion.

An alternative is to base the numerical representation on the integral form of the diffusion equation. In integral form (1) may be written

$$\frac{\partial \mathscr{M}(t)}{\partial t} = -\int_A \mathbf{J} \cdot \mathbf{n}\, da + \int_V \mathscr{S}\, d\mathbf{x}, \tag{3}$$

where

$$\mathcal{M}(t) = \int_V c(\mathbf{x}, t)\, d\mathbf{x}. \tag{4}$$

$\mathcal{M}(t)$ is the mass at time t in a volume V with surface A, da is an infinitesimal element of area of this surface, and $\mathbf{n}$ is a unit vector normal to the element of area da. $\mathbf{J}$ does not usually depend on $\mathcal{M}(t)$ or masses in other volumes in the explicit manner that it depends on $c(\mathbf{x}, t)$. Therefore the analytical solution to the diffusion equation is normally found using (1); the mass $\mathcal{M}(t)$ may then be found using (4). But in constrast to the analytical situation, numerically it is convenient to use a representation based on the integral form (3).

We now describe the general features of a class of numerical representations based on this integral form. The medium is partitioned into N boxes, and time divided into intervals of finite duration. Each box contains at a given time a certain mass; these masses change with time according to the algorithm

$$\mathbf{m}(t_{s+1}) = \mathbf{m}(t_s)\, \mathsf{P}(s) + \mathbf{S}(s). \tag{5}$$

$\mathbf{m}(t)$ is an N-dimensional row vector whose ith component is the mass in box i at time t. $\mathsf{P}(s)$ is an $N \times N$ matrix of transition probabilities, and $\mathbf{S}(s)$ an N-vector of sources. The positive integer s indicates the number of applications of the algorithm (5) to the vector of masses $\mathbf{m}$, calculated from the initial time t_0. Thus t_s indicates the time t after s applications or iterations of the algorithm (5). The duration $t_{s+1} - t_s$ of time interval $s + 1$ is denoted by τ_s.

Conceptually the essential difference between numerical representations based on the differential and integral forms of the diffusion equation is that in the former one thinks about and works with concentrations at grid points, while in the latter one thinks about and works with masses in boxes. The direct advantage of basing a numerical representation on the integral form of the diffusion equation is that it is easy to ensure conservation of mass and nonnegativity of the masses, as we shall see in Section 4. However there is no guarantee that such representations will be successful in other respects. Each element of the matrix $\mathsf{P}(s)$ is calculated using functions of properties of the medium such as of the diffusion parameters $\mathsf{D}(\mathbf{x}, t)$ and $\mathbf{V}(\mathbf{x}, t)$. The values of the elements naturally depend on the numerical procedure by which they are calculated. If finite difference approximations are used to obtain $\mathsf{P}(s)$ then the representation will suffer many of the same shortcomings as those based on the differential form of the diffusion equation which use finite differences. The original method proposed in this paper, and detailed in Sections 4 and 5, involves determining the numerical representation by making it satisfy restrictions sufficient to make the numerical solution exactly model certain properties of the analytical solution. By avoiding finite differences, the representation can be formulated expressly to satisfy certain desirable properties, which we now consider.

(There are in addition to the methods described here a number of other ways of obtaining a numerical solution to the diffusion equation; these will be considered in Section 6.)

3. Criteria for Measuring the Success of a Numerical Representation

There are a number of properties which are desirable in a numerical representation of the diffusion equation. Some of the more important may be stated as follows.

Ease of generation and computation. The representation should be obtainable without undue difficulty, and its use should not entail excessive computation.

Goodness of fit or accuracy. The mass distribution generated by the numerical representation should be as similar as possible to the distribution which is the solution to the diffusion equation.

Stability. The numerical expressions for the masses should remain finite at all times and, if the diffusion parameters, sources and sinks are time-independent, converge after long periods of time to appropriate steady values which are independent of the initial mass distribution.

Nonnegativity of the masses. The numerical expressions for the masses should remain always and everywhere positive or zero.

Mass conservation. During any time interval, numerically the change in mass in a given volume should equal the mass flux across the borders of the volume, plus any change due to sources or sinks interior to the volume.

Correct center of mass motion. Numerically, the center of mass of a given amount of diffusing substance should move at the same average velocity as the ambient medium, at least for distances over which the variation in the velocity of the medium is small.

It is obvious that these properties are neither of equal importance nor mutually independent. The primary requirement of any numerical model is the adequate approximation of the solution to the equation being represented, within the limitation of reasonable computational effort. In terms of our properties, this means a reasonable combination of goodness of fit and ease of generation and computation. The four other properties stated above are characteristics of the solution to the diffusion equation which is being approximated. Therefore if the numerical solution is sufficiently accurate, then the numerical representation will be stable, will at least approximately conserve mass and approximately model the correct center of mass motion, and will not produce large negative masses.

But these latter four properties do not always follow as nicely as one would like from goodness of fit. For example a given numerical representation might give an expectation of a certain goodness of fit, but on occasions not be stable. In this circumstance we might wish to use a numerical representation guaranteed to be stable. Similarly we might wish to use a numerical representation for which masses are guaranteed never to become negative, and which both conserves mass and models the center of mass motion exactly.

For most of the properties, it is straightforward to measure the success of a numerical representation in satisfying them. Ease of generation and computation may be measured by the number of arithmetical operations required, or by execution time on a computer. One may without undue difficulty count the number of negative masses generated by the numerical representation at any time, and calculate the accuracy to which total mass is conserved and correct center of mass motion maintained. Representations can often be proved to be stable, and unstable behavior is usually readily apparent.

The appropriate procedure for measuring goodness of fit is not so obvious. In numerical representations obtained using finite difference approximations, it is usual to determine the size of the expected error consequent to each application of the numerical algorithm. This technique can provide limits on the maximum error in the numerical solution, but does not give a direct measure of its goodness of fit at a given time. Furthermore, very often this technique does not put more then wide bounds on the goodness of fit after numerous applications of the numerical algorithm.

With the method presented in this paper, numerical accuracy is obtained as a consequence of making the numerical representation satisfy certain restrictions which ensure that the numerical solution models certain properties of the analytical solution. To measure the goodness of fit obtained from representations generated with this method, we use a simple direct measure: the sum of the squares of the differences between the two distributions, $\mathbf{m}(t)$ and $c(\mathbf{x}, t)$. It is computationally convenient to take this sum over the masses in each box, in which case it is given by

$$L(t) = \sum_{i=1}^{N} \left(m_i(t) - \int_{\text{box } i} c(\mathbf{x}, t)\, d\mathbf{x} \right)^2, \tag{6}$$

where the integral is over the volume of box i. We shall say that, given two numerical representations which generate distributions approximating the solution $c(\mathbf{x}, t)$, the one which results in a smaller $L(t)$ gives a better fit to the solution at time t. $L(t)$ will be used only to measure the goodness of fit of numerical representations, and not as a basis for developing them.

Given two numerical representations, the one that gives a better fit for small t, for example at t_1, will not necessarily give a better fit for large t. Is it more

important that $L(t)$ be small for small t or for large t? In most cases when a numerical representation of the diffusion equation is used, one desires to know the mass distribution only after a large number of applications of the numerical algorithm. The mass distribution after only a few applications need not be exact, so long as the final result is sufficiently accurate. Accurate determination of the mass distribution in a certain region and after a certain time interval T requires either an analytical solution or its equivalent, the result of numerous applications of some satisfactory numerical algorithm having a time interval much shorter than the interval T.

In the next two sections we show how it is possible to make the numerical solution exactly model certain properties of the analytical solution. The numerical solution will possess these properties after an arbitrarily large number of applications of the numerical algorithm, and this leads to smaller values of $L(t)$ than for representations that do not exactly model these properties, as will be seen in the concluding section.

4. Ensuring Conservation of Mass, Nonnegativity of the Masses, and Stability

We show here that certain restrictions on the matrix $\mathbf{P}(s)$ are necessary and sufficient for the numerical representation with algorithm (5) to conserve mass and ensure nonnegativity of the masses. If $\mathbf{P}(s)$ satisfies these restrictions, then under certain rather general conditions the masses can be shown to be always finite. Furthermore, if $\mathbf{P}(s)$ and $\mathbf{S}(s)$ are independent of s, then the masses can be shown to converge to appropriate steady values.

In numerical representations based on finite difference methods, finding criteria for stability is often a major problem. In our representations based on the integral form of the diffusion equation, the problem is handled quite simply. The material in this section represents a generalization of the results of Bassett, Hewitt, and Martin [1], while the method used is also simpler and more convenient; all detailed proofs are given in Martin [2].

A. *Diffusion with No Sources or Sinks*

Let $\mathbf{S}(s) = \mathbf{0}$, so that (5) becomes

$$\mathbf{m}(t_{s+1}) = \mathbf{m}(t_s)\,\mathbf{P}(s). \tag{7}$$

It is easy to show that the restriction

$$\sum_{j=1}^{N} P_{ij}(s) = 1, \qquad i = 1,\ldots, N, \tag{8}$$

is necessary and sufficient to ensure conservation of mass, and that the restriction

$$P_{ij}(s) \geqslant 0, \qquad i, j = 1,..., N, \tag{9}$$

is necessary and sufficient to ensure that each mass always remains nonnegative. These two restrictions reflect a process by which a fraction $P_{ij}(s)$ of the mass in box i at time t_s moves to box j at time t_{s+1}; the sum of the fractions must equal one to conserve mass, and each fraction must be nonnegative to avoid negative masses. The restrictions (8) and (9) thus reflect properties of the physical system being modeled by the diffusion equation, and justify the terms transition probabilities for the quantities $P_{ij}(s)$.

A matrix is called stochastic if each of its elements is nonnegative and each row sum equals unity. Therefore a matrix satisfying (8) and (9) by definition is a stochastic matrix. It is easy to show that the product of any two stochastic matrices is stochastic (see [3, 4]). Therefore, if (8) and (9) hold for each s, the masses determined by each successive application of the algorithm (7) are guaranteed to remain nonnegative and to have a constant sum. Hence also each mass remains finite, and the representation is stable in the sense that errors do not grow without bound.

If $\mathsf{P}(s)$ is independent of s, (7) becomes

$$\mathbf{m}(t_{s+1}) = \mathbf{m}(t_s)\,\mathsf{P}. \tag{10}$$

The masses $\mathbf{m}(t)$ will converge to a set of limiting masses $\mathbf{m}^\infty$,

$$\mathbf{m}^\infty \equiv \lim_{s\to\infty} \mathbf{m}(t_s) = \lim_{s\to\infty} \mathbf{m}(t_0)\,\mathsf{P}^s \equiv \mathbf{m}(t_0)\,\mathsf{P}^\infty, \tag{11}$$

if and only if the matrix P^∞ of limiting transition probabilities exists. It is not hard to show that P^∞ exists if and only if P is not cyclic (P is cyclic if there exists an $n \geqslant 2$ such that $\mathsf{P}^n = \mathsf{P}$). An irreducible cyclic matrix has no diagonal elements; therefore the presence of at least one positive diagonal element is sufficient to ensure that P^∞ exists. Alternatively stated, if some of the original mass in at least one box remains there after a time interval, then the stochastic nature of P guarantees convergence to steady masses. A simple way to change a cyclic matrix into one having a nonzero diagonal element is to decrease the duration τ_s of each time interval by any amount, no matter how small (see for example the matrix elements presented in Section 6). The masses $\mathbf{m}^\infty$ to which $\mathbf{m}(t)$ converge may be considered "appropriate," in that they satisfy (10) with $\mathbf{m}(t_{s+1}) = \mathbf{m}(t_s)$.

B. *Diffusion in the Presence of Sources and Sinks*

In the theory of stochastic matrices a persistent state is one of a group of states (or in our terminology, boxes) from which mass cannot escape to other states not

in the group. Persistent states effectively may be considered to be the sinks in the set of boxes for the transport process. The simplest case of a persistent state is an absorbing state, for which $P_{ii}(s) = 1$. Once mass gets into such a state i, it cannot escape as long as state i is absorbing, for $P_{ij}(s) = 0, j \neq i$.

Instead of the conventional picture of mass transport causing loss of mass from the system by efflux across its boundaries or the like, in the numerical representation we may imagine such lost mass to have moved to a single additional box (now one of the set of N), which in this case would be an absorbing state. Only the transient states (defined as those states that are not persistent) correspond directly to the physical system being modeled. With this picture, loss of mass from the transient or physical system states may occur, while the transition probabilities still satisfy (8). The single absorbing state in the above example, or all the persistent states in general, may be considered to be the sinks for the transport process. The term $\mathbf{S}(s)$ then represents only sources, and each component may be taken to be nonnegative.

When there are sources $\mathbf{S}$ independent of s, it is not hard to show that if P is independent of time, the masses in the transient states will converge to appropriate steady values, while the masses in the persistent states grow without bound. If the sources or the transition probabilities vary with time, there is no set of limiting masses. In these situations it is possible to show that, if the set of persistent states is independent of s, the total mass in the transient states is always bounded. Thus under the conditions prevailing in most realistic problems, the restrictions guaranteeing mass conservation and nonnegativity of the masses ensure that the numerical representation will be stable, and when sources and diffusion parameters do not vary with s, give convergence to appropriate steady masses in those (transient) states corresponding to the physical system.

5. Moment-Fitting

Here we show that it may be possible to place restrictions on a numerical representation of the form (5), sufficient to ensure that certain of the moments of the numerical solution will always remain equal to corresponding moments of the solution to the diffusion equation. This method we call "moment-fitting." First we describe the method in general terms, then illustrate how it may be applied to a few specific problems, and finally comment on the use of the method.

A. *General Description of Moment-Fitting*

First consider certain moments of the analytical solution to the diffusion equation at time t, denoted by $\mu_{r_i}^{(c)}(t)$. The superscript (c) indicates that the moment is of the solution $c(\mathbf{x}, t)$ to the diffusion equation, and r_i denotes an element of a set R, each

of whose elements r_1 ,..., r_n stands for a particular moment. For example, we might have in a 1-dimensional unbounded problem, $r_i = i - 1$ and

$$\mu^{(c)}_{r_{i+1}}(t) \equiv \mu^{(c)}_i(t) = \int_{-\infty}^{\infty} x^i c(x, t)\, dx. \tag{12}$$

By appropriately manipulating the differential form of the diffusion equation, it may be possible to write each moment $\mu^{(c)}_{r_i}(t_{s+1})$ in terms of moments $\mu^{(c)}_{r_j}(t_s)$, $j = 1,..., n$:

$$\mu^{(c)}_{r_i}(t_{s+1}) = f_{i,s}(\mu^{(c)}_{r_1}(t_s),..., \mu^{(c)}_{r_n}(t_s), \quad \text{diffusion parameters, sources and sinks}), \qquad i = 1,..., n. \tag{13}$$

As well as depending on the moments $\mu^{(c)}_{r_j}(t_s), j = 1,..., n$, $\mu^{(c)}_{r_i}(t_{s+1})$ depends on the diffusion parameters, sources and sinks, such as D, $\mathbf{V}$ and $\mathscr{S}$ in (1) and (2). $\mu^{(c)}_{r_i}(t_{s+1})$ may depend on complicated functions of these quantities at different positions and times.

One general technique by which the functions $f_{i,s}$ in (13) may be obtained is to expand $c(\mathbf{x}, t_{s+1})$ in a power series about t_s :

$$c(\mathbf{x}, t_{s+1}) = \sum_{k=0}^{\infty} \frac{1}{k!} \frac{\partial^k c}{\partial t^k}\bigg|_{t_s} \tau_s^k, \tag{14}$$

where as before $\tau_s = t_{s+1} - t_s$. Each time derivative $(\partial^k c/\partial t^k)|_{t_s}$ in (14) may be expressed in terms of spatial derivatives of $c(\mathbf{x}, t_s)$ by repeated use of (1), with coefficients that are functions of the diffusion parameters, sources and sinks. By substituting the resulting expression for $c(\mathbf{x}, t_{s+1})$ into the expression for the moment $\mu^{(c)}_{r_i}(t_{s+1})$, and integrating out the spatial derivatives, a function of the form (13) may be obtained. The use of this technique will be illustrated in the examples given later. If the procedure is successful there will exist a closed set of n functions $f_{i,s}$ which are not unduly complicated.

Next consider corresponding moments of the numerical solution at time t_{s+1}, denoted by $\mu^{(m)}_{r_i}(t_{s+1})$. The superscript (m) indicates that the moment is of the numerical solution $\mathbf{m}(t)$. By use of the algorithm (5), it may be possible to write each moment $\mu^{(m)}_{r_i}(t_{s+1})$ in terms of moments $\mu^{(m)}_{r_j}(t_s)$, $j = 1,..., n$:

$$\mu^{(m)}_{r_i}(t_{s+1}) = g_{i,s}(\mu^{(m)}_{r_1}(t_s),..., \mu^{(m)}_{r_n}(t_s), \mathsf{P}(s), \mathbf{S}(s)), \qquad i = 1,..., n. \tag{15}$$

As well as depending on the moments $\mu^{(m)}_{r_j}(t_s), j = 1,..., n$, at time t_s, $\mu^{(m)}_{r_i}(t_{s+1})$ depends on the transition probabilities and sources in the numerical algorithm. If

the procedure is to be fully successful there will exist a closed set of n functions $g_{i,s}$.

Given the sets of functions $f_{i,s}$ and $g_{i,s}$, to make the numerical solution model at each time t_s each of the moments r_j, $j = 1,..., n$, it is sufficient to satisfy two conditions. First, each such moment of $\mathbf{m}(t)$ must equal the corresponding moment of $c(\mathbf{x}, t)$ at the initial time t_0:

$$\mu_{r_j}^{(m)}(t_0) = \mu_{r_j}^{(c)}(t_0), \qquad j = 1,..., n. \tag{16}$$

(16) is a set of equations to be solved for the initial masses $m_i(t_0)$, $i = 1,..., N$. Normally there will be an infinity of real solutions to these equations since $N \gg n$. Any solution to (16) will give the correct moments for $\mathbf{m}(t_0)$. One may choose from among the available solutions one that has other useful properties besides giving the correct moments. Most importantly, if possible the chosen solution should be nonnegative: $m_i(t_0) \geqslant 0$, $i = 1,..., N$. Not only is this physically realistic, but nonnegativity of the masses and stability can only be guaranteed if the masses are nonnegative at the initial time t_0. Among other criteria that may be used to choose among solutions to (16) that are nonnegative, convenience perhaps ranks highest; for example the first appropriate solution determined might be used.

Second, we require, assuming identity of the moments at time t_s ($\mu_{r_j}^{(m)}(t_s) = \mu_{r_j}^{(c)}(t_s) \equiv \mu_{r_j}$, $j = 1,..., n$), that

$$g_{i,s}(\mu_{r_1},..., \mu_{r_n}, \mathsf{P}(s), \mathbf{S}(s)) = f_{i,s}(\mu_{r_1},..., \mu_{r_n}, \text{diffusion parameters, sources and sinks}), \qquad i = 1,..., n, \qquad s \geqslant 0. \tag{17}$$

This condition and the condition given by (16) are sufficient to ensure that corresponding moments of the numerical and analytical solutions will always be equal. The Eqs. (17) provide n sets of restrictions on the transition probabilities $\mathsf{P}(s)$ and the sources $\mathbf{S}(s)$. If the procedure has been successful, there will usually be an infinity of solutions to (17). Whatever solution is chosen will be adequate to make the numerical solution model the moments r_j, $j = 1,..., n$. In this case there are several ways to decide which solutions to (17) to choose. Most importantly, if at all possible the probabilities should be nonnegative and so satisfy (9). Convenience usually will dictate that the number of nonzero transition probabilities will be as small as possible and that these values be grouped together on each row of $\mathsf{P}(s)$. Judgement must be used in choosing the most appropriate solution, which will depend upon the features of the particular problem at hand.

If $\mathbf{m}(t_0)$, $\mathsf{P}(s)$ and $\mathbf{S}(s)$ can be determined to satisfy (16) and (17), then at each time t_s, $s \geqslant 0$, it will be true that

$$\mu_{r_j}^{(m)}(t_s) = \mu_{r_j}^{(c)}(t_s), \qquad j = 1,..., n. \tag{18}$$

Thus given that we wish to ensure that the numerical representation models the moments as indicated in (18), we first determine a set of initial masses $\mathbf{m}(t_0)$ which satisfy (16). Then for each i and s transition probabilities and sources which satisfy (17) are used in the appropriate numerical algorithm.

B. *Moment-Fitting with a Simple 1-Dimensional Form of the Diffusion Equation*

Consider the 1-dimensional form of the diffusion equation,

$$\partial c(x, t)/\partial t = D(\partial^2 c(x, t)/\partial x^2) - V(\partial c(x, t)/\partial x), \tag{19}$$

where D and V are constants. We take the set of moments

$$\mu_j^{(c)}(t) = \int_{-\infty}^{\infty} x^j c(x, t)\, dx, \qquad j = 0,..., n. \tag{20}$$

We apply (14) to $c(x, t_{s+1})$, express each time derivative $(\partial^k c/\partial t^k)|_{t_s}$ as a sum of derivatives with respect to x by using (19) as many times as necessary, put the result in (20) and integrate by parts with respect to x until all spatial derivatives are removed. For example for $j = 2$ we have

$$\begin{aligned}
\mu_2^{(c)}(t_{s+1}) &= \int_{-\infty}^{\infty} x^2 c(x, t_{s+1})\, dx \\
&= \int_{-\infty}^{\infty} x^2 \left[c(x, t_s) + \frac{\partial c}{\partial t}\bigg|_{t_s} \tau_s + \frac{1}{2}\frac{\partial^2 c}{\partial t^2}\bigg|_{t_s} \tau_s^2 + \cdots\right] dx \\
&= \int_{-\infty}^{\infty} x^2 c(x, t_s)\, dx + \tau_s \int_{-\infty}^{\infty} x^2 \left\{D\, \frac{\partial^2 [c(x, t_s) + \tfrac{1}{2}(\partial c/\partial t)|_{t_s} \tau_s + \cdots]}{\partial x^2}\right. \\
&\quad \left. - V\, \frac{\partial [c(x, t_s) + \tfrac{1}{2}(\partial c/\partial t)|_{t_s} \tau_s + \cdots]}{\partial x}\right\} dx \\
&= \mu_2^{(c)}(t_s) + \tau_s \left\{2D \int_{-\infty}^{\infty} [c(x, t_s) + \cdots]\, dx \right. \\
&\quad \left. + 2V \int_{-\infty}^{\infty} x \left(c(x, t_s) + \frac{1}{2}\left[\cdots - V\, \frac{\partial [c(x, t_s) + \cdots]}{\partial x}\right] \tau_s\right) dx\right\} \\
&= \mu_2^{(c)}(t_s) + 2V\tau_s \mu_1^{(c)}(t_s) + (2D\tau_s + V^2 \tau_s^2)\, \mu_0^{(c)}(t_s).
\end{aligned}$$

In this and other calculations we assume that c and its spatial derivatives vanish as $x \to \pm\infty$. More generally we may write

$$\mu_j^{(c)}(t_{s+1}) = \sum_{k=0}^{j} \binom{j}{k} W_k(\tau_s)\, \mu_{j-k}^{(c)}(t_s). \tag{21}$$

The weight $W_k(\tau_s)$ is

$$W_k(\tau_s) = \sum_{l=0}^{[1/2k]} \frac{k!}{l!(k-2l)!} (D\tau_s)^l (V\tau_s)^{k-2l}. \tag{22}$$

Now consider moments of the numerical solution $\mathbf{m}(t)$. There are a number of possible types of moments that may be defined. We consider only one here, which is simple and convenient, and which results from assuming that each mass m_i is concentrated at a point x_i in box i. The jth moment of this distribution is

$$\mu_j^{(m)}(t_s) = \sum_{i=1}^{N} x_i^j m_i(t_s). \tag{23}$$

For simplicity assume each box is of length h and the center of box i_0 is at the origin. Setting $x_i = (i - i_0)\, h$, we have

$$\mu_j^{(m)}(t) = \sum_{i=1}^{N} h^j (i - i_0)^j\, m_i(t). \tag{24}$$

From (7), the appropriate numerical representation for (19), we know that at time t_{s+1} a fraction $P_{i\ i+k}(s)$ of the mass $m_i(t_s)$ will be in box $i + k$, for each i and k. Therefore we may determine $\mu_j^{(m)}(t_{s+1})$ as a function of the masses $m_i(t_s)$:

$$\begin{aligned} \mu_j^{(m)}(t_{s+1}) &= \sum_{i=1}^{N} h^j m_i(t_s) \sum_{k=-i+1}^{N-i} (i + k - i_0)^j\, P_{i\ i+k}(s) \\ &= \sum_{i=1}^{N} h^j m_i(t_s) \sum_{k=-i+1}^{N-i} \sum_{l=0}^{j} \binom{j}{l} (i - i_0)^{j-l}\, k^l P_{i\ i+k}(s). \end{aligned} \tag{25}$$

We shall assume what will be true in almost every practical case, that the nonzero (or significantly greater than zero) values of $P_{i\ i+k}$ are grouped near $k = 0$, and that the number of these nonzero values is much smaller than N. $\mathsf{P}(s)$ is calculated from the diffusion parameters D and V, the box length h, and the duration of the time interval τ_s. Since each of these parameters is constant here, nearly every row of the matrix $\mathsf{P}(s)$ can be made identical relative to the diagonal element. The exceptions to this rule occur near the ends of the array, where the probabilities must be modified. Here this detail will be ignored by assuming that little or no mass approaches the ends during the time under consideration. So for all i not near the ends of the array we may write

$$P_{i\ i+k}(s) = Q_k(s). \tag{26}$$

With the above assumptions, $\mu_j^{(m)}(t_{s+1})$ from (25) may be written

$$\mu_j^{(m)}(t_{s+1}) = \sum_{l=0}^{j} \binom{j}{l} h^l \sum_k k^l Q_k(s) \sum_{i=1}^{N} h^{j-l}(i - i_0)^{j-l} m_i(t_s)$$

$$= \sum_{l=0}^{j} \binom{j}{l} w_l(s) \mu_{j-l}^{(m)}(t_s). \tag{27}$$

The weight $w_l(s)$ is

$$w_l(s) = h^l \sum_k k^l Q_k(s). \tag{28}$$

Given the above expressions we may determine expressions for $\mathbf{m}(t_0)$ and $\mathsf{P}(s)$ which ensure that for each t_s

$$\mu_j^{(m)}(t_s) = \mu_j^{(c)}(t_s), \qquad j = 0,..., n. \tag{29}$$

The appropriate moments will be equal at t_0 if $\mathbf{m}(t_0)$ is a solution to

$$\mu_j^{(m)}(t_0) = \sum_{i=1}^{N} h^j(i - i_0)^j m_i(t_0)$$

$$= \int c(x, t_0)\, x^j\, dx = \mu_j^{(c)}(t_0), \qquad j = 0,..., n. \tag{30}$$

As noted before, it is most important to choose a nonnegative solution for $\mathbf{m}(t_0)$, and probably most convenient to take the first such solution that can be determined, since the results after long periods of time are unlikely to depend strongly on this choice. For most realistic distributions $c(x, t_0)$, at least one nonnegative solution will exist, at least if n is not too large.

that the corresponding moments $\mu_j^{(m)}$ and $\mu_j^{(c)}$ are equal at each later time t_s, the weights w_j and W_j must satisfy

$$w_j(s) = W_j(\tau_s), \qquad j = 0,..., n, \qquad s \geqslant 0. \tag{31}$$

Explicitly for $n = 4$ these relations are

$$\sum_k P_{i\,i+k}(s) = 1, \tag{32}$$

$$h \sum_k k P_{i\,i+k}(s) = V\tau_s, \tag{33}$$

$$h^2 \sum_k k^2 P_{i\,i+k}(s) = (V\tau_s)^2 + 2D\tau_s, \tag{34}$$

$$h^3 \sum_k k^3 P_{i\,i+k}(s) = (V\tau_s)^3 + 6(V\tau_s)(D\tau_s), \tag{35}$$

$$h^4 \sum_k k^4 P_{i\,i+k}(s) = (V\tau_s)^4 + 12(V\tau_s)^2(D\tau_s) + 12(D\tau_s)^2, \quad i = 1,..., N. \tag{36}$$

In Section 6 we will present typical solutions to these equations, and describe the success of the resulting representations on selected test problems.

C. *Some Generalizations of the Method*

Here we show how the moment-fitting method may be applied when the diffusion parameters are time-dependent and when the boxes are not of equal size.

If the diffusion parameters are functions of t but not of x, the diffusion equation (19) may be written

$$\partial c(x, t)/\partial t = D(t)[\partial^2 c(x, t)/\partial x^2] - V(t)[\partial c(x, t)/\partial x]. \tag{37}$$

The procedure in part A may be used to obtain a relation of the form

$$\mu_j^{(c)}(t_{s+1}) = \sum_{k=0}^{j} \binom{j}{k} W_k(t_s, t_{s+1})\, \mu_{j-k}^{(c)}(t_s). \tag{38}$$

The first three weights $W_j(t_s, t_{s+1})$ may be expressed in the following form.

$$W_0 = 1, \tag{39}$$

$$W_1 = \int_{t_s}^{t_{s+1}} V(t)\, dt, \tag{40}$$

$$W_2 = 2 \int_{t_s}^{t_{s+1}} \left[D(t) + V(t) \int_{t_s}^{t} V(t')\, dt' \right] dt. \tag{41}$$

Of course it may not be possible or convenient to calculate the weights such as (41), and numerical approximations to the various integrals might be used. Often for cases in which the parameters $D(t)$ and $V(t)$ do not vary significantly during a time interval, a representative value of each parameter during each interval may be used directly in (22), and satisfactory accuracy obtained.

When the box sizes are not uniform along the x-direction, an expression of the form (27) may still be obtained. Starting from the moment definition (23), and defining $h_{ij} = x_i - x_j$, we have

$$\begin{aligned}
\mu_j^{(m)}(t_{s+1}) &= \sum_{i=1}^{N} x_i{}^j m_i(t_{s+1}) \\
&= \sum_{i=1}^{N} m_i(t_s) \sum_k (x_i + h_{i+k\,i})^j\, P_{i\,i+k}(s) \\
&= \sum_{i=1}^{N} m_i(t_s) \sum P_{i\,i+k}(s) \sum_{l=0}^{j} \binom{j}{l} x_i^{j-l} h_{i+k\,i}^l \\
&= \sum_{l=0}^{j} \binom{j}{l} \sum_{i=1}^{N} m_i(t_s)\, x_i^{j-l} \sum_k P_{i\,i+k}(s)\, h_{i+k\,i}^l .
\end{aligned} \tag{42}$$

Comparing (42) to (27), (31) may be written

$$\sum_k P_{i\,i+k}(s)\, h^j_{i+k\,i} = W_j(\tau_s), \qquad i = 1,\ldots, N, \qquad j = 0,\ldots, n, \qquad s \geqslant 0. \quad (43)$$

If (43) is satisfied then the moment equations (29) may be satisfied for a numerical representation having nonuniform box sizes.

The moment-fitting method may also be applied with considerable rigor and without undue difficulty to more general parabolic equations, problems involving sources, problems in more than one dimension having nondiagonal diffusion coefficient matrices D, problems with absorbing and reflecting boundaries, and problems with spatially varying diffusion parameters. We hope to consider these problems in a later paper.

D. *Satisfying the Restrictions*

It may be noted that fitting of the zeroth moment, e.g. (32), is equivalent to (8), which ensures mass conservation. Also, fitting of the first moment, e.g. (33), provides for correct center of mass motion. Fitting of the higher moments tends to further increase the accuracy of the numerical solution. In particular, the fitting of the second moment, e.g. (34), in effect overcomes the problem of spurious diffusion often encountered when treating the term $\partial c/\partial x$ in (19). As we will see in the next section, it is easy to generate representations satisfying restrictions such as (32)-(36), and economical to compute with them. The only desirable properties for a numerical representation not necessarily satisfied by the fitting of low-order moments are nonnegativity of the masses and stability. When a nonnegative solution to (17) can be obtained, then along with the fitting of the zeroth moment, stability is guaranteed.

Obtaining a nonnegative solution to (17) for a particular set of moments is not always possible. In cases where it is not, one may have the choice between a solution with negative values which gives a *promise* of improved goodness of fit but a possibility of instability, or a nonnegative solution to (17) with some reduced set of moments, giving a possible loss of accuracy but a *guarantee* of stability. However sometimes this choice can be avoided. In some cases (see Section 6) increasing the time interval may make available a nonnegative solution. Also it may be possible to achieve the same effect by changing the sizes of the boxes, a procedure particularly appropriate when the diffusion parameters vary in space. In summary, when it is possible to obtain a nonnegative solution when fitting at least the first three moments (for 1-dimensional problems) then the resulting numerical representation almost certainly will adequately satisfy all the desirable properties listed in Section 3.

6. Results of Tests of Selected Representations

To illustrate the potential usefulness of the methods presented in this paper for developing specific numerical representations based on modeling properties of the solution to the diffusion equation, presented here are results of some tests of selected representations. First we describe the representations, then the criteria used to evaluate them in selected tests, and then the particular tests and the results obtained. Finally we discuss some alternative numerical methods for solving the diffusion equation which are not included in the tests.

A. *The Representations*

Each of the representations presented here is designed to give a numerical solution to the one-dimensional diffusion equation (19) or (37). Each representation is labeled for later reference (e.g. *I(a)*).

I. *Representations Based on the Differential Form of the Diffusion Equation, Using Finite Differences*

Grid points separated by a distance h are used, with one point at the origin. On each application of the specified algorithm the concentration change at each grid point is calculated. Write the diffusion equation (1) in the form

$$\partial c/\partial t = f(t, c). \tag{44}$$

Using simple finite difference approximations for the spatial derivatives in (19) or (37), the function $f(t, c)$ becomes

$$\begin{aligned} f(t, c) = {} & (D(t)/h^2)(c(x + h, t) - 2c(x, t) + c(x - h, t)) \\ & - (V(t)/2h)(c(x + h, t) - c(x - h, t)). \end{aligned} \tag{45}$$

Five different algorithms result from the use of the following approximations for the time derivative in (44).

I(a). *Point-slope formula.*

$$c(t_{s+1}) = c(t_s) + \tau_s f(t_s, c(t_s)). \tag{46}$$

I(b). *Second-order Runge–Kutta formula.*

$$\begin{gathered} c(t_{s+1}) = c(t_s) + \tfrac{1}{2}(k_1 + k_2), \\ k_1 = \tau_s f(t_s, c(t_s)), \qquad k_2 = \tau_s f(t_{s+1}, c(t_s) + k_1). \end{gathered} \tag{47}$$

I(c). *Second-order predictor–corrector formula.*

$$\begin{aligned} &\text{predictor: } c(t_{s+1}) = c(t_{s-1}) + (\tau_{s-1} + \tau_s) f(t_s, c(t_s)). \\ &\text{corrector: } c(t_{s+1}) = c(t_s) + \tfrac{1}{2}\tau_s(f(t_s, c(t_s)) + f(t_{s+1}, c(t_{s+1}))). \end{aligned} \tag{48}$$

(A second-order Taylor expansion is used for the first time interval.)

I(d). *Crank–Nicolson implicit method formula.*

$$c(t_{s+1}) = c(t_s) + \tfrac{1}{2}\tau_s(f(t_s, c(t_s)) + f(t_{s+1}, c(t_{s+1}))). \tag{49}$$

I(e). *Fully implicit method formula.*

$$c(t_{s+1}) = c(t_s) + \tau_s f(t_{s+1}, c(t_{s+1})). \tag{50}$$

I(f). Instead of the formula (45), an uncentered difference scheme is used to approximate the expression $\partial c/\partial x$. Equation (45) becomes

$$\begin{aligned} f(t, c) = {} & (D(t)/h^2)(c(x + h, t) - 2c(x, t) + c(x - h, t)) \\ & - \begin{cases} (V(t)/h)(c(x, t) - c(x - h, t)), & \text{if } V(t) > 0, \\ (V(t)/h)(c(x + h, t) - c(x, t)), & \text{if } V(t) < 0. \end{cases} \end{aligned} \tag{51}$$

The point-slope formula (46) is used to approximate the time derivative in (44).

II. *Representations Based on the Integral Form of the Diffusion Equation, Using Moment-Fitting*

As in Section 5B, each box in a linear array has length h and one box is centered on the origin. Listed here are the expressions for the only nonzero elements in the ith row of the matrix $\mathsf{P}(s)$. The number N of rows of each matrix is taken to be large enough that for the following tests only an insignificant amount of mass reaches the ends of the array by t_{1000}.

In these representations determined using (31), the particular row elements chosen here to have nonzero values are not unique, but seem likely to be typical. They are chosen first to minimize the number of nonzero elements, and second to make nonnegative values probable for most sets of diffusion parameters. For convenience in the representations we use the parameters

$$\begin{aligned} d(t) &= D(t)\,\tau_s/h^2, \\ v(t) &= V(t)\,\tau_s/h, \end{aligned} \tag{52}$$

where $D(t)$ and $V(t)$ are evaluated at time $\frac{1}{2}(t_s + t_{s+1})$, and the time dependences of d, v, and probabilities P_{ij} are not written explicitly.

II(a). *From* (31) *with* $n = 2$, *using* (22) *and* (28); *that is, a solution to* (32)–(34).

$$\begin{aligned} P_{i\,i+1} &= d + \tfrac{1}{2}v(v + 1), \\ P_{i\,i-1} &= d + \tfrac{1}{2}v(v - 1), \\ P_{i\,i} &= 1 - P_{i\,i+1} - P_{i\,i-1}. \end{aligned} \tag{53}$$

II(b). *From* (31) *with* $n = 3$, *using* (22) *and* (28); *that is, a solution to* (32)–(35). Define $\gamma = v(v^2 + 6d) - v$.
If $\gamma > 0$,

$$\begin{aligned} P_{i\,i+2} &= \gamma/6, \\ P_{i\,i+1} &= d + \tfrac{1}{2}(v(v+1) - \gamma), \\ P_{i\,i-1} &= d + \tfrac{1}{2}v(v-1) - P_{i\,i+2}, \\ P_{i\,i} &= 1 - P_{i\,i+2} - P_{i\,i+1} - P_{i\,i-1}. \end{aligned}$$

If $\gamma < 0$,

$$\begin{aligned} P_{i\,i-2} &= -\gamma/6, \\ P_{i\,i+1} &= d + \tfrac{1}{2}v(v+1) - P_{i\,i-2}, \\ P_{i\,i-1} &= d + \tfrac{1}{2}(v(v-1) + \gamma), \\ P_{i\,i} &= 1 - P_{i\,i-2} - P_{i\,i+1} - P_{i\,i-1}. \end{aligned}$$

II(c). *From* (31) *with* $n = 4$, *using* (22) *and* (28); *that is, a solution to* (32)–(36).

$$\begin{aligned} P_{i\,i+2} &= (d(12d - 2 + 12v(v+1)) \\ &\quad + v(-2 + v(-1 + v(2 + v))))/24, \\ P_{i\,i-2} &= (d(12d - 2 + 12v(v-1)) \\ &\quad + v(2 + v(-1 + v(-2 + v))))/24, \\ P_{i\,i+1} &= d + \tfrac{1}{2}v(v+1) - 3P_{i\,i+2} - P_{i\,i-2}, \\ P_{i\,i-1} &= d + \tfrac{1}{2}v(v-1) - P_{i\,i+2} - 3P_{i\,i-2}, \\ P_{i\,i} &= 1 - P_{i\,i+2} - P_{i\,i-2} - P_{i\,i+1} - P_{i\,i-1}. \end{aligned}$$

II(d). *From* (31) *with* $n = 2$, *using* (39)–(41) *and* (28), *for when the diffusion parameters vary with time.*

$$\begin{aligned} P_{i\,i+1} &= W_2\tau_s/h^2 + W_1\tau_s/h, \\ P_{i\,i-1} &= W_2\tau_s/h^2 - W_1\tau_s/h, \\ P_{i\,i} &= 1 - P_{i\,i+1} - P_{i\,i-1}. \end{aligned}$$

III. *Representation Giving a Perfect Fit at* t_1

The matrix of transition probabilities (identical for each s) is determined by setting $L(t_1) = 0$ (*see Eq.* (6)).

B. *The Criteria for Evaluation.*

Ease of Generation

The ease of generation of an algorithm may be roughly measured by the numbers of multiplications, additions, and equals required to prepare the algorithm for execution. An algorithm may be considered "ready for execution" when it is in the

form through which the same computations will take place for each application, for example, the form which is presented in a computer program. The numbers for ease of generation naturally will depend on what particular form of the algorithm is considered ready for execution. We assume the parameters d and v are given. Then, for example, the algorithm in I(a) may be considered to require only one multiplication and one equals to set up a parameter $v_h = \frac{1}{2}v$, while according to (53) the algorithm in II(a) requires four multiplications, six additions, and three equals.

Ease of Computation

The ease of computation may be measured reasonably well by the numbers of multiplications, additions, and equals required for each application of the algorithm and for each box or grid point. For example, our utilization of the algorithm in I(a) requires the following operations for each application and grid point.

$$\begin{aligned} \Delta c(x, t_s) &= d(t_s)(c(x + h, t_s) - 2c(x, t_s) + c(x - h, t_s)) \\ &\quad - v_h(t_s)(c(x + h, t_s) - c(x - h, t_s)), \\ c(x, t_{s+1}) &= c(x, t_s) + \Delta c(x, t_s). \end{aligned} \tag{54}$$

Total operations are three multiplications, five additions, and two equals. On the other hand, the algorithm in II(a) requires the following operations per application per box.

$$\begin{aligned} \tilde{m}_i(t_{s+1}) &= P_{i\,i-1}(s)\, m_{i+1}(t_s) + P_{i\,i}(s)\, m_i(t_s) + P_{i\,i+1}(s)\, m_{i-1}(t_s); \\ m_i(t_{s+1}) &= \tilde{m}_i(t_{s+1}). \end{aligned} \tag{55}$$

Total operations are three multiplications, two additions, and two equals.

The numbers presented for this criterion, as well as those for ease of generation, are not intended to be the lowest possible values attainable in each particular circumstance. Rather they are taken more or less straightforwardly from the relevant equations, and thus indicate a typical amount of computational effort to be expected when using the algorithms. For example note that (54) is by no means optimal as it could be recast in the form (55), with c's replacing the m_i's, making computation for I(a) easier but generation less easy.

Stability

We note which representations of type II are guaranteed to be stable because they satisfy the restrictions (8) and (9). Several of the representations of type I may be proved to be stable for one or more of the tests, but we do not inquire into this

problem here. However we note when any algorithm in any test exhibits obviously unstable behavior (namely, divergent oscillations between positive and negative values).

Goodness of Fit, Nonnegativity of the Masses, Mass Conservation, and Correct Center of Mass Motion

Each of these criteria is considered for times after 1, 10, 100, and 1000 applications of each algorithm ($t = t_1, t_{10}, t_{100}, t_{1000}$). The function $L(t)$ is used to measure goodness of fit; for type I representations, the mass associated with each grid point is calculated assuming a constant concentration in the interval $(x_i - \frac{1}{2}h, x_i + \frac{1}{2}h)$. The number of negative masses at each time may depend on the smallest number available on the computer used; in the present calculations done in single precision on a CDC Cyber 72 all numbers below about 10^{-288} become zero, so the number of negative masses is sometimes affected at t_{1000}. Any changes in the total mass or correct center of mass motion are noted, neglecting changes due to round-off error.

C. *The Tests and the Results*

In each of the following tests the analytical distribution being modeled arises from a unit delta function at the origin, with no sources or sinks present. In the numerical models the initial mass distribution is represented by either (type I representations) a unit concentration at the origin or (type II and III representations) a single unit mass in the box at the origin as indicated by (30). In the first four tests each of the representations conserves mass except I(d) and I(e). Each of these latter two representations, based on implicit methods, gives after the first iteration a considerably reduced total mass. On later iterations, as the mass distribution becomes smoother, the loss of mass per iteration is less. For example, in test type (ii) representation I(d) gives at times t_1, t_{10}, t_{100}, and t_{1000} total masses of .965, .949, .949, and .949, respectively. For tests type (i)–(iii) all representations give correct center of mass motion, except again I(d) and I(e).

Test type (i): $d = 0.2, v = 0$ (*constants*). With constant diffusion parameters generation of the algorithm must be carried out only once; computation with the algorithm is identical for each iteration and each box. Results of the tests of type (i) for the criteria of goodness of fit are given in Table I. None of the representations gave any negative masses or concentrations, except I(c) which gave 6 and 94 negative concentrations at t_{10} and t_{100}, respectively. With $v = 0$, I(f), II(a), and II(b) become identical with I(a), and results are not presented separately. All type II representations are guaranteed stable by the results in Section 4.

Test type (ii): $d = 0.2, v = 0.1$ (constants). With $v \neq 0$, these parameters give a situation typically troubled by spurious diffusion. All comments for test type (i) apply, except that results are in Table II.

TABLE I

Results of Test Type (i) for Goodness of Fit ($1.1(-3) = 1.1 \times 10^{-3}$)

Representation	$L(t_1)$	$L(t_{10})$	$L(t_{100})$	$L(t_{1000})$
I(a)	1.1(−3)	4.0(−6)	1.3(−8)	4.0(−11)
(b)	3.7(−2)	2.3(−4)	5.5(−7)	1.7(−9)
(c)	3.7(−2)	1.9(−4)	5.4(−7)	1.7(−9)
(d)	2.3(−2)	3.0(−4)	5.0(−5)	1.8(−5)
(e)	5.0(−2)	1.0(−3)	1.2(−4)	4.3(−5)
II(c)	3.2(−3)	1.2(−5)	3.6(−8)	1.1(−10)
III	0	7.0(−4)	2.7(−4)	[a]

[a] No results due to computational effort required.

TABLE II

Results of Test Type (ii) for Goodness of Fit ($1.7(-3) = 1.7 \times 10^{-3}$)

Representation	$L(t_1)$	$L(t_{10})$	$L(t_{100})$	$L(t_{1000})$
I(a)	1.7(−3)	4.0(−5)	6.0(−6)	1.7(−6)
(b)	3.9(−2)	3.5(−4)	4.1(−6)	1.1(−7)
(c)	3.9(−2)	3.2(−4)	4.2(−6)	1.2(−7)
(d)	2.5(−2)	1.2(−2)	3.9(−2)	2.7(−2)
(e)	5.1(−2)	8.4(−3)	3.5(−2)	2.5(−2)
(f)	8.1(−3)	1.0(−3)	3.3(−4)	1.0(−4)
II(a)	9.7(−4)	8.0(−6)	1.7(−7)	5.1(−9)
(b)	1.6(−3)	5.8(−6)	1.8(−8)	5.7(−11)
(c)	3.1(−3)	1.2(−5)	3.6(−8)	1.1(−10)
III	0	7.0(−4)	2.8(−4)	[a]

[a] No results due to computational effort required.

Test type (iii): $d = 0.1$, $v = 0.5$ (constants). These parameters give a situation typically troubled by negative masses as well as spurious diffusion. All comments for test type (i) apply, except that results, including number of negative masses, are in Table III, and no type II representations are guaranteed stable by the results in Section 4. In Section 5D it was noted that by changing the time interval it is often easy to avoid spurious diffusion while still guaranteeing nonnegative masses. This may be accomplished in the present case by using representation II(a) with the time interval τ_s increased by a factor, among others, of 1.25 ($d = 0.125$, $v = 0.625$).

TABLE III

Results of Test Type (iii) for Goodness of Fit and Nonnegativity of the Masses
(1.4(−1) = 1.4 × 10^{-1})

Representation	$L(t_1)$	$L(t_{10})$	$L(t_{100})$	$L(t_{1000})$	Number of negative masses at t_1, t_{10}, t_{100}, t_{1000}
I(a)	1.4(−1)	1.8(−1)	5.7(−1)	[b]	1, 8, 74, [b]
(b)	1.4(−1)	1.7(−2)	5.7(−4)	1.8(−5)	1, 11, 121, 562
(c)	1.4(−1)	2.6(−2)	1.2(−3)	4.1(−5)	1, 11, 122, 584
(d)	1.3(−1)	3.5(−1)	7.0(−2)	2.0(−2)	1, 8, 95, 980
(e)	9.9(−2)	1.3(−1)	6.4(−2)	2.0(−2)	1, 7, 79, 892
(f)	5.6(−2)	2.0(−2)	6.2(−3)	1.9(−3)	0, 0, 0, 0
II(a)	5.7(−3)	6.5(−4)	2.3(−5)	7.3(−7)	1, 6, 66, 231
(b)	1.7(−2)	4.5(−4)	1.5(−6)	4.5(−9)	1.9, 99, 186
(c)	8.5(−3)	1.5(−4)	2.5(−7)	6.6(−10)	2, 14, 156, 193
III	0	5.0(−3)	1.7(−3)	[a]	0, 0, 0, [a]

[a] No results due to computational effort required.
[b] Unstable behavior.

Because of the increased time interval, the distribution resulting from this altered representation is at t_8, t_{80}, and t_{800} comparable to the other representations at t_{10}, t_{100}, and t_{1000}. Hence we note that for the representation with increased time interval, $L(t_8) = 6.9 \times 10^{-4}$, $L(t_{80}) = 2.0 \times 10^{-5}$, and $L(t_{800}) = 6.4 \times 10^{-7}$, with no negative masses generated, a guarantee of stability, and 0.8 as much computational effort required. Note that this technique cannot be used for representations such as I(a).

Test type (iv). $d(t) = 0.2 + 0.1 \mid \sin 2\pi t/20 \mid$, $v(t) = \frac{1}{2}d(t)$; $t_s = s$. With time-varying diffusion pramaters, generation of the algorithm must be carried out once for each different set of values of the parameters. With the parameters $d(t)$ and $v(t)$ given above this requires six generations. Results of the tests of type (iv) for the criteria of goodness of fit and center of mass motion are given in Table IV. The center of mass motion figure is the fractional deviation from the correct figure. Only I(c) produced any negative masses. All type II representations are guaranteed stable by the results of Section 4.

Test type (v). For nonuniform box sizes or grid point spacing we here only briefly summarize the results obtained. Representations of type I, when generalized in a natural manner for nonuniform grid point spacing, do not conserve total mass,

TABLE IV

Results of Test Type (iv) for Goodness of Fit and Center of Mass Motion
$(3.6(-3) = 3.6 \times 10^{-3})$

Representation	$L(t_1)$	$L(t_{10})$	$L(t_{100})$	$L(t_{1000})$	Fractional deviation from correct center of mass motion at t_1	Fractional deviation from correct center of mass motion at t_{10}, t_{100}, t_{1000}
I(a)	3.6(−3)	5.0(−5)	9.5(−6)	3.6(−6)	−7.2(−2)	−2.0(−3)
(b)	4.2(−2)	2.1(−4)	4.2(−6)	1.3(−6)	−6.0(−4)	−2.0(−3)
(c)	4.2(−2)	8.7(−4)	[a]	[a]	3.0(−4)	−1.9(−3)
(d)	3.0(−2)	6.4(−3)	1.5(−2)	2.1(−2)	−9.2(−1)	−5.6(−1)
(e)	4.8(−2)	9.0(−3)	3.5(−2)	2.1(−2)	4.1(−2)	5.2(−1)
(f)	5.2(−3)	1.0(−3)	2.7(−4)	8.3(−5)	−7.2(−2)	−2.0(−3)
II(a)	7.3(−4)	4.0(−5)	1.4(−6)	3.5(−7)	3.0(−4)	1.0(−3)
(b)	9.2(−4)	2.2(−6)	7.6(−8)	2.0(−7)	3.0(−4)	1.0(−3)
(c)	2.8(−3)	5.2(−6)	6.5(−8)	2.0(−7)	3.0(−4)	1.0(−3)
(d)	7.3(−4)	3.9(−5)	9.5(−7)	2.9(−8)	0	0

[a] Unstable behavior.

and as in the case of representations I(d) and I(e) on the other tests, this leads to very poor goodness of fit after long periods of time. Representations of type II conserve total mass, and perform better according to the other criteria as well.

D. *Discussion of Results*

The results in Tables I–IV show that when comparing two numerical representations, the one producing a better fit after one application does not necessarily produce a better fit after numerous applications. Specifically, representation III with a perfect fit at t_1 gives a poor fit relative to most other representations at later times. Also in comparing any two representations based on moment-fitting, the one modeling more moments is more likely to give a better fit when the time of comparison is larger. In Table IV we see that as the time of comparison becomes [redacted] successful compared to the other representations, the centers of mass of whose numerical distributions slowly but steadily become displaced from the center of mass of the analytical solution.

Finally, in Table V we list characteristic numbers for ease of generation and ease of computation, determined by inspection of the algorithms presented in part A above. These figures demonstrate that while representations II(a)–(c), which model moments of the diffusion equation with constant parameters, require considerable

TABLE V

Characteristic Values for Ease of Generation and Computation for the Sample Representations, Listed as Numbers of Multiplications, Additions, and Equals Per Different Set of Diffusion Parameters (Generation) or Per Box or Grid Point and Per Application of the Numerical Algorithm (Computation)

	Generation	Computation
I. Representations based on the differential form of the diffusion equation, using finite differences.		
(1) Using (45) for the space derivatives, and for the time derivative,		
(a) point-slope formula (46):	1, 0, 1	3, 5, 2
(b) second-order Runge–Kutta formula (47):	1, 0, 1	7, 11, 4
(c) second-order predictor–corrector formula (48):	1, 0, 1	8, 11, 4
(d) Crank–Nicolson method formula (49):	2, 4, 6	8, 5, 5
(e) fully implicit method formula (50):	1, 3, 3	5, 3, 4
(2) Using (51) with an uncentered difference for the space derivative, and for the time derivative,		
(f) point-slope formula (46):	0, 0, 0	3, 5, 2
II. Representations based on the integral form of the diffusion equation, using moment-fitting.		
(a) conserving 3 moments:	4, 6, 3	3, 2, 2
(b) conserving 4 moments:	9, 11, 5	4, 3, 2
(c) conserving 5 moments:	22, 26, 5	5, 4, 2
(d) conserving 3 moments when diffusion parameters vary with time:	depends on difficulty of integrations	3, 2, 2
III. $L(t_1) = 0$:	lengthy	$K, K-1, 2$ $K \approx 15$

effort to generate the transition probabilities for the algorithm, after generation is completed the computational effort required is quite reasonable. So if the transition probabilities in such a representation are constant or change in time in a regular manner, then the total ease of generation and computation for the representation, measured by execution time on a computer, will be smaller than for any but the very simplest of alternative representations.

In complicated problems it will of course not usually be possible to determine $L(t)$ or other measures of the success of a numerical representation. But when using representations based on modeling properties of the solution to the diffusion equation, one will know that, for example, the solution has no negative masses, or has the same first three moments as the exact solution. The examples in this

section, and in particular the measure $L(t)$, are designed to show that, by ensuring that it has certain desirable properties, the numerical solution will indeed be satisfactory.

E. *Comments on Other Methods for Numerically Solving the Diffusion Equation.*

In the previous tests we have considered a cross-section of representations based on finite differences (type I), sample representations obtained using the method proposed in this paper (type II), and a representation illustrating that a good fit at small times does not necessarily lead to a good fit at large times (type III). As well as these representations, there exists a wide variety of other approaches to the numerical solution of the diffusion equation (see [5, Section 2.2]). Here we discuss a number of the methods not included in the above tests.

Other representations based on the differential form of the diffusion equation using finite differences (see, for example, [6, pp. 93–95]) cannot be expected to be significantly more successful than the ones tested here, unless the representation is designed for a special problem. In particular, implicit methods tend to give poor results because of nonconservation of mass.

The representation of Bassett *et al.* [1] is based on the integral form of the diffusion equation; however for purposes other than conservation of mass and nonnegativity of the masses, it is equivalent to $I(f)$ on tests (i)–(iv).

The method of Egan and Mahoney [7] involves as part of the numerical algorithm explicit calculation of the first three moments of the concentration in each box. When applied to the diffusion equation (see their Appendix) this method involves a large computational effort, seems to require small time intervals to attain reasonable accuracy, and gives results hard to interpret; the center of mass of a "grid-element" may move right out of the box with which it is associated.

The finite element approach when applied to the diffusion equation seems to require finite difference approximations for the time derivative, and except for special problems appears to give no better results than representations obtained using finite differences [8].

In methods using "Lagrangian" point masses, the movement of each point mass is calculated individually, and the mass in a box at a given time is the aggregate of all the mass points in that box. Since the mass points move independently of the set of boxes, this method is quite different from the one presented in this paper. Applied to the diffusion equation, the most successful of these methods use random displacements of the point masses to simulate diffusion. The accuracy of such methods depends on the number of point masses used. However in most cases, to obtain an accuracy comparable with that of the methods tested here, the necessary computational effort is orders of magnitude larger (see [9] and discussion thereafter).

Finally, it is possible to solve fluid dynamical equations by numerically following the motion of suitable markers of constant concentration. When applied to the diffusion equation [10] this method is limited by its use of finite differences, and except for special problems appears not to be superior to conventional approaches based on finite differences.

ACKNOWLEDGMENTS

I wish to thank Professor H. Messel and the Science Foundation for Physics at Sydney University for supporting this work. In addition, personal support was received from a Commonwealth Postgraduate Studentship.

I would like to thank the following people for their help: Hugh Comins for many conversations; Dr. R. G. L. Hewitt for various discussions and for encouraging the inclusion of actual numerical results; George Vorlicek for helpful readings of the manuscript; Jos Beunen for painstakingly reading the manuscript and offering several useful comments; and Dr. I. M. Bassett for kindly reading several versions of portions of the manuscript and providing copious stimulating criticism.

REFERENCES

1. I. M. BASSETT, R. G. L. HEWITT, AND B. MARTIN, *Mon. Weath. Rev.* **101** (1973), 528.
2. B. MARTIN, A numerical method for diffusion problems, Ph.D. thesis, University of Sydney, 1975.
3. F. R. GANTMACHER, "Applications of the Theory of Matrices," Interscience, New York, 1959.
4. W. FELLER, "An Introduction to Probability Theory and Its Applications," Vol. 1, Wiley, New York, 1968.
5. B. A. BOLEY, *Nucl. Engng. Design* **18** (1972), 377.
6. R. D. RICHTMYER, "Difference Methods for Initial-Value Problems," Interscience, New York, 1957.
7. B. A. EGAN AND J. R. MAHONEY, *J. Appl. Meteor.* **11** (1972), 312.
8. G. E. MYERS, "Analytical Methods in Conduction Heat Transfer," McGraw–Hill, New York, 1971.
9. A. HAJI-SHEIKH AND E. M. SPARROW, *J. Heat Transfer* **89** (1967), 121.
10. R. C. DIX AND J. CIZEK, The isotherm migration method for transient heat conduction analysis, *in* "Heat Transfer 1970," Vol. I, Elsevier, Amsterdam, 1970.