A Numerical Transport Scheme which Avoids Negative Mixing Ratios[1]

HANS-RAINER SCHNEIDER[2]

Rosenstiel School of Marine and Atmospheric Sciences, University of Miami, Miami, FL 33149
Laboratory of Planetary Atmospheres, NASA/Goddard Space Flight Center, Greenbelt, MD 20771

(Manuscript received 15 October 1983, in final form 17 February 1984)

ABSTRACT

A numerical transport scheme that avoids the problem of spurious generation of negative mixing ratios has been developed. The scheme is computationally simple in any number of spatial dimensions, can be used with nonperiodic boundary conditions, and preserves shapes reasonably well. It is based on the idea of formulating a quadratically conservative finite-difference advection equation and then advecting the square root of the concentration instead of the concentration itself. The requirement of step-by-step quadratic conservation in time places restrictions on the time-differencing method. A modified Lax-Wendroff procedure is shown to be suitable under this constraint and is used here. The transport scheme has properties comparable to fourth-order centered spatial differencing combined with leapfrog time-stepping and filling.

1. Introduction

In the study of atmospheric tracer transport, large gradients in the mixing ratio of a trace constituent, as well as localized sources and sinks, are often encountered. Most finite-difference methods for the integration of the continuity equation do not perform well when the quantity to be advected is constrained to be positive and when steep gradients and/or rapid changes in slope occur. Since a strongly peaked, positive definite function of finite support cannot be represented well on a finite grid or with a truncated series, almost every numerical scheme tends to generate overshoots in regions where gradient changes occur. This usually results in negative mixing ratios and unwanted reduction of real peaks in the advected function. An illustration of the problem is shown in Fig. 1a where a narrow Gaussian distribution has been advected with constant velocity across 150 grid points using a leapfrog scheme with fourth-order spatial differencing. The exact solution is indicated by the dashed line.

Perhaps the simplest way to prevent overshooting is to use an advection scheme with sufficient inherent diffusion. For example, upstream differencing avoids negative mixing ratios and has a very good phase speed. The Matsuno scheme also leads to a relatively small phase error and generates only small negative mixing ratios. These methods are too diffusive for many applications, however. Fig. 1b shows the deformation of an initially wedge-shaped distribution after it has been advected with uniform velocity over a distance of 150 grid spaces using the two methods mentioned above.

Another method used to avoid negative mixing ratios is filling. After an advection time step, grid boxes are checked for negative mixing ratios and brought to zero by borrowing in a suitable way from neighboring boxes. The problem with filling is that it becomes quite complicated and somewhat ambiguous for multidimensional applications and higher order spatial differencing schemes. It is also difficult to devise a filling algorithm that can be easily vectorized for modern high-speed computers.

Mahlman and Sinclair (1977) have compared a number of transport algorithms both with and without filling, using the advection of a narrow wedge-shaped distribution as a test problem. They found that a pseudospectral method gives very good results, but only if a very large number of harmonics is computed and the time step is very small. The spectral method combined with a small time step was also found superior to second- and fourth-order spatial differencing by Orszag (1971). We have obtained a comparable result with finite elements, but this method is about as computationally efficient as the use of high spectral resolution. None of the other schemes tested by Mahlman and Sinclair is completely satisfactory; however, it appears that the fourth-order centered spatial differencing combined with a leapfrog time stepping and filling gives the best results. Therefore, our scheme will be compared against this method.

Another class of techniques, not tested by Mahlman and Sinclair (1977), that can be used to integrate the continuity equation are the flux corrected transport (FCT) algorithms (Boris and Book, 1973, 1976; Book *et al.*, 1975; Zalesak, 1979, 1981a,b) and the partial

[1] Contribution No. 9, Stratospheric General Circulation with Chemistry Modeling Project at NASA/GSFC.

[2] Present affiliation: Applied Research Corporation, 8201 Corporate Drive, Landover, MD 20785.

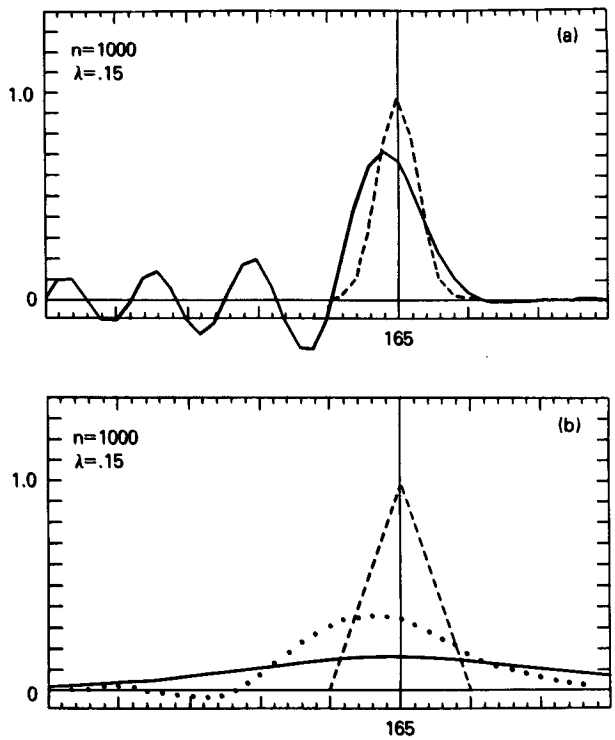

FIG. 1. (a) Result of advecting a narrow Gaussian over 150 grid points with the standard fourth-order leapfrog method. Dashed line shows the analytic solution; n is the number of time steps performed to arrive at the final distribution, and λ is the CFL-parameter. See text for discussion. (b) Result of using upstream differencing (solid line) and the second-order (spatial) Matsuno scheme (dotted line) to advect an initially wedge-shaped distribution the same distance.

donor cell (PDM) method (Hain, 1978). These schemes were developed for the numerical treatment of hydraulic shocks and are a combination of a low and a higher order differencing scheme. The FCT technique consists of two steps; the first is an advection with the dispersive low-order scheme, and the second is a nonlinear, antidispersive or corrective step that eliminates the overshoots produced by the first step in such a way that no new maxima occur. This correction depends on the gradients and velocities. The PDM refers to a method used to reduce the diffusion in the classic donor cell scheme, i.e., upstream differencing, to the minimum amount necessary to prevent the generation of new extrema.

Both PDM and FCT, which are actually classes of schemes, work very well for square waves and shock problems, maintaining steep gradients almost exactly. Some variations of these techniques perform less well for strongly peaked distributions. For example, in the FCT method the first step tends to draw out the peak and the corrective step does not completely restore it; however, these schemes have several degrees of freedom and Zalesak (1981b) has described flux-limiting parameters that maintain shapes other than shock fronts well. A two-dimensional comparison between two FCT methods and various other algorithms, not mentioned here, has been done by Chock and Dunker (1983). The biggest drawback of the FCT techniques is that they become quite complicated in more than one space dimension.

A simpler, efficient scheme based on the same idea as FCT was developed by Smolarkiewicz (1983). It uses upstream differencing combined with a correction step to counteract the inherent diffusion. The correction is less complicated than in the FCT methods. The scheme was found to have small implicit diffusion and to give results comparable to the more expensive techniques for the test problem, which was chosen by Smolarkiewicz. Independent tests showed, however, that the remaining numerical diffusion in the scheme prevents it from giving good results with the narrow test shapes used in Section 3 of this paper. Another scheme that can give good results is the so-called slopes scheme (Russel and Lerner, 1981). This scheme uses flux-limiting criteria, and its complexity increases considerably with the number of spatial dimensions. There are many more methods described in the literature, and no attempt is made here to give a complete enumeration.

In this paper we will describe one more method for integrating the transport equation. This method, called the square-root scheme (SRS), has the advantage of being very simple, both conceptually and computationally. Furthermore, the square-root scheme is easily generalized to two and three dimensions and boundary conditions other than periodic.[3] We will describe the scheme in Section 2, and in Section 3 several test results and a comparison to a conventional leapfrog scheme with filling will be presented.

2. The square-root scheme

The square-root scheme is based on the idea of advecting the square root of a concentration instead of the concentration itself in such a way that total mass is conserved. Advecting the square root one step still results in overshoots and negative values near the steep gradients, but we are only interested in the square of the result of the advection. Squaring back after the step to obtain the concentration, the small negative lobes become even smaller positive lobes. By analogy with a filling algorithm, the filling is done automatically and negative boxes are actually slightly overfilled. The tests in Section 3 will show that this does not lead to appreciable errors, but can lead to the generation of noise. This problem also will be discussed in Section 3.

[3] [After this paper was originally submitted, it was found that the version of SRS using the trapezoidal implicit time scheme as described in Section 2c, had been proposed earlier by Wright (1972). However, to our knowledge the properties of the scheme have not been investigated.]

a. Derivation of the square-root scheme

In the absence of sources and sinks, the advection equation for a quantity C is

$$\frac{\partial C}{\partial t} + (\mathbf{v}\cdot\nabla)C = 0. \tag{1}$$

The form of the square-root scheme is simplest when the continuity equation is formally nondivergent as is the case for homogeneous fluids or Boussinesq fluids, or if pressure coordinates are used. For the following derivation, we will assume pressure coordinates. The modifications necessary in other coordinates and for explicitly time-varying densities will be described in the Appendix. The fluid continuity equation is then

$$\nabla\cdot\mathbf{v} \equiv \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial \omega}{\partial p} = 0, \tag{2}$$

where u and v are the horizontal velocity components in the x and y directions, respectively, and ω is the time rate of change of the pressure p, following the fluid parcel. If the vertical coordinate is a function of pressure, such as $\log(p)$, the last term in (2) has to be modified as discussed in the Appendix. All the following arguments still hold, however.

Eq. (1) can be written in flux form as

$$\frac{\partial C}{\partial t} + \nabla\cdot(\mathbf{v}C) = 0. \tag{3}$$

Analytically, all moments of C are conserved, and if we define

$$\chi = C^{1/2},$$

Eq. (3) is trivially satisfied if

$$\frac{\partial}{\partial t}\chi + \nabla\cdot(\mathbf{v}\chi) = 0. \tag{4}$$

Omitting the constant gravity acceleration, the constraint for mass conservation is now

$$\frac{d}{dt}\int \chi^2 dxdydp = 0, \tag{5}$$

and the goal is to construct a finite-difference analog to (4) such that

$$\sum_s (\chi_s^{n+1})^2 = \sum_s (\chi_s^n)^2. \tag{6}$$

Here, and through the remainder of this paper, superscript n refers to the time step, and subscript $s = (i, j, k)$ to the spatial grid location. Uniform grid spacing is assumed in (6).

Equation (6) requires quadratic conservation at each time step. This is a stronger constraint than that which is normally understood by "quadratic conservation." Usually, this term refers only to the space derivatives in (4). However, the constraint (6) also places restrictions on the time differencing method. To be more specific, we repeat briefly the usual argument (e.g., Haltiner and Williams, 1980, pp. 177 and 235). Equation (4) is written in semifinite-difference form as

$$\frac{\partial}{\partial t}\chi \approx -\delta(\mathbf{v}, \chi), \tag{7}$$

where the right-hand side represents some finite-difference expression for the divergence. Multiplying both sides by χ and integrating over space, assuming no boundary contributions, the operator δ is called quadratically conservative if it fulfills the condition

$$\sum_s \chi_s\delta(\mathbf{v}, \chi)_s = 0. \tag{8}$$

However, if we write the time derivative in finite-difference form also choosing, for example, the leapfrog scheme,

$$\chi_s^{n+1} = \chi_s^{n-1} - 2\Delta t\delta(\mathbf{v}, \chi^n)_s, \tag{9}$$

and then again multiply both sides by χ_s^n and sum over all space indices, assuming that (8) holds, we see that what is actually conserved is

$$\sum_s \chi_s^{n+1}\chi_s^n. \tag{10}$$

Not every term in this sum is necessarily positive, and therefore cannot be interpreted as a density or mixing ratio. Even if it could be so interpreted, $\chi_s^{n+1}\chi_s^n$ would represent a concentration at a half time step, $C_s^{n+1/2}$. However, since other terms in the advection equation such as the sources and sinks are given at integral time levels, it is also necessary to obtain the concentration and conserve total mass at integral levels.

Condition (6) rules out most explicit, centered-in-time schemes and many uncentered schemes. In order to find a suitable time integration, consider the forward scheme, ignoring for the moment the fact that it is unstable:

$$\chi_s^{n+1} = \chi_s^n - \Delta t\delta(\mathbf{v}, \chi^n)_s. \tag{11}$$

Squaring both sides of this equation and summing over s, assuming again that (8) is satisfied, gives

$$\sum_s (\chi_s^{n+1})^2 = \sum_s (\chi_s^n)^2 + \Delta t^2 \sum_s [\delta(\mathbf{v}, \chi^n)_s]^2. \tag{12}$$

This means that mass conservation is violated to $O(\lambda^2)$, where λ is the CFL parameter, given by

$$\lambda = v\Delta t/\Delta x,$$

v being a typical magnitude of the velocity and Δx a grid distance appearing in the operator δ.

The nonconservation of mass can be reduced to $O(\lambda^4/4)$ by replacing (11) with

$$\chi_s^{n+1} = \chi_s^n - \Delta t\delta(\mathbf{v}, \chi^n)_s + \tfrac{1}{2}\Delta t^2\delta(\mathbf{v}, \tau^n)_s. \tag{13}$$

In order to simplify writing, we have defined

$$\tau_s^n \equiv \delta(\mathbf{v}, \chi^n)_s,$$

and both notations will be used in the following.

Equation (13) defines the square-root scheme used in the remainder of this paper. This scheme is a modified version of the Lax–Wendroff method. Its conservation and numerical properties will be investigated in the following sections.

b. Conservation properties and spatial differencing

Squaring both sides of (13) and integrating, we obtain

$$\sum_s (\chi_s^{n+1})^2 = \sum_s (\chi_s^n)^2 + \Delta t^2 \sum_s \tau_s^n \delta(\mathbf{v}, \chi^n)_s + \Delta t^2 \sum_s \chi_s^n \delta(\mathbf{v}, \tau^n)_s - 2\Delta t \sum_s \chi_s^n \delta(\mathbf{v}, \chi^n)_s - \Delta t^3 \sum_s \tau_s^n \delta(\mathbf{v}, \tau^n)_s + \tfrac{1}{4}\Delta t^4 \sum_s [\delta(\mathbf{v}, \tau^n)_s]^2. \quad (14)$$

If the expression for the space derivatives satisfies the condition

$$\sum_s [\alpha_s \delta(\mathbf{v}, \beta)_s + \beta_s \delta(\mathbf{v}, \alpha)_s] = 0 \quad (15)$$

for all fields α and β, Eq. (13) reduces to

$$\sum_s (\chi_s^{n+1})^2 = \sum_s (\chi_s^n)^2 + \tfrac{1}{4}\Delta t^4 \sum_s [\delta(\mathbf{v}, \tau^n)_s]^2. \quad (16)$$

The second term in this equation is normally very small and can easily be compensated for, as will be described in Section 3.

The constraint (15) on the spatial derivatives is, for practical purposes, the same as the usual constraint (8) for quadratic conservation. Most of the time, a finite-difference operator that fulfills (8) also will satisfy the more general form (15). To give an example for the operator δ, let us assume that the velocity components satisfy the second-order finite difference continuity equation, i.e.,

$$\frac{1}{2\Delta x}(u_{i+1} - u_{i-1}) + \frac{1}{2\Delta y}(v_{j+1} - v_{j-1}) + \frac{1}{2\Delta p}(\omega_{k+1} - \omega_{k-1}) = 0, \quad (17)$$

and that fourth-order differences are to be used for the advection equation. Then (15) is fulfilled if we write

$$\frac{\partial}{\partial x}(u\chi) = \frac{1}{4\Delta x}\{a[(u_{i+1} + u_i)(\chi_{i+1} + \chi_i) - (u_i + u_{i-1})(\chi_i + \chi_{i-1})] + b[u_{i+1}(\chi_{i+2} + \chi_i) - u_{i-1}(\chi_i + \chi_{i-2})]\}. \quad (18)$$

The y and z derivatives may be formulated analogously. Nonessential indices have been suppressed in Eqs. (17) and (18). It is further assumed that all quantities are carried at the same grid points. The constants a and b in (18) are the relative weights of the second- and fourth-order differences; the values $a = \frac{4}{3}$ and $b = -\frac{1}{3}$ are used throughout this paper. The finite-difference operator (18) conserves the first and second moment of χ for periodic boundary conditions. With other boundary conditions, only one moment can be conserved, in general.

c. Numerical properties of the modified Lax–Wendroff scheme and comparison with other methods

In order to compare the scheme (13) to other commonly used methods, we write Eq. (13) for the special case of one space dimension, constant velocity u and second-order spatial differencing. Then (13) becomes

$$\chi_i^{n+1} = \chi_i^n - \tfrac{1}{2}\lambda(\chi_{i+1}^n - \chi_{i-1}^n) + \tfrac{1}{8}\lambda^2(\chi_{i+2}^n - 2\chi_i^n + \chi_{i-2}^n), \quad (19)$$

with λ again being the CFL parameter.

The Lax–Wendroff scheme can be defined as a two-step procedure (e.g., Haltiner and Williams, 1980, p. 149):

$$\left.\begin{aligned} \chi_i^* &= \tfrac{1}{2}(\chi_i^n + \chi_{i+1}^n) - \tfrac{1}{2}\lambda(\chi_{i+1}^n - \chi_i^n) \\ \chi_i^{n+1} &= \chi_i^n - \lambda(\chi_i^* - \chi_{i-1}^*) \end{aligned}\right\}. \quad (20)$$

That is, a forward step to half points in space and time is performed first, followed by a forward step from time n to $n + 1$ using the space derivatives at the intermediate level. If instead the intermediate step is made to a half level in time and full points in space,

$$\left.\begin{aligned} \chi_i^* &= \chi_i^n - \tfrac{1}{4}\lambda(\chi_{i+1}^n - \chi_{i-1}^n) \\ \chi_i^{n+1} &= \chi_i^n - \tfrac{1}{2}\lambda(\chi_{i+1}^* - \chi_{i-1}^*) \end{aligned}\right\}, \quad (21)$$

and the two steps are combined, Eq. (19) is obtained. In passing, we note that the Matsuno scheme differs from (19) by having a factor $\lambda^2/4$ instead of $\lambda^2/8$ in front of the last term on the right-hand side. Therefore, the Matsuno scheme does not satisfy the requirement of step-by-step quadratic conservation.

The modified Lax–Wendroff scheme (19) or (21) is weakly unstable. The amplification factor A for fourth-order spatial differences with weights a and b as in (18) is given by

$$A^2 = 1 + \tfrac{1}{4}\lambda^4 F^4, \quad (22)$$

where

$$F = a \sin k\Delta x + \tfrac{1}{2}b \sin 2k\Delta x. \quad (23)$$

Since the instability is proportional to $\lambda^4/8$, it is not critical for practical applications.

The modified Lax–Wendroff scheme is not the only possible time integration method that can be used with the square-root scheme. For the sake of completeness, we will comment on two more procedures. Equation (13) is the second-order truncation of the translation operator

$$T = \exp[-\Delta t \nabla(\mathbf{v},)]. \quad (24)$$

The scheme corresponding to the third-order truncation,

$$\chi_s^{n+1} = \chi_s^n - \Delta t\delta(\mathbf{v}, \chi^n)_s + \tfrac{1}{2}\Delta t^2\delta(\mathbf{v}, \tau^n)_s - \tfrac{1}{6}\Delta t^3\delta[\mathbf{v}, \delta(\mathbf{v}, \tau^n)]_s, \quad (25)$$

is conditionally stable with an amplification factor

$$A_{(3)}^2 = 1 - \frac{\lambda^4}{12}F^4 + \frac{\lambda^6}{36}F^6. \quad (26)$$

The nonconservation of mass is reduced to

$$\sum_s \left(-\frac{\Delta t^4}{12}[\delta(\mathbf{v}, \tau^n)_s]^2 + \frac{\Delta t^6}{36}\{\delta[\mathbf{v}, \delta(\mathbf{v}, \tau^n)]_s\}^2\right). \quad (27)$$

There is, however, no practical advantage in using this method as compared to the second-order time scheme. Test results are almost indistinguishable from those obtained with (13), and the violation of mass conservation for both methods is near the truncation error of a computer in single precision.

Table 1 shows the amplification factor and the ratio of the numerical phase speed c to the true advection velocity u for the modified Lax–Wendroff scheme (13) as well as for the third-order time scheme (25) in one dimension and for fourth-order space derivatives. The phase speed for the leapfrog time stepping with fourth-order spatial differencing is shown for comparison. The three methods have almost identical properties. In particular, we see that the amplification factor for the modified Lax–Wendroff scheme is very close to 1 and the phase speed for waves of all wavelengths is practically the same as for the leapfrog time differencing.

TABLE 1. Amplification factor A and ratio of the numerical phase speed c to the true advection velocity u for different values of the CFL parameter and different wavelength $N\Delta x$. Fourth-order spatial differencing has been used for all three schemes.

		Second order (time)		Third order (time)		Leap-frog
	N	A	c/u	A	c/u	c/u
$\lambda = 0.1$	2	1.0000	0.000	1.0000	0.000	0.000
	4	1.0004	0.851	0.9999	0.849	0.851
	6	1.0001	0.966	0.9999	0.965	0.966
	8	1.0000	0.989	0.9999	0.988	0.989
	12	1.0000	0.992	0.9999	0.998	0.998
$\lambda = 0.2$	2	1.0000	0.000	1.0000	0.000	0.000
	4	1.0006	0.858	0.9998	0.849	0.859
	6	1.0002	0.971	0.9999	0.965	0.971
	8	1.0001	0.992	0.9999	0.988	0.992
	12	1.0000	0.999	0.9999	0.998	0.999
$\lambda = 0.4$	2	1.0000	0.000	1.0000	0.000	0.000
	4	1.0101	0.885	0.9969	0.851	0.895
	6	1.0033	0.990	0.9989	0.966	0.993
	8	1.0011	1.004	0.9996	0.988	1.005
	12	1.0002	1.005	0.9999	0.998	1.005

TABLE 2. Ratio of the numerical phase speed c to the true advection velocity u for the trapezoidal implicit scheme with fourth-order spatial differencing. N indicates the wavelength $N\Delta x$.

N	$\lambda = 0.4$	$\lambda = 1.0$
2	0.00	0.00
4	0.78	0.59
6	0.91	0.76
8	0.96	0.84
12	0.98	0.92

The trapezoidal implicit scheme,

$$\chi_s^{n+1} = \chi_s^n - \tfrac{1}{2}\Delta t[\delta(\mathbf{v}, \chi^{n+1})_s + \delta(\mathbf{v}, \chi^n)_s], \quad (28)$$

can also be used for the advection of the square root of a concentration. It not only has the advantage of being unconditionally stable, but also conserves mass exactly, without any correction terms (M. R. Schoeberl, personal communication, 1983; Wright and Weidlinger, 1972). Unfortunately, the numerical phase speed for this method is not very good for large values of the CFL parameter. Table 2 shows the ratio of the numerical phase speed to the true advection velocity for this scheme for various values of λ and fourth-order spatial differencing. For large λ, where the additional computational effort for an implicit scheme would be justified, the distribution would be significantly retarded.

3. Tests and comparisons

All the results presented in this section are for the modified Lax–Wendroff scheme (13) with fourth-order spatial differencing. As a standard for comparison in one dimension, we have chosen the fourth-order centered leapfrog scheme with filling. The filling algorithm we used is a simple form of downstream borrowing (Mahlman, 1973). If a box is negative, we try to borrow first from the two downstream gridboxes, and if this is not possible, from the two upstream boxes; if this also fails, we march in the direction of the maximum of the concentration and take from the first box that has a sufficient amount of constituent material. With this method, we obtained results consistent with those of Mahlman and Sinclair (1977).

Tests in one dimension are presented for four different initial conditions, a 10-grid-point-wide wedge, a highly skewed wedge with the leading flank extending over only three grid spaces, a square distribution with rounded (Gaussian) edges, and a narrow Gaussian with a half-width of two grid points. The initial distributions are advected with constant velocity over a distance of 150 grid spaces. The results from this transport experiment are shown at the same time, using two different time steps corresponding to $\lambda = 0.3$ and 0.1, that is, after 500 and 1500 time steps, respectively. The figures show a window of 40 grid points. The

whole grid consists of 250 equally spaced points. At time step 0, the peak of the test distributions was located at grid point 15. Periodic boundary conditions were used.

Figures 2a, b, c and d and 3a, b, c and d display the final distributions obtained with the leapfrog filling method. The results are in agreement with Mahlman and Sinclair's (1977) finding that this scheme behaves

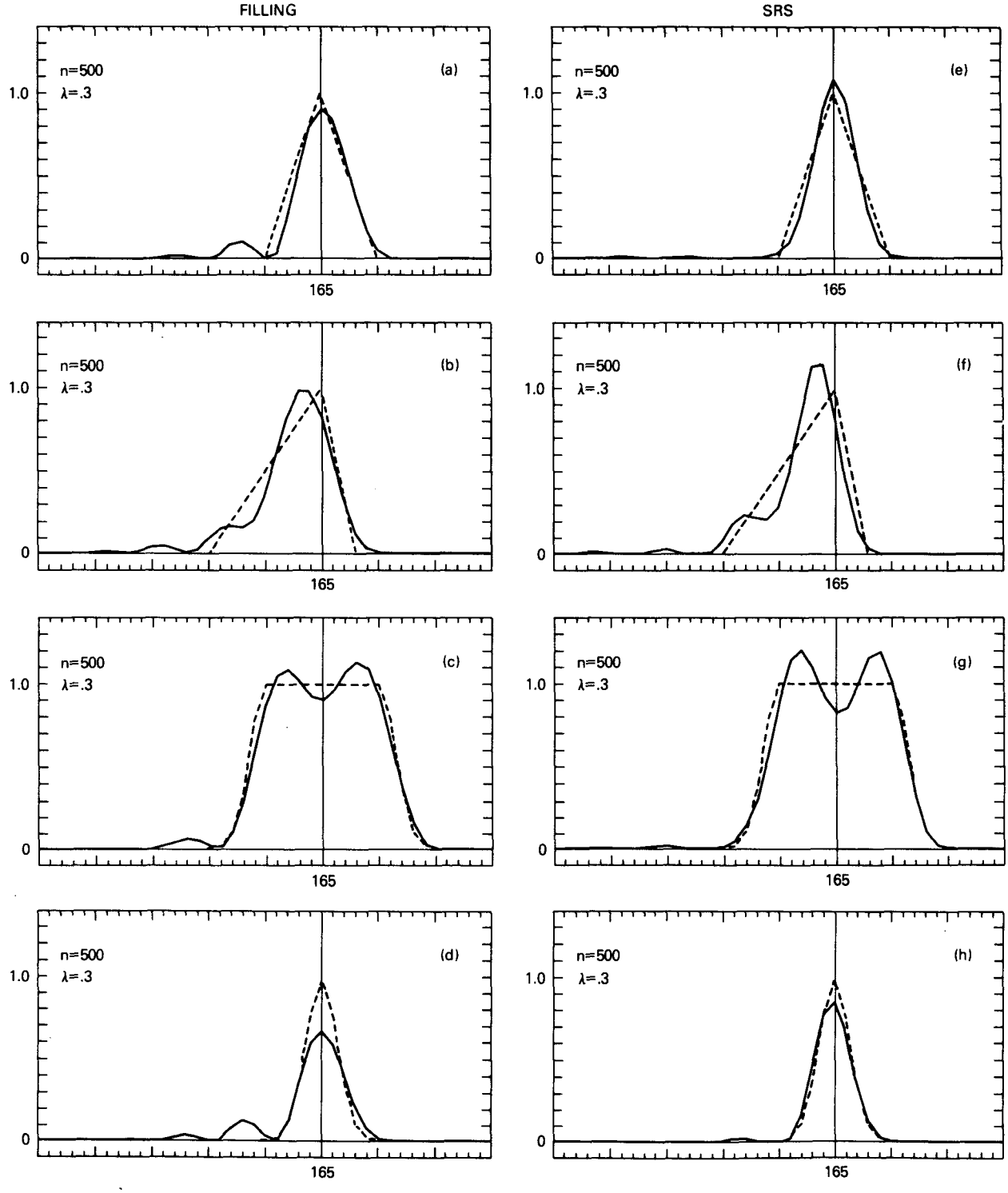

FIG. 2. Advection of various test shapes over 150 grid points. (a), (b), (c) and (d) show results of the leapfrog filling scheme, and (e), (f), (g) and (h) results of the square-root scheme. Dashed line is the exact solution, which is centered at grid point 165. The time step corresponds to $\lambda = 0.3$, and distributions are shown after 500 steps.

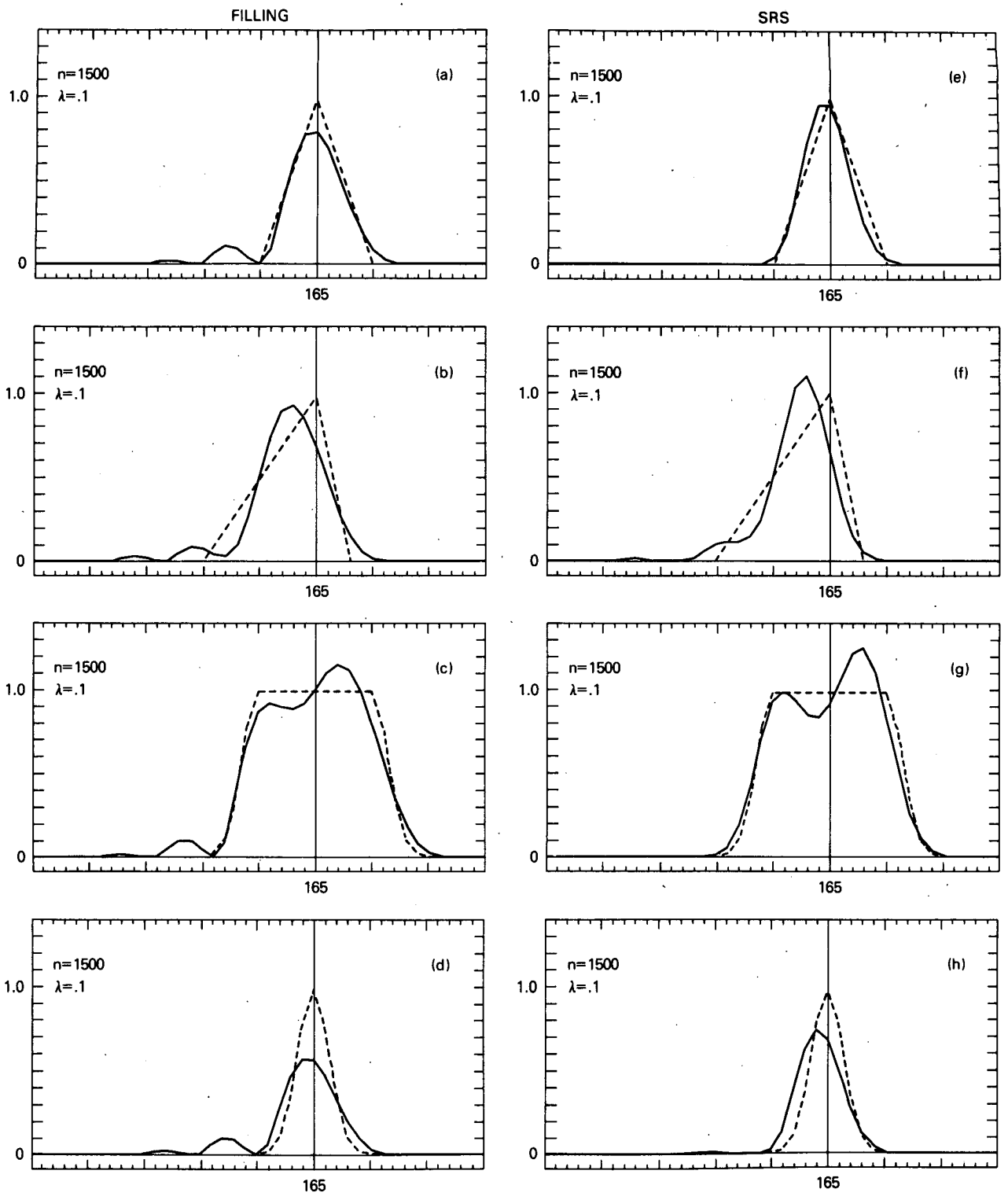

FIG. 3. As in Fig. 2, but for $\lambda = 0.1$, and after 1500 steps.

best for relatively large time steps, and not quite as well for very small steps. Decreasing the time step below $\lambda = 0.1$ does not, however, lead to an increased deterioration from the 0.1 case.

Figures 2e, f, g and h and 3e, f, g and h show the behavior of the square-root scheme for the same initial conditions and the same values of the CFL parameter. These results were obtained as follows: At each step,

we took the square root of the concentration C^n, χ^n, then computed χ^{n+1} according to (13) and obtained the new concentration at time $n + 1$ as

$$C^{n+1} = \{(\chi^{n+1})^2 - \tfrac{1}{4}\Delta t^4[\delta(\mathbf{v}, \tau^n)]^2\}^{1/2}. \quad (29)$$

The Δt^4 correction term was only subtracted if it did not lead to negative values under the square-root sign. All the corrections that could not be subtracted in this way were summed up to give the total error. Whenever the total error accumulated to 10^{-5} times the total initial mass, it was subtracted uniformly wherever possible. This condition occurred in the one-dimensional tests with $\lambda = 0.3$ about every 10–20 steps depending on the initial distribution, but with $\lambda = 0.1$ and smaller it occurred less than once in a thousand steps.

There is no compelling reason to subtract the correction term pointwise. Equation (16) holds only if the summation is carried out. However, we found this to be the most convenient treatment, and since the term is small, the way in which it is redistributed does not seem to be critical. Note that there is no additional computing effort required to obtain the correction. If one routine is used to compute the term proportional to Δt in (13), namely $\tau_s^n = \delta(\mathbf{v}, \chi^n)_s$, the same routine can be used again to obtain the divergence of τ^n which is the Δt^2 term in (13), and whose square is proportional to the above correction term.

As is apparent from the figures, the square-root scheme does not produce any trailing lobes, and the results do not depend as much on the time step as in the case of fourth-order differencing with filling. Both methods have a tendency to make the skewed triangle symmetrical. The filling method is clearly inferior in the case of the narrow Gaussian. For the square distribution the trailing flank is reproduced a little more accurately with the square-root technique than with the filling technique. On the other hand, there is a larger overshoot at the leading edge.

A tendency to generate overshoots can be seen for almost all the test cases, especially if the time step is relatively large. Figure 4 shows the result of the square-root scheme for three of the initial distributions where we have passed a boxcar filter,

$$\bar{C}_i = 0.05C_{i-1} + 0.9C_i + 0.05C_{i+1}, \quad (30)$$

over the distribution every 50 time steps in order to reduce this tendency. This is the only result shown in this paper that has been obtained by using any type of filter.

An important drawback of the square-root scheme is the generation of noise. Squaring the result of an advective step generates a distribution with a larger high wavenumber content. It takes comparatively long for the high frequency components to be damped out, since the time-differencing scheme used is not very diffusive. Figures 5a, b, c and d show the advection of our test wedge over the first 25 grid points, starting from point 15, and it can be seen that the noise is

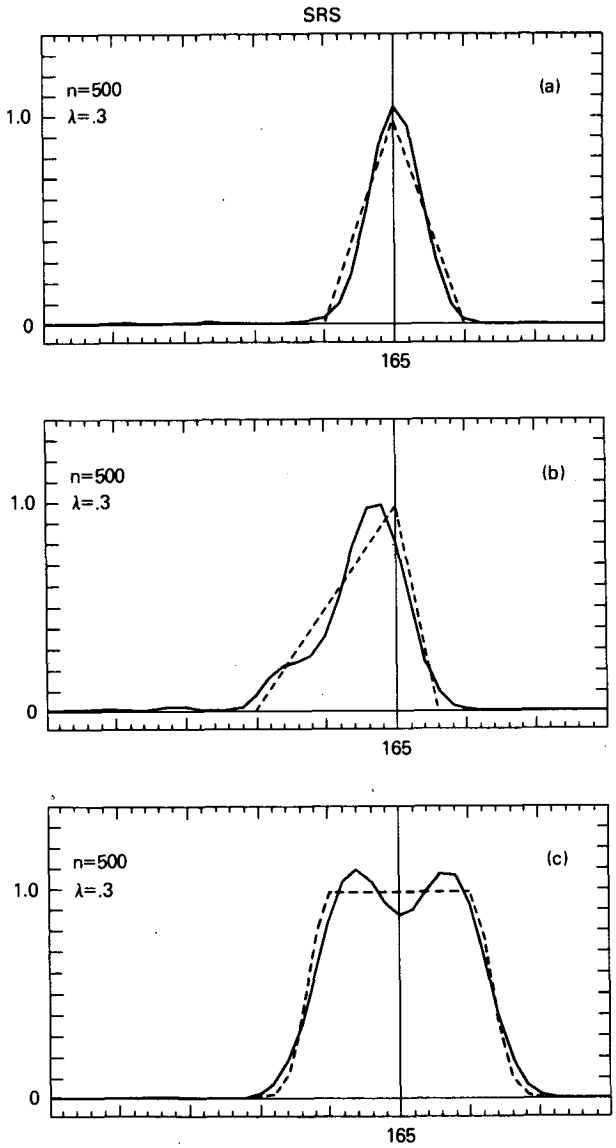

FIG. 4. Results of advection where a boxcar filter has been passed over the distributions every 50 time steps (see text). Time steps and CFL parameters are as indicated.

significant until the triangle has been transported across about 15 grid spaces. This may be related to the fact that the derivatives are not defined everywhere for a triangle shape. For the narrow Gaussian, almost no noise is generated and the results are smooth from the beginning, as can be seen in Figs. 5e and f.

In Fig. 6, the solid-body rotation of a Gaussian blob on a flat rectangular grid is shown. In this case, the boundaries are specified as open and about 1% of the total mass has been lost through them at the end of the integration. The upper-left panel shows the initial condition, a Gaussian with a 2.5-grid-distance half-width, cutoff outside a square. Because of the cutoff, we notice some distortion after a rotation of 55°, as

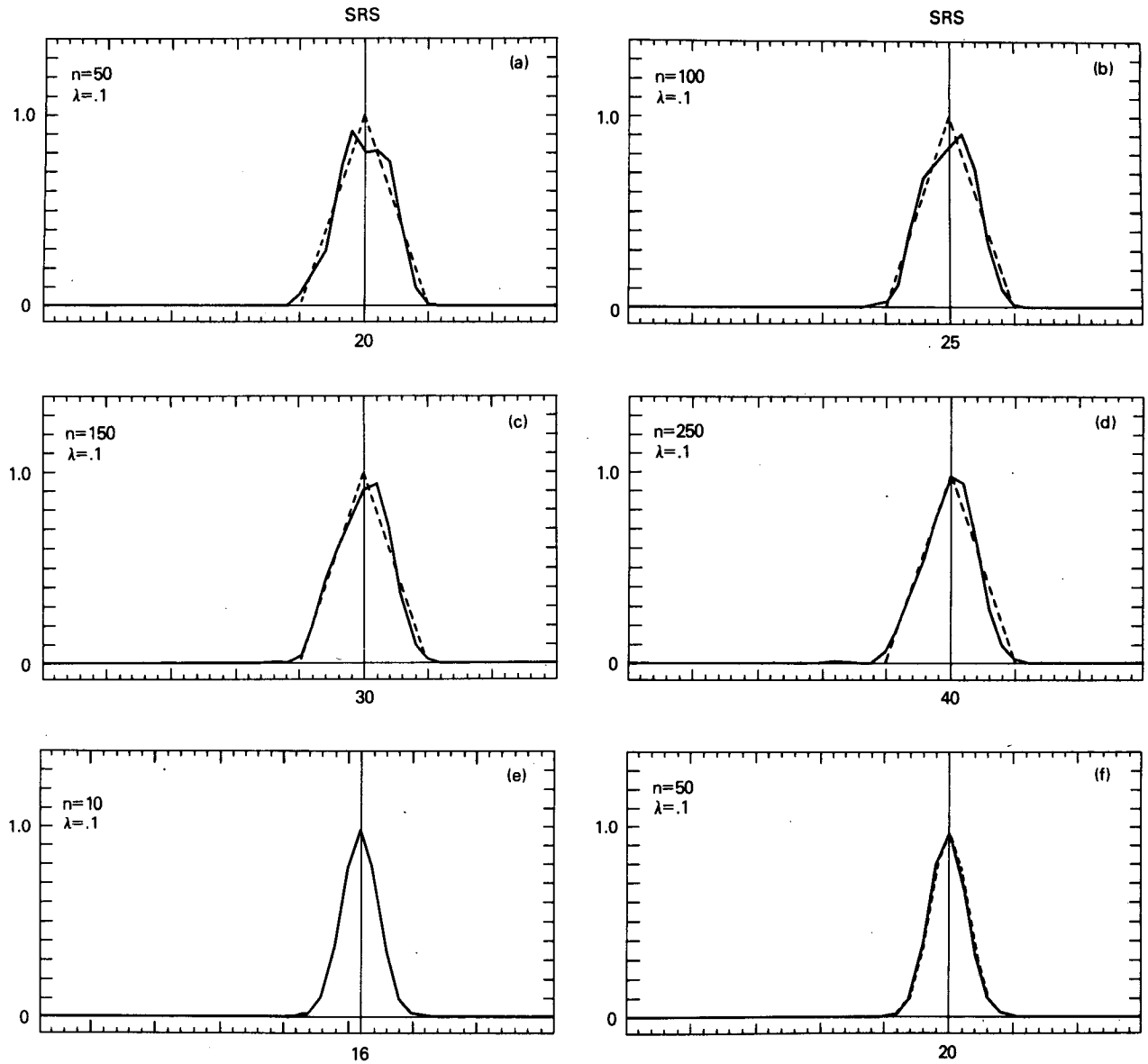

FIG. 5. Distributions obtained with the square-root scheme after 50 (a), 100 (b), 150 (c) and 200 (d) steps. The exact position of the wedge, initially located at grid point 15, is indicated by the dashed line; (e) and (f) show the advection of a Gaussian after 10 and 50 time steps, respectively.

in the one-dimensional tests. The shape becomes smooth soon after that, and a steepening of the peak by 6% can be observed after somewhat more than one full rotation. After one more rotation, the peak mixing ratio starts to drop slowly. The exact position of the maximum is indicated in the figures, and to permit an estimate of the noise level, the 0.05 contour has been drawn in. Other tests in two dimensions with highly skewed distributions yielded results consistent with those shown in one dimension; that is, a slight tendency to retard the peak and to symmetrize the distribution can be observed.

We want to emphasize at this point that the extension of the square-root scheme to higher dimensions is completely natural and does not require any special treatments beyond what has been described already. This is its main advantage over flux-limiting and filling techniques.

Figure 7 gives an example of the use of SRS with impenetrable boundaries. The figure shows a meridional cross section with the North and South Poles on the right and left, respectively. The grid is spherical and the density varies with height, the scale height being taken as 7 km. A blob of tracer material, initially situated near one pole, is advected by a meridional velocity field reminiscent of the diabatic circulation in the stratosphere. The positions of a few points as obtained from a trajectory computation are indicated.

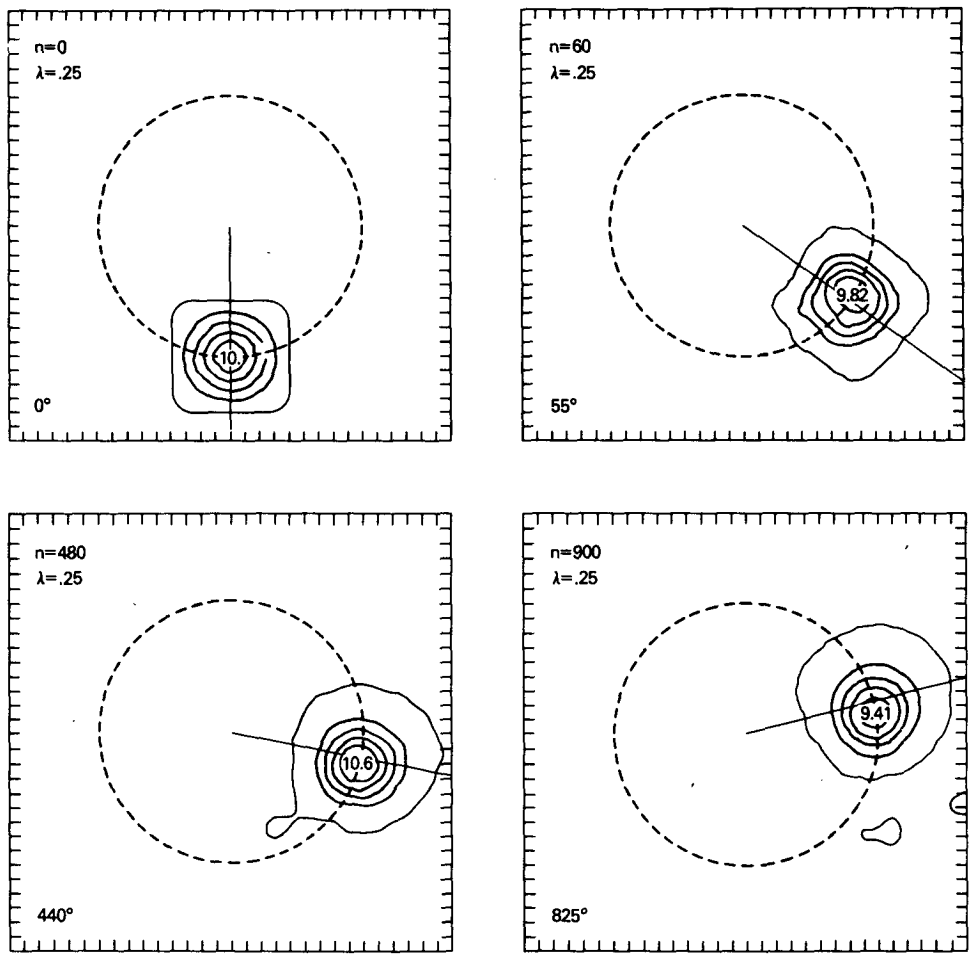

FIG. 6. Rotation on a rectangular flat grid, using the square-root scheme. Results are shown after 60, 480 and 900 steps, corresponding to rotations by 55, 480 and 825°, respectively. The exact position of the maximum of the distribution is the intersection of the straight line with the circle. The contour interval is 2.0, and the 0.05 contour (outer contour) also has been drawn.

Again, the 0.05 contour has been added. The figure shows the mixing ratio, not the amount of material per box. The drop in the mixing ratio after 23 days is due to the compression of the initial parcels, which causes the mixing ratio, averaged over a grid box, to be reduced. Minor noise can be noticed upstream of the distribution, the phase speed of $2\Delta x$ waves being zero; because of the spherical geometry, the noise increases towards the pole.

4. Summary and discussion

A new class of schemes for numerically integrating the transport equation has been developed. The square-root method (SRS) avoids the problem of negative mixing ratios by using the square root of the concentration instead of the concentration itself in an advective time step. In order to ensure conservation of total mass at every time step, the concept of quadratic conservation had to be extended to the numerical time integration method. We have described a few time schemes that fulfill the requirement of step-by-step quadratic conservation. The simplest of these is the modified Lax–Wendroff method (13).

The main advantage of the SRS technique as compared to other methods is its simplicity. It can be easily generalized to three dimensions and non-periodic boundary conditions can be incorporated. The performance of the SRS is comparable to a leapfrog method combined with a filling algorithm.

Since a Lax–Wendroff-type time differencing with its intermediate step is used, or in other words since we have to compute the divergence of two quantities, the scheme uses twice as much computer time as a leapfrog method without filling. In addition, the square root of the mixing ratio at each grid point has to be taken before every time step, and a whole field has to be squared after every step. Furthermore, correction terms have to be kept track of in order to ensure exact conservation of mass. In pressure coordinates there is

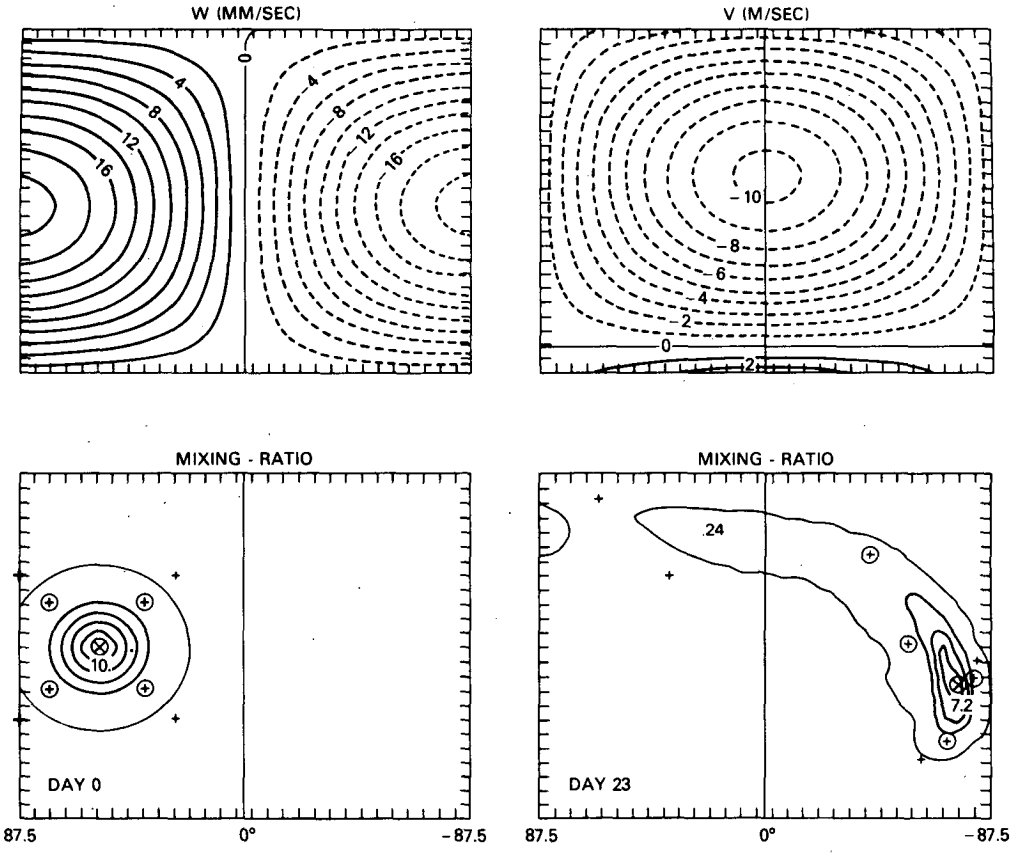

FIG. 7. Advection in a meridional plane. Spherical coordinates are used and the scale height for the density stratification is 7 km. The lower boundary is at 10 km and open. There is no flux through the upper boundary at 100 km. Lateral boundaries are the poles. The upper left and right panels show the vertical and horizontal velocity fields, respectively. The initial condition is displayed in the lower left panel, and the lower right panel shows the result of the square-root scheme after 23 days. Contour interval is 2.0, and the 0.05 contour also is shown. The initial and final position of a few points, as obtained from a trajectory computation, are indicated.

only one term, but in general there can be four terms, as is shown in the Appendix. These correction terms are small and require little additional programming effort. These disadvantages have to be weighed against the use of a filling algorithm in three dimensions or the computational effort required by some flux-correcting techniques.

Another disadvantage of the square-root scheme is the generation of noise which is damped out only slowly for some initial conditions.

The scheme cannot compete with FCT or PDM techniques for the numerical treatment of very sharp gradients. However, its shape-preserving properties might still be good enough for many atmospheric applications where fourth-order accuracy combined with the advantage of avoiding negative mixing ratios at a low cost is sufficient.

Acknowledgments. The author is indebted for many helpful discussions and suggestions to Drs. M. R. Schoeberl, M. A. Geller and R. B. Rood. This work was supported through NASA Grant NAS5-27266 and conducted at the Laboratory for Planetary Atmospheres at the NASA/Goddard Space Flight Center as part of the research for a doctoral dissertation at the University of Miami. Thanks are due to the Laboratory for providing the necessary facilities. We further thank the NASA/Goddard Space Flight Center, Information Processing Division, Mission Support Branch, Computing Facility, for allowing us to use their Univac 1180/82 computer.

APPENDIX

The Square-Root Scheme for General Coordinates

If the density appears explicitly in the fluid continuity equation, Eq. (13) has to be replaced by

$$\rho_s^{n+1}\chi_s^{n+1} = \rho_s^n\chi_s^n - \Delta t\delta[(\rho v)^n, \chi^n]_s + \tfrac{1}{2}\Delta t^2\delta[(\rho v)^n, \tau^n/\rho^n]_s, \quad \text{(A1)}$$

where

$$\tau_s^n = \delta[(\rho v)^n, \chi^n]$$

is again the second term on the right-hand side of (A1).

Since we will have to write the last term frequently, we abbreviate

$$\sigma_s^n = \delta[(\rho\mathbf{v})^n, \tau^n/\rho^n]_s.$$

The finite-difference operator δ is the same as (18), except that the velocities, not the concentrations, are multiplied by the density. It can easily be proven that this operator fulfills the equation

$$\alpha_s\delta(\rho\mathbf{v}, \beta)_s + \beta_s\delta(\rho\mathbf{v}, \alpha)_s = \Gamma + \alpha_s\beta_s D(\rho\mathbf{v}). \quad \text{(A2)}$$

Here

$$D(\rho\mathbf{v}) = \delta(\overline{\rho u}^x) + \delta(\overline{\delta v}^y) + \delta(\overline{\rho w}^z) \quad \text{(A3)}$$

is the second-order space derivative part of the continuity equation, and δ and $\overline{\ }$ are the conventional differencing and averaging operators, such that, for example,

$$\delta(\overline{\rho u}^x)_i = \frac{1}{2\Delta x}(\rho_{i+1}u_{i+1} - \rho_{i-1}u_{i-1}).$$

Only essential indices are written. In Eq. (A2) Γ is a total derivative which will not be written out in detail here. If the densities do not vary with time and the finite difference form (A3) is used for the continuity equation, (A1) implies Eq. (15), and all the previous arguments hold. In $\log(p)$ coordinates, for example, the last term of the fluid continuity equation (2) is replaced by

$$\frac{1}{\rho_0}\delta_z(\rho_0 w),$$

where

$$\rho_0(z) = \rho_{\text{ref}}e^{-z/H}$$

is the basic state density, H is the scale height, and $z = -H\log(p/p_0)$ the vertical coordinate. The vertical derivative in the flux form of the transport equation (4) is in these coordinates

$$\frac{1}{\rho_0}\delta_z(\rho_0 w\chi),$$

and mass conservation is achieved by writing for the vertical part of δ:

$$\frac{1}{4\Delta z}\{a[(\rho^-\omega_{k+1} + w_k)(\chi_{k+1} + \chi_k) - (w_k + \rho^+\omega_{k-1})(\chi_k + \chi_{k-1})] + b[\rho^-\omega_{k+1}(\chi_{k+2} + \chi_k) - \rho^+\omega_{k-1}(\chi_k + \chi_{k-2})]\}, \quad \text{(A4)}$$

where

$$\rho^{+/-} \equiv e^{+/-(\Delta z/2H)}.$$

For the general case where the density varies with time, and/or a finite difference expression different from (17) or (A3), respectively, has been used to obtain the velocity components, we define $\tilde{D}$ by

$$\rho_s^{n+1} \equiv \rho_s^n - \Delta t\tilde{D}. \quad \text{(A5)}$$

Then it follows from (A1) that

$$\sum_s \rho_s^{n+1}(\chi_s^{n+1})^2 = \sum_s \Big(\rho_s^n(\chi_s^n)^2 + \Delta t(\chi_s^n)^2[(\rho_s^n/\rho_s^{n+1})\tilde{D} - D(\rho\mathbf{v})] + \Delta t^2\chi_s^n(\tau_s^n/\rho_s^n)[D(\rho\mathbf{v}) - 2(\rho_s^n/\rho_s^{n+1})\tilde{D}] + \Delta t^3\Big\{[(\tau_s^n)^2/(\rho_s^n)^2]\Big[(\rho_s^n/\rho_s^{n+1})\tilde{D} - \frac{1}{2}D(\rho\mathbf{v})\Big] + \chi_s^n(\sigma_s^n/\rho_s^{n+1})\tilde{D}\Big\} + \Delta t^4\{-(\tau_s^n\sigma_s^n/\rho_s^{n+1})\tilde{D} + [(\sigma_s^n)^2/4\rho_s^{n+1}]\}\Big). \quad \text{(A6)}$$

REFERENCES

Book, D. L., J. P. Boris and K. Hain, 1975: Flux corrected transport II. Generalizations of the method. *J. Comput. Phys.*, **18**, 248–283.

Boris, J. P., and D. L. Book, 1973: Flux corrected transport I. SHASTA, A fluid transport algorithm that works. *J. Comput. Phys.*, **11**, 38–69.

Chock, D. P., and A. M. Dunker, 1983: A comparison of numerical methods for solving the advection equation. *Atmos. Environ.*, **17**, 11–24.

Hain, K., 1978: The partial donor cell method. *NRL Memo. Rep. No. 3713*, Naval Research Laboratory, Washington, DC, 18 pp.

Haltiner, G. J., and R. T. Williams, 1980: *Numerical Prediction and Dynamic Meteorology*, 2nd ed., Wiley, 477 pp.

Mahlman, J. D., 1973: Preliminary results from a three-dimensional, general-circulation tracer model. *Proc. Second Conf. Climatic Impact Assessment Program*, A. J. Broderic, Ed., U.S. Department of Transportation, Transportation Systems Center, (DOT-TSC-OST-73-4) 321–337.

——, and R. W. Sinclair, 1977: Tests of various numerical algorithms applied to a simple trace constituent air transport problem. *Fate of Pollutants in the Air and Water Environments*, I. H. Suffet, Ed., Wiley, 223–252.

Orszag, S. A., 1971: Numerical simulation of incompressible flows within simple boundaries: Accuracy. *J. Fluid Mech.*, **49**, 75–112.

Russel, G. L., and J. A. Lerner, 1981: A new finite differencing scheme for the tracer transport equation. *J. Appl. Meteor.*, **20**, 1483–1498.

Smolarkiewicz, P. K., 1983: A simple positive definite advection scheme with small implicit diffusion. *Mon. Wea. Rev.*, **111**, 479–486.

Wright, J. P., 1972: Positive conservative second and higher order difference schemes for the equations of fluid dynamics. *Proc. Third Int. Conf. on Numerical Methods in Fluid Mechanics, Lecture Notes in Physics*, Vol. 18, J. Ehlers *et al.*, Eds., Springer-Verlag, 169–173.

Zalesak, S. T., 1979: Fully multidimensional flux corrected transport algorithms for fluids. *J. Comput. Phys.*, **31**, 335–362.

——, 1981a: High order "ZIP" differencing for convective terms. *J. Comput. Phys.*, **40**, 497–508.

——, 1981b: Very high order and pseudospectral flux corrected transport (FTC) algorithms for conservation laws. *Advances in Computer Methods for Partial Differential Equations*, Vol. 4, R. Vichnevetsky and R. S. Stepleman, Eds., International Association for Mathematics and Computers in Simulation, Rutgers University, New Brunswick, NJ, 126–134.