

Boundary Conditions for Semi-Lagrangian Schemes: Testing Some Alternatives in One-Dimensional Models

A. McDONALD

Met Éireann, Dublin, Ireland

(Manuscript received 24 January 2000, in final form 27 April 2000)

ABSTRACT

The one-dimensional advection equation and the one-dimensional advection adjustment equation with rotation are used as test beds for the design of well-posed boundary conditions for the initial-boundary-value problem using semi-Lagrangian discretization. Three options are found to be stable in experimental tests: trajectory truncation, time interpolation, and a well-posed buffer zone. Stability is proved for all three for the one-dimensional advection equation when linear interpolation is used for the interpolation associated with the semi-Lagrangian discretization.

1. Introduction

In a limited area model the lateral borders are not physical boundaries to the flow but, instead, artificial constructs imposed by our desire to model what is happening in a subvolume of the global atmosphere. In order to solve the equations describing the evolution of the atmosphere in that subvolume we must supply values for the fields at these artificial boundaries. Doing this correctly is a rather subtle matter. As Olliger and Sundström (1978) have shown, for each inward pointing characteristic a field must be supplied at the lateral boundary. This implies that only a certain subset of all the variables should be imposed. If this subset has been correctly chosen the initial-boundary-value problem is “well posed.”

Constructing a well-posed system for the primitive meteorological equations is a daunting task. For that reason, and because an easy-to-implement alternative solution to the boundary problem exists, numerical weather prediction models with well-posed lateral boundaries have never gotten farther than the “drawing board” stage of existence; see, for example, Elvius and Sundström (1973). The above-mentioned alternative is to overspecify the boundaries and damp the resulting noise with a relaxation scheme; see Davies (1976). This gives stable forecasts and is almost trivial to implement. Evidence of its popularity is its almost universal use in operational limited area numerical weather prediction models. It is also used in many research models.

Despite this there are a number of reasons for taking a fresh look at well-posed boundaries. (i) There is evidence that the flow relaxation scheme can cause mass loss or gain; see McDonald (1998). (ii) At points at which the characteristic velocity is pointing out of the domain of integration we are imposing the external model solution when we do not need to if we use the flow relaxation scheme. Thus we may introduce *unnecessary* errors when these fields are incorrect (as they will be in an operational environment). (iii) Consider the situation where both the “host model” (the model supplying the boundary fields for the limited area “guest model”) and guest model fields are each in geostrophic balance. Then, as Cats and Åkesson (1983) demonstrate in their appendix the relaxation scheme can destroy this balance throughout the boundary relaxation zone. (iv) In the future 4DVAR (Le Dimet and Talagrand 1986) will be used in the forecast–analysis cycle. This has implications for our boundary treatment. Because of the iterative nature of 4DVAR the influence of the boundary conditions spreads farther into the limited area domain with each iteration. When the adjoint model is integrated backward in time, what were formerly boundary points with incoming characteristic velocities may become points describing error gradient fields with outgoing characteristic velocities. These must now pass out of the limited area without reflection. If our boundary treatment is only partially successful in doing this then one may speculate that this error could build on itself as we continue the forecast–analysis cycle. See Gustafsson et al. (1998). The Davies relaxation scheme may not be sufficiently subtle to accommodate these requirements. (v) In the past the operational limited area philosophy could reasonably be described as follows. The bound-

Corresponding author address: Aidan McDonald, Met Éireann, Glasnevin Hill, Dublin 9, Ireland.
E-mail: Aidan.McDonald@met.ie

aries were placed far enough away from the region of interest that the boundary errors would not have time to corrupt the forecast. The source of these boundary errors was twofold: (a) the boundary formulation, and (b) the inaccuracy of the host model fields (the discretization mesh was coarse, the fields were “old,” they were available only every 6 h, possibly). For the future, with increasing computing power, a new scenario can be envisaged. The host model fields will be sufficiently accurate to describe the broad-scale features of the atmosphere accurately (highs, lows, fronts). The purpose of the operational limited area model will be to forecast the fine details of the meteorological phenomena associated with these broad-scale features (rainfall, surface temperatures and winds, fog, cloud information, pollution dispersion, etc.). In a nesting situation, for example, the finest mesh area could be so small that a front can cross the area during the forecast period. Thus in the later stages the forecast will be dominated by the boundary conditions. Now, if there are errors in our boundary formulation they *will* corrupt the forecast.

For these reasons the High Resolution Limited Area Model Group has set itself the goal of building a numerical weather prediction model that has well-posed boundaries. This paper takes a first step in that direction by looking at the problems associated with the “multiply upstream well-posedness” problem. That is, how do we invent well-posed boundary conditions when using semi-Lagrangian semi-implicit discretizations? This question is addressed for two one-dimensional systems: the advection equation, and the advection–adjustment equation. Three possible answers are examined: trajectory truncation, time interpolation, and a well-posed buffer zone. Examining the one-dimensional advection equation (see section 2) allows us to ask the following question: how do we maintain stability when the departure point is outside the integration area, in isolation from the complications introduced by the C grid and the semi-implicit scheme? These latter complications are considered in section 3. In sections 2 and 3 we demonstrate stability experimentally. To complement this, in section 4 we make a first attempt at establishing stability theoretically. We do so by applying the energy method to the one-dimensional advection equation for our three postulated solutions for the multiply upstream well-posedness problem. Last, section 5 contains a discussion of the strengths and weaknesses of these different approaches to the boundary value problem.

Naïve extrapolations from the interior to the boundary are risky; see section 6 of Gustafsson et al. (1972). Therefore, as a design strategy, they are introduced only when unavoidable in what follows.

2. The one-dimensional advection equation

The one-dimensional advection equation is ideal for looking at the semi-Lagrangian discretization in isolation. Well-posedness is transparent for the analytical

system, and complications introduced by the C grid, the semi-implicit scheme, and higher spatial dimensions are absent. Therefore let us start with it:

$$\frac{\partial \phi(x, t)}{\partial t} + \bar{u} \frac{\partial \phi(x, t)}{\partial x} = 0, \tag{2.1}$$

where $0 \leq x \leq L, t \geq 0$, and \bar{u} is a constant velocity. An initial condition is needed: $\phi(x, 0) = f(x)$. Also, a boundary condition is needed: $\phi(0, t) = g(t)$, if $\bar{u} > 0$. If $\bar{u} < 0$, the boundary condition must be supplied at L .

a. Discretization of the equations

If the solution enters through the boundary at $x = 0$ and exits through the boundary at $x = L$ without distortion or reflection, the *discretization* is well posed. For that discretization let us use the semi-Lagrangian scheme. The solution to Eq. (2.1) is

$$\phi_i^{n+1} = \phi(i\Delta x - \bar{u}\Delta t, n\Delta t) = \phi_*^n, \tag{2.2}$$

where $\phi_i^n = \phi(i\Delta x, n\Delta t)$. The arrival point is at $i\Delta x$ and the departure point, x_* , is at $i\Delta x - \bar{u}\Delta t$. With a Lagrange cubic interpolation,

$$\begin{aligned} \phi_*^n = & -\frac{1}{6}\hat{\alpha}(1 - \hat{\alpha})(2 - \hat{\alpha})\phi_{i-p+1}^n \\ & + \frac{1}{2}(1 - \hat{\alpha})(1 + \hat{\alpha})(2 - \hat{\alpha})\phi_{i-p}^n \\ & + \frac{1}{2}\hat{\alpha}(1 + \hat{\alpha})(2 - \hat{\alpha})\phi_{i-p-1}^n \\ & - \frac{1}{6}\hat{\alpha}(1 - \hat{\alpha})(1 + \hat{\alpha})\phi_{i-p-2}^n, \end{aligned} \tag{2.3}$$

where $\alpha = \bar{u}\Delta t/\Delta x; p = [\alpha]$ if $\alpha \geq 0$ and $p = [\alpha - 1]$ if $\alpha < 0$; and $[\alpha]$ is the integer part of $\alpha; p + \hat{\alpha} = \alpha$. To maintain stability p must be chosen such that $(i - p - 1)\Delta x \leq x_* \leq (i - p)\Delta x$ or equivalently $0 \leq \hat{\alpha} \leq 1$. See Bates and McDonald (1982) for more details.

If the departure point is outside the area we cannot use Eq. (2.3). We examine various methods for addressing this problem below. Nor can we use Eq. (2.3) when the departure point lies in the grid box immediately next to the boundary. In that case let us use a quadratic Lagrange interpolation:

$$\begin{aligned} \phi_*^n = & -\frac{1}{2}\hat{\alpha}(1 - \hat{\alpha})\phi_{i-p+1}^n + (1 + \hat{\alpha})(1 - \hat{\alpha})\phi_{i-p}^n \\ & + \frac{1}{2}(1 + \hat{\alpha})\hat{\alpha}\phi_{i-p-1}^n \end{aligned} \tag{2.4a}$$

when $0 \leq x_* < \Delta x$ and

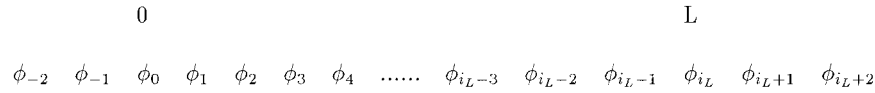


FIG. 1. Distribution of variables on the x axis.

$$\begin{aligned} \phi_*^n &= \frac{1}{2}(1 - \hat{\alpha})(2 - \hat{\alpha})\phi_{i-p}^n + \hat{\alpha}(2 - \hat{\alpha})\phi_{i-p-1}^n \\ &\quad - \frac{1}{2}\hat{\alpha}(1 - \hat{\alpha})\phi_{i-p-2}^n, \end{aligned} \tag{2.4b}$$

when $L - \Delta x \leq x_* < L$.

Let us consider the outflow boundary first. Recall that for well posedness we *must not impose* a value of ϕ at this boundary; we must extrapolate from the interior. The semi-Lagrangian scheme provides us with an ideal way of doing this; see Robert and Yakimiw (1986). When $|\alpha| < 1$, then we use Eq. (2.4). It yields an $O(\Delta x^3/\Delta t)$ accurate solution. If $|\alpha| \geq 1$, then we use Eq. (2.3), which is $O(\Delta x^4/\Delta t)$ accurate.

It is interesting to note that if, for example, we were using the leapfrog Eulerian scheme to integrate Eq. (2.1) we would have to switch to either Eq. (2.4) or some other upwind type of discretization in order to compute ϕ at the outflow boundary point. See McDonald (1999a), for example.

Next consider the inflow boundary and points in its immediate vicinity. On the boundary line itself we must *impose* a value on the field. If $|\alpha| > 1$, then updating ϕ at the grid points adjacent to the inflow boundary causes difficulties, which we examine in sections 2d–2f. If $|\alpha| \leq 1$, then $p = 0$, and we proceed as in section 2c.

b. Setup for the numerical tests

To test this let us model a bell curve of “width” $L/10$ being advected in the positive direction along the x axis

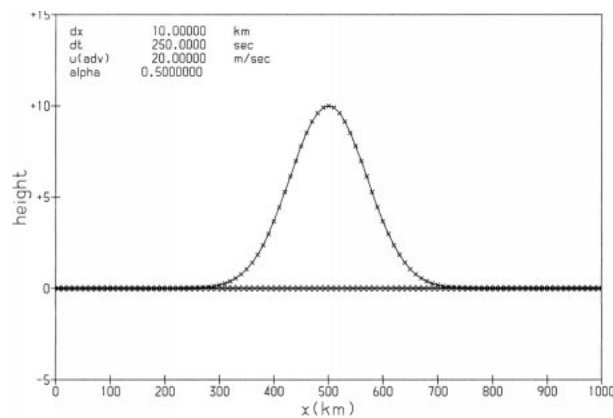


FIG. 2. The solution for test 1. At time zero the bell curve, centered at 500 km, is shown by the \times 's. The analytical solution, for which the bell curve is centered at 1500 km, is shown by the diamonds. The result of integrating with $\alpha = 0.5$ is shown by the dots.

with a velocity \bar{u} , starting with the maximum of the bell curve at $x = x_s$. At any subsequent time t the position of that maximum will be at $x(t) = x_s + \bar{u}t$. The initial condition is

$$\phi(x, 0) = \hat{\phi} \exp\{ -[(x - x_s)/(L/10)]^2 \}. \tag{2.5}$$

Consider $\bar{u} > 0$ without loss of generality. Then we must supply a boundary condition at $x = 0$:

$$\begin{aligned} \phi(0, t) &= \hat{\phi} \exp\{ -[(0 - x_s - \bar{u}t)/(L/10)]^2 \}; \\ t &\geq 0. \end{aligned} \tag{2.6}$$

The following parameters will be used in the tests of our discretization: $\Delta x = 10$ km, $\bar{u} = 20$ m s⁻¹, $L = 1000$ km, and $\hat{\phi} = 10$. We will perform the following two experiments.

Test 1: start with the bell curve centered at $x = L/2$ and choose the time such that at the end of the integration it will be centered at $3L/2$ in order to see whether it passes through the boundary at $x = L$ without false reflections.

Test 2: start with the bell curve centered at $x = -L/2$ and choose the time such that at the end of the integration it will be centered at $L/2$ in order to see whether it passes through the boundary at $x = 0$ without distortion.

c. Testing with $|\alpha| < 1$

Assuming $\bar{u} > 0$, we impose ϕ_0^{n+1} . We compute ϕ_1^{n+1} using Eq. (2.4a). Over the range $i = 2 \rightarrow i_L - 1$, we use Eq. (2.3), and for $\phi_{i_L}^{n+1}$ we use Eq. (2.4b) ($x_{i_L} = L$; see Fig. 1).

With $\alpha = 0.5$ it passes both tests. Figure 2 shows the results of test 1. The bell curve has exited through the boundary accurately. There are no false reflections (the dots coincide exactly with the diamonds). Figure 3 shows the results of test 2. The bell curve has entered the area accurately (the dots coincide almost exactly with the diamonds).

Let us return to the problem near the inflow boundary when $|\alpha| > 1$. We now have $|p| > 0$, and when, for example, $\bar{u} > 0$, the departure point associated with ϕ_1 is outside the area. What are the options?

d. Option 1, when $|\alpha| > 1$: Trajectory truncation

If the departure point is beyond the boundary, then truncate the trajectory at the boundary. Thus, for example, if the departure point associated with ϕ_2 were at $x = -\Delta x/2$ we would use for Eq. (2.2)

$$\phi_2^{n+1} = \phi^n(-\Delta x/2, n\Delta t) \approx \phi^n(0, n\Delta t). \tag{2.7}$$

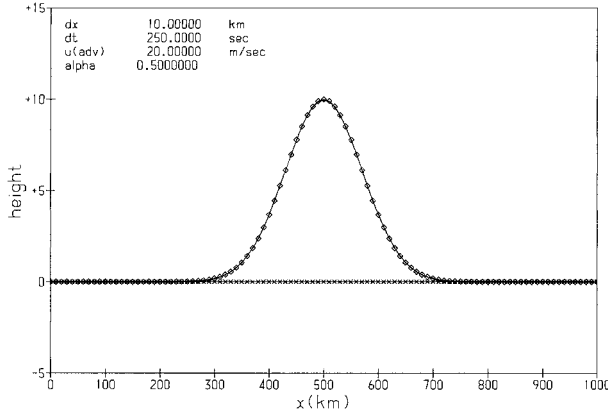


FIG. 3. The solution for test 2. At time zero the bell curve, centered at -500 km, is shown by the \times 's. The analytical solution, for which the bell curve is centered at 500 km, is shown by the diamonds. The result of integrating with $\alpha = 0.5$ is shown by the dots.

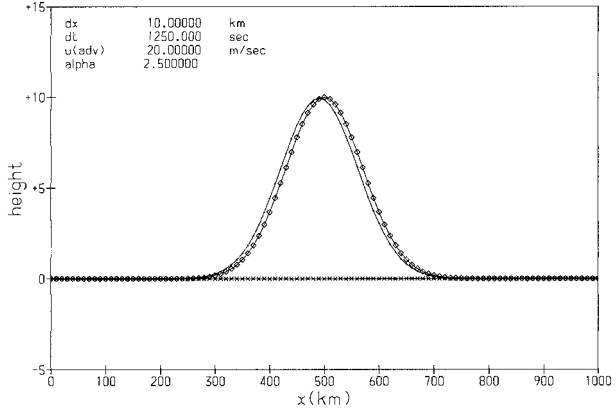


FIG. 4. Same as Fig. 3 but using the trajectory truncation scheme (option 1) and $\alpha = 2.5$.

This method is stable but slightly inaccurate, as can be seen from Fig. 4, which shows the result of repeating test 2 with $\alpha = 2.5$

e. Option 2, when $|\alpha| > 1$: Time interpolation

Let us try to improve the accuracy of the trajectory truncation method. Recall that at inflow we know the field value at every time step on the boundary line, in principle, anyway. Let us use this information by invoking an idea of Skålin and Lie (1995). See also Falcone and Ferretti (1998). Consider the following argument. The solution to Eq. (2.1) can be written as

$$\phi(x_i, t + \delta t) = \phi(x_i - \bar{u}\delta t, t). \quad (2.8)$$

Choose $\bar{u}\delta t = \mu\Delta x$, where μ is a nonzero integer. Also choose $t + \delta t = (n + 1)\Delta t$. Substituting in Eq. (2.8) we get

$$\phi[x_i, (n + 1)\Delta t] = \phi\left[x_{i-\mu}, \left(n + 1 - \frac{\mu}{\alpha}\right)\Delta t\right]. \quad (2.9)$$

For the following argument consider $\bar{u} > 0$. There is no loss of generality. Let $i = \mu$; then this equation expresses the value of the field at time $(n + 1)\Delta t$ at the interior grid point, μ , in terms of the field on the boundary line, $i = 0$, at time $(n + 1 - \mu/\alpha)\Delta t$. The field is not known at this time. However, we can “interpolate in time” because, in principle, on the boundary line we know $\phi(x_0, m\Delta t)$ for all integer $m > 0$. For example, using a linear Lagrange interpolation for the right-hand side of Eq. (2.9) yields

$$\phi_\mu^{n+1} = \left(1 - \frac{\mu}{\alpha}\right)\phi_0^{n+1} + \frac{\mu}{\alpha}\phi_0^n. \quad (2.10)$$

This is stable from a von Neumann point of view, anyway, because $0 < \mu/\alpha < 1$. When $|\alpha| < \mu$, we do not use this approach to finding the departure point value

of the field; we use Eq. (2.3) or (2.4). Equation (2.9) is only used when $i < [\alpha]$, where $[\alpha]$ is the integer part of α . Thus, in Eq. (2.10), where we have chosen $i = \mu$, $\mu/\alpha \leq 1$.

If, instead, we have inflow at the $x = L$ boundary we choose $i = \mu + i_L$. Both μ and α are now negative but the arguments are the same and instead of Eq. (2.10) we use

$$\phi_{\mu+i_L}^{n+1} = \left(1 - \frac{\mu}{\alpha}\right)\phi_{i_L}^{n+1} + \frac{\mu}{\alpha}\phi_{i_L}^n. \quad (2.11)$$

Obviously, $O(\Delta t^2)$ accuracy can be achieved by using a Lagrange quadratic interpolation,

$$\phi^{n+1-(\mu/\alpha)} = C_+\left(\frac{\mu}{\alpha}\right)\phi^{n+1} + C_0\left(\frac{\mu}{\alpha}\right)\phi^n + C_-\left(\frac{\mu}{\alpha}\right)\phi^{n-1}, \quad (2.12)$$

where

$$C_+\left(\frac{\mu}{\alpha}\right) = \frac{1}{2}\left(1 - \frac{\mu}{\alpha}\right)\left(2 - \frac{\mu}{\alpha}\right), \quad C_0\left(\frac{\mu}{\alpha}\right) = \frac{\mu}{\alpha}\left(2 - \frac{\mu}{\alpha}\right), \\ C_-\left(\frac{\mu}{\alpha}\right) = -\frac{1}{2}\frac{\mu}{\alpha}\left(1 - \frac{\mu}{\alpha}\right). \quad (2.13)$$

Repeating test 2 with $\alpha = 2.5$, using quadratic interpolation in time at the inflow boundary, we see from Fig. 5 that full accuracy is restored.

f. Option 3, when $|\alpha| > 1$: Well-posed buffer zone

From a semi-Lagrangian point of view it would be ideal if we had a buffer zone external to the boundary. Then if the departure point were located there, we could interpolate in the same way as if it were in the interior. In this section we generate such a buffer zone, which gives stable integrations by using only information from the inflow boundary line. Again, for the purpose of dem-

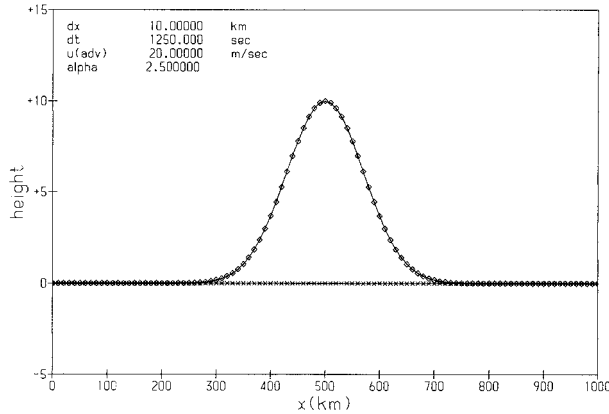


FIG. 5. Same as Fig. 3 but using time interpolation (option 2) and $\alpha = 2.5$.

onstrator consider the situation when $\bar{u} > 0$. Then we wish to compute ϕ_{-1} , ϕ_{-2} , etc., in Fig. 1.

Naïve extrapolation from the interior to the inflow buffer zone will cause instability. Thus we cannot, for example, use Eq. (2.3) to extrapolate, abandoning the restriction $0 < \hat{\alpha} \leq 1$. [A von Neumann stability analysis shows that for $\hat{\alpha} = -2$ the amplification factor is 15 for the two-grid wave; see Eq. (23) of McDonald (1984).]

Since for well-posedness we must supply $\phi(0, t)$ for all t we know $\partial\phi(0, t)/\partial t$, $\partial^2\phi(0, t)/\partial t^2$, and so on. Thus, from Eq. (2.1) we know

$$\begin{aligned} \frac{\partial\phi(0, t)}{\partial x} &= -\frac{1}{\bar{u}} \frac{\partial\phi(0, t)}{\partial t}, \\ \frac{\partial^2\phi(0, t)}{\partial x^2} &= \frac{1}{\bar{u}^2} \frac{\partial^2\phi(0, t)}{\partial t^2}, \end{aligned} \quad (2.14)$$

and so on, enabling us to estimate ϕ_{-1} , ϕ_{-2} , etc., using a Taylor expansion and substituting from Eq. (2.14):

$$\phi(-i\Delta x, t) = \left[\phi + i \frac{\Delta x}{\bar{u}} \frac{\partial\phi}{\partial t} + \frac{(i\Delta x)^2}{2\bar{u}^2} \frac{\partial^2\phi}{\partial t^2} + \dots \right]_0^t. \quad (2.15)$$

When $\bar{u} < 0$, the same procedure can be used to compute $\phi(L + i\Delta x)$.

Repeating test 2 with $\alpha = 2.5$, using this procedure to compute the necessary fields in the boundary zone restores full accuracy. The figure (not shown) looks identical to Fig. 5.

3. The one-dimensional advection-adjustment equation

We saw in section 2 that we could integrate the one-dimensional advection equation stably for large $|\alpha|$ by using time interpolation, a well-posed buffer zone, or, less accurately, trajectory truncation. In the real atmosphere adjustment and rotation also play significant

roles. Thus in this section we add adjustment and rotation terms to the advection equation in order to see whether these schemes can still be implemented in this more realistic environment.

a. Discretization of the equations

The equations we wish to discretize are

$$\frac{\partial\phi(x, t)}{\partial t} + \bar{u} \frac{\partial\phi(x, t)}{\partial x} + \bar{\phi} \frac{\partial u(x, t)}{\partial x} = 0, \quad (3.1)$$

$$\frac{\partial u(x, t)}{\partial t} + \bar{u} \frac{\partial u(x, t)}{\partial x} + \frac{\partial\phi(x, t)}{\partial x} - \bar{f}v(x, t) = 0, \quad (3.2)$$

$$\frac{\partial v(x, t)}{\partial t} + \bar{u} \frac{\partial v(x, t)}{\partial x} + \bar{f}u(x, t) = 0, \quad (3.3)$$

where the advection velocity \bar{u} , the Coriolis parameter \bar{f} , and the mean geopotential height $\bar{\phi}$ are constants.

By rearranging Eqs. (3.1) and (3.2) we can write $(\bar{c} = \sqrt{\bar{\phi}})$

$$\left[\frac{\partial}{\partial t} + (\bar{u} + \bar{c}) \frac{\partial}{\partial x} \right] [\phi + \bar{c}u] - \bar{c}\bar{f}v = 0, \quad (3.4)$$

$$\left[\frac{\partial}{\partial t} + (\bar{u} - \bar{c}) \frac{\partial}{\partial x} \right] [\phi - \bar{c}u] + \bar{c}\bar{f}v = 0. \quad (3.5)$$

The characteristic variables of these equations (and their associated characteristics) are

$$\begin{aligned} v, & & (\bar{u}); \\ p = \phi + \bar{c}u, & & (\bar{u} + \bar{c}); \\ q = \phi - \bar{c}u, & & (\bar{u} - \bar{c}). \end{aligned} \quad (3.6)$$

For each inward pointing characteristic a field must be imposed on the boundary; see, for example, Sundström and Elvius (1979).

Substituting $\phi = \hat{\phi} \exp[i(kx - \omega t)]$ into Eqs. (3.1)–(3.3) yields three solutions; a “slow” solution with $\omega = k\bar{u}$ and two “fast” solutions with $\omega = k(\bar{u} \pm c_k)$, where

$$c_k = \sqrt{\bar{\phi} + \bar{f}^2/k^2}. \quad (3.7)$$

Let us now perform a two-time-level semi-Lagrangian and implicit discretization on Eqs. (3.1)–(3.3) using C-grid staggering of the fields (see Fig. 6):

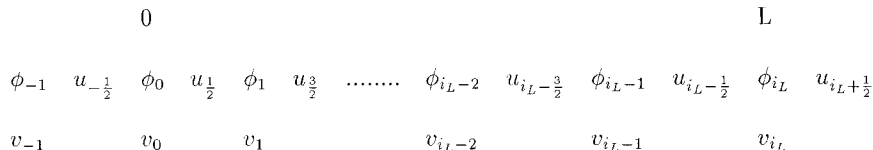


FIG. 6. Distribution of variables on the x axis using the C grid.

$$\begin{aligned} \phi_i^{n+1} + \frac{\bar{\phi}\Delta t}{2\Delta x}(u_{i+(1/2)}^{n+1} - u_{i-(1/2)}^{n+1}) \\ = R_i^\phi = (X^\phi)_{i,*}^n, \end{aligned} \tag{3.8}$$

$$\begin{aligned} u_{i+(1/2)}^{n+1} + \frac{\Delta t}{2\Delta x}(\phi_{i+1}^{n+1} - \phi_i^{n+1}) - \frac{\bar{f}\Delta t}{4}(v_{i+1}^{n+1} + v_i^{n+1}) \\ = R_{i+(1/2)}^u = (X^u)_{i+(1/2),*}^n, \end{aligned} \tag{3.9}$$

$$\begin{aligned} v_i^{n+1} + \frac{\bar{f}\Delta t}{4}(u_{i+(1/2)}^{n+1} + u_{i-(1/2)}^{n+1}) \\ = R_i^v = (X^v)_{i,*}^n, \end{aligned} \tag{3.10}$$

where

$$(X^\phi)_i^n = \phi_i^n - \frac{\bar{\phi}\Delta t}{2\Delta x}(u_{i+(1/2)}^n - u_{i-(1/2)}^n), \tag{3.11}$$

$$\begin{aligned} (X^u)_{i+(1/2)}^n = u_{i+(1/2)}^n - \frac{\Delta t}{2\Delta x}(\phi_{i+1}^n - \phi_i^n) \\ + \frac{\bar{f}\Delta t}{4}(v_{i+1}^n + v_i^n), \end{aligned} \tag{3.12}$$

$$(X^v)_i^n = v_i^n - \frac{\bar{f}\Delta t}{4}(u_{i+(1/2)}^n + u_{i-(1/2)}^n), \tag{3.13}$$

where $X_{i,*}^n$ means interpolate X to the departure point associated with the grid point i . Having computed R^ϕ , R^u , and R^v , one approach to solving this system of equations is as follows. Equations (3.8)–(3.10) can be rearranged to yield a tridiagonal system for u^{n+1} . Having solved this, ϕ^{n+1} and v^{n+1} emerge from Eqs. (3.8) and (3.10), respectively.

If we were to proceed in this way for the two-dimensional shallow water equations we would generate the doubly averaged terms $\bar{f}\bar{u}$ and $\bar{f}\bar{v}$, each of which has a nine-point template. Near the boundary these terms would further complicate the already difficult problem of inventing a stable discrete system of equations. With this in mind let us use the following iterative approach to solving these equations; its two-dimen-

sional generalization will cause minimal complications on the boundary:

$$\left[\phi_i + \frac{\bar{\phi}\Delta t}{2\Delta x}(u_{i+(1/2)} - u_{i-(1/2)}) \right]^{(k+1)} = R_i^\phi, \tag{3.14}$$

$$\begin{aligned} \left[u_{i+(1/2)} + \frac{\Delta t}{2\Delta x}(\phi_{i+1} - \phi_i) \right]^{(k+1)} \\ = R_{i+(1/2)}^u + \frac{\bar{f}\Delta t}{4}(v_{i+1} + v_i)^{(k)} \equiv \tilde{R}_{i+(1/2)}^u, \end{aligned} \tag{3.15}$$

$$v_i^{(k+1)} = R_i^v - \frac{\bar{f}\Delta t}{4}(u_{i+(1/2)} + u_{i-(1/2)})^{(k)}. \tag{3.16}$$

Having computed R^ϕ , R^u , and R^v , substituting Eq. (3.15) in Eq. (3.14) results in

$$\begin{aligned} \left[\phi_i - \bar{\phi} \left(\frac{\Delta t}{2\Delta x} \right)^2 (\phi_{i+1} - 2\phi_i + \phi_{i-1}) \right]^{(k+1)} \\ = R_i^\phi - \frac{\bar{\phi}\Delta t}{2\Delta x}(R_{i+(1/2)}^u - R_{i-(1/2)}^u) \\ - \frac{\bar{\phi}\bar{f}\Delta t^2}{8\Delta x}(v_{i+1} - v_{i-1})^{(k)}. \end{aligned} \tag{3.17}$$

Having solved this equation for ϕ with appropriate boundary conditions, $u_{i+(1/2)}^{(k+1)}$ can be computed from Eq. (3.15), and $v_i^{(k+1)}$ from Eq. (3.16). The procedure can be iterated until sufficient accuracy is attained.

The indices for which Eqs. (3.14)–(3.17) are valid depend on the grid points at which the boundaries are imposed. Examples are given in Table 1 for four different boundary choices: $u_{1/2}$, $u_{i_L+1/2}$ imposed; ϕ_0 , ϕ_{i_L} imposed; p_0 , q_{i_L} imposed; and $p_{1/2}$, $q_{i_L+1/2}$ imposed. As we shall see in section 3b, for example, it will be necessary to approximate Eq. (3.3) on the boundary line. Because Eq. (3.10) cannot be used we must replace it with an equation that uses a stable and accurate extrapolation from the interior of the required unknown fields.

TABLE 1. Range of validity the indices in Eqs. (3.14)–(3.17) for different boundary choices.

Boundary	Eq. (3.14)	Eq. (3.15)	Eq. (3.16)	Eq. (3.17)
$u_{1/2}$, $u_{i_L+1/2}$	$1 \rightarrow i_L$	$1 \rightarrow i_L - 1$	$1 \rightarrow i_L$	$2 \rightarrow i_L - 1$
ϕ_0 , ϕ_{i_L}	$1 \rightarrow i_L - 1$	$0 \rightarrow i_L - 1$	$1 \rightarrow i_L - 1$	$1 \rightarrow i_L - 1$
p_0 , q_{i_L}	$0 \rightarrow i_L$	$0 \rightarrow i_L - 1$	$0 \rightarrow i_L$	$1 \rightarrow i_L - 1$
$p_{1/2}$, $q_{i_L+1/2}$	$1 \rightarrow i_L$	$0 \rightarrow i_L$	$1 \rightarrow i_L$	$2 \rightarrow i_L - 1$

b. Trajectory truncation

To clarify the discussion let us impose our boundary values on ϕ_0 and ϕ_{i_L} and v at inflow. They fit naturally into Eq. (3.17). There is a complication: we need to know v on the outflow boundary line in order to solve Eq. (3.17). Also, as we discuss below, v at inflow is problematical. Thus, we are forced into making extrapolations from the interior.

Consider the problem of computing v at outflow. We can guess that “semi-Lagrangian extrapolation” will be stable. That is, we assume an approximation to Eq. (3.3) is obeyed at the outflow boundary point. Notice we must extrapolate unknown fields from the interior; let us use the simplest possible approximation:

$$\bar{u} < 0:$$

$$v_0^{n+1} + \frac{\bar{f}\Delta t}{4}(3u_{1/2}^{n+1} - u_{3/2}^{n+1}) = R_0^v = (X^v)_0^{n,*}, \quad (3.18)$$

$$\bar{u} > 0:$$

$$v_{i_L}^{n+1} + \frac{\bar{f}\Delta t}{4}(3u_{i_L-(1/2)}^{n+1} - u_{i_L-(3/2)}^{n+1}) = R_{i_L}^v = (X^v)_{i_L}^{n,*}. \quad (3.19)$$

When the departure point is within two grid points of the boundary, we will need $(X^v)_0^n$ or $(X^v)_{i_L}^n$ for the computation of R^v . For consistency we define them as

$$(X^v)_0^n = v_0^n - \frac{\bar{f}\Delta t}{4}(3u_{1/2}^n - u_{3/2}^n), \quad (3.20)$$

$$(X^v)_{i_L}^n = v_{i_L}^n - \frac{\bar{f}\Delta t}{4}(3u_{i_L-(1/2)}^n - u_{i_L-(3/2)}^n). \quad (3.21)$$

In fact, as is shown in McDonald (2000) by experiment, the integration is inaccurate unless we make sure to use X_0^u and $X_{i_L}^u$ at inflow as well. Thus trajectory truncation for these boundary conditions will mean using $R^u = (X^u)_{1/2}^n$, when $x_* < \Delta x/2$; $R^u = (X^u)_{i_L-1/2}^n$, when $x_* > L - \Delta x/2$; $R^\phi = (X^\phi)_1^n$, when $x_* < \Delta x$; $R^\phi = (X^\phi)_{i_L-1}^n$, when $x_* > L - \Delta x$; $R^v = (X^v)_0^n$, when $x_* < 0$; and $R^v = (X^v)_{i_L}^n$, when $x_* > L$.

Extrapolations from the interior are also necessary if we impose boundary conditions on u , p , or q . See McDonald (2000) for details.

c. Time interpolation

Is the time interpolation scheme still a feasible option when we add adjustment and rotation to the advection equation? To answer this, repeat the arguments of section 2, and discretize Eq. (3.1) as

$$\frac{\phi(x_i, t + \delta t) - \phi(x_i - \bar{u}\delta t, t)}{\delta t} + \frac{\bar{\phi}}{2} \left[\frac{\partial u}{\partial x}(x_i, t + \delta t) + \frac{\partial u}{\partial x}(x_i - \bar{u}\delta t, t) \right] = 0, \quad (3.22)$$

which becomes

$$\begin{aligned} \left[\phi + \frac{\mu}{\alpha} \frac{\bar{\phi}\Delta t}{2} \frac{\partial u}{\partial x} \right]_{b+\mu}^{n+1} &= \left[\phi - \frac{\mu}{\alpha} \frac{\bar{\phi}\Delta t}{2} \frac{\partial u}{\partial x} \right]_b^{n+1-(\mu/\alpha)} \\ &\equiv \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^{n+1-(\mu/\alpha)}. \end{aligned} \quad (3.23)$$

The same arguments lead to the following discretization for Eqs. (3.2) and (3.3):

$$\begin{aligned} \left[u + \frac{\nu}{\alpha} \frac{\Delta t}{2} \left(\frac{\partial \phi}{\partial x} - \bar{f}v \right) \right]_{b+\nu}^{n+1} &= \left[u - \frac{\nu}{\alpha} \frac{\Delta t}{2} \left(\frac{\partial \phi}{\partial x} - \bar{f}v \right) \right]_b^{n+1-(\nu/\alpha)} \\ &\equiv \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^{n+1-(\nu/\alpha)}, \end{aligned} \quad (3.24)$$

$$\begin{aligned} \left[v + \frac{\mu}{\alpha} \frac{\bar{f}\Delta t}{2} u \right]_{b+\mu}^{n+1} &= \left[v - \frac{\mu}{\alpha} \frac{\bar{f}\Delta t}{2} u \right]_b^{n+1-(\mu/\alpha)} \\ &\equiv \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^{n+1-(\mu/\alpha)}. \end{aligned} \quad (3.25)$$

Here b represents a boundary line, and μ is the integer number of points the ϕ point of interest is from the boundary. Also, ν is a half odd-integer, which measures the distance of the u point of interest from the boundary.

Assume $\bar{c} > \bar{u}$. Then well-posedness implies that we impose two fields on the inflow boundary and one on the outflow boundary. To keep the arguments clear let us assume we are imposing ϕ and v at inflow and ϕ at outflow. Then as can be seen, most of the fields required to compute \hat{X}^ϕ , \hat{X}^u , and \hat{X}^v are not known on the boundary, even at time level n . How can we proceed?

The terms u , $\partial u/\partial x$, and $\partial \phi/\partial x$ are not known on the boundary and, thus, must be extrapolated. The simplest possible extrapolations give, at $x = 0$,

$$\begin{aligned} u_0 &= \frac{3}{2}u_{1/2} - \frac{1}{2}u_{3/2}; & \left(\frac{\partial u}{\partial x} \right)_0 &= \frac{1}{\Delta x}(u_{3/2} - u_{1/2}); \\ \left(\frac{\partial \phi}{\partial x} \right)_0 &= \frac{1}{\Delta x}(\phi_1 - \phi_0), \end{aligned} \quad (3.26)$$

and at $x = L$,

$$\begin{aligned} u_{i_L} &= \frac{3}{2}u_{i_L-(1/2)} - \frac{1}{2}u_{i_L-(3/2)}; \\ \left(\frac{\partial u}{\partial x} \right)_{i_L} &= \frac{1}{\Delta x}(u_{i_L-(1/2)} - u_{i_L-(3/2)}); \\ \left(\frac{\partial \phi}{\partial x} \right)_{i_L} &= \frac{1}{\Delta x}(\phi_{i_L} - \phi_{i_L-1}). \end{aligned} \quad (3.27)$$

Let us assume for now that these are stable. We will test them in sections 3e and 3f.

How do we solve Eqs. (3.23)–(3.25)? We have a number of choices. (i) Use \hat{X}^n and \hat{X}^{n-1} to extrapolate to time $\hat{X}^{n+1-\mu/\alpha}$ ($\hat{X}^{n+1-\nu/\alpha}$). I have not been able to invent a stable method for doing this. (ii) Incorporate the time interpolation into the semi-implicit scheme. That is, use Eq. (2.12) to rewrite Eqs. (3.23)–(3.25), resulting in the following:

$$\left[\phi + \frac{\bar{\phi} \mu \Delta t}{2\alpha} \frac{\partial u}{\partial x} \right]_{b+\mu}^{n+1} - C_+ \left(\frac{\mu}{\alpha} \right) \left[\phi - \frac{\bar{\phi} \mu \Delta t}{2\alpha} \frac{\partial u}{\partial x} \right]_b^{n+1} = C_0 \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^{n-1}; \quad (3.28)$$

$$\left[u + \frac{\nu \Delta t}{2\alpha} \left(\frac{\partial \phi}{\partial x} - \bar{f} \nu \right) \right]_{b+\nu}^{n+1} - C_+ \left(\frac{\nu}{\alpha} \right) \left[u - \frac{\nu \Delta t}{2\alpha} \left(\frac{\partial \phi}{\partial x} - \bar{f} \nu \right) \right]_b^{n+1} = C_0 \left(\frac{\nu}{\alpha} \right) \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\nu}{\alpha} \right) \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^{n-1}; \quad (3.29)$$

$$\left[v + \frac{\bar{f} \mu \Delta t}{2\alpha} u \right]_{b+\mu}^{n+1} - C_+ \left(\frac{\mu}{\alpha} \right) \left[v - \frac{\bar{f} \mu \Delta t}{2\alpha} u \right]_b^{n+1} = C_0 \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^{n-1}. \quad (3.30)$$

Even using the simplest of extrapolations given in Eqs. (3.26) and (3.27) the discretization of Eqs. (3.28)–(3.30) leads to a set of equations that are unattractive from the point of view of programming efficiency. The complications are as follows. (a) Both the left- and right-hand sides of the discretized versions Eqs. (3.28)–(3.30) can formally differ for each half-integer increment of α . As is shown in the appendix of McDonald (1999b) there are three alternative equations for $u_{1/2}$, depending on the size of α . If we use the next highest order of approximation for u_0 , there will be five. (The programming complication being that we need “if” statements or their equivalents.) (b) The number of floating point operations increases significantly. (c) Because the semi-Lagrangian and semi-implicit systems have become intertwined we no longer have a tridiagonal system of equations to solve.

From a computing point of view these drawbacks may not be too serious in a single-processor scalar environment. We would simply pay a possibly moderate computational price for getting the boundaries right. However, with vector processors in a multiprocessor environment these complications seem very serious. For these reasons it is interesting to consider an alternative that avoids these pitfalls, that of iterating.

(iii) Move the troublesome terms in Eqs. (3.28)–(3.30) to the right-hand side and compute them iteratively:

$$\left[\phi + \frac{\bar{\phi} \mu \Delta t}{2\alpha} \frac{\partial u}{\partial x} \right]_{b+\mu}^{(k+1)} = C_+ \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^{(k)} + C_0 \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^\phi \left(\frac{\mu}{\alpha} \right) \right]_b^{n-1}, \quad (3.31)$$

$$\left[u + \frac{\nu \Delta t}{2\alpha} \frac{\partial \phi}{\partial x} \right]_{b+\nu}^{(k+1)} = \frac{\bar{f} \nu \Delta t}{2\alpha} v_{b+\nu}^{(k)} + C_+ \left(\frac{\nu}{\alpha} \right) \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^{(k)} + C_0 \left(\frac{\nu}{\alpha} \right) \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\nu}{\alpha} \right) \left[\hat{X}^u \left(\frac{\nu}{\alpha} \right) \right]_b^{n-1}, \quad (3.32)$$

$$v_{b+\mu}^{(k+1)} = -\frac{\bar{f} \mu \Delta t}{2\alpha} u_{b+\mu}^{(k)} + C_+ \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^{(k)} + C_0 \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^n + C_- \left(\frac{\mu}{\alpha} \right) \left[\hat{X}^v \left(\frac{\mu}{\alpha} \right) \right]_b^{n-1}. \quad (3.33)$$

When discretized, these equations give rise to a set that is formally the same as Eqs. (3.14)–(3.16) and is thus very easy to program. (The tridiagonal system now has variable coefficients and there are a few extra terms on the right side to be evaluated during each iteration.)

d. Well-posed buffer zone

How can we fill a well-posed buffer zone when some of the fields needed for the extrapolation cannot be imposed externally? For discussion purposes assume $\bar{c} > \bar{u}$. Then only two fields at inflow will be imposed externally. In order to compute fields for a buffer zone at inflow we need to know the third field, which to maintain well-posedness we must not impose externally.

How can we estimate it stably using only fields from the interior?

Gustafsson et al. (1972) showed that, for the Lax–Wendroff scheme, many extrapolations cause instability on $0 \leq x \leq L; t > 0$. We can infer that such extrapolations will also be unstable for the semi-Lagrangian schemes since the Lax–Wendroff scheme corresponds to the latter with quadratic interpolation when $|\alpha| < 1$. A scheme that they proved to be stable uses Eqs. (3.3)–(3.5) to extrapolate the field corresponding to the outgoing characteristic to the boundary. Let us put a semi-Lagrangian twist to this idea.

For $\bar{u} > 0$ we need a buffer zone for the region $x < 0$. Thus we extrapolate q_0 from the interior using Eq. (3.5). For $\bar{u} < 0$ we need a buffer zone for the region

$x > L$. Thus we extrapolate p_{i_L} from the interior using Eq. (3.4). To maintain stability we solve Eqs. (3.5) and (3.4) by finding the value of the field at the departure points $x_{*q} = x_0 - (\bar{u} - \bar{c})\Delta t$ and $x_{*p} = x_{i_L} - (\bar{u} + \bar{c})\Delta t$, respectively:

$$q_0^{n+1} = \left(q - \frac{\Delta t \bar{c}}{2} f \bar{c} v \right)_{*q}^n - \frac{\Delta t \bar{c}}{2} f \bar{c} v_0^{n+1}, \quad (3.34)$$

$$p_{i_L}^{n+1} = \left(p + \frac{\Delta t \bar{c}}{2} f \bar{c} v \right)_{*p}^n + \frac{\Delta t \bar{c}}{2} f \bar{c} v_{i_L}^{n+1}. \quad (3.35)$$

Say we are imposing ϕ at both boundaries and v at inflow. For $\bar{u} > 0$, v_0^{n+1} is imposed. Thus we know q^{n+1} at x_0 from Eq. (3.34), meaning we can approximate $(\partial q/\partial t)^n$ and $(\partial^2 q/\partial t^2)^n$ there. We can also estimate $(\partial p/\partial t)^n$ and $(\partial^2 p/\partial t^2)^n$ there, using $p^{n+1} = 2\phi^{n+1} - q^{n+1}$. In the same way, for $\bar{u} < 0$, $v_{i_L}^{n+1}$ is imposed, and we can approximate $(\partial p/\partial t)^n$ and $(\partial^2 p/\partial t^2)^n$ there, using Eq. (3.35). We can also estimate $(\partial q/\partial t)^n$ and $(\partial^2 q/\partial t^2)^n$ there, using $q^{n+1} = 2\phi^{n+1} - p^{n+1}$. Thus we can use Eqs. (3.4) and (3.5) to compute $\partial p/\partial x$ and $\partial q/\partial x$:

$$\frac{\partial p}{\partial x} = \frac{1}{\bar{u} + \bar{c}} \left(-\frac{\partial p}{\partial t} + \bar{f} \bar{c} v \right), \quad (3.36)$$

$$\frac{\partial q}{\partial x} = \frac{1}{\bar{u} - \bar{c}} \left(-\frac{\partial q}{\partial t} - \bar{f} \bar{c} v \right), \quad (3.37)$$

$$\frac{\partial v}{\partial x} = \frac{1}{\bar{u}} \left[-\frac{\partial v}{\partial t} - \frac{\bar{f}}{2\bar{c}} (p - q) \right]. \quad (3.38)$$

From these we can deduce $\partial \phi/\partial x$ and $\partial u/\partial x$ using the definitions of p and q given in Eq. (3.6). Differentiating Eqs. (3.36)–(3.38) with respect to x we can compute the second order terms. Then we use the Taylor expansion to compute ϕ , u , and v in the buffer zone accurate to $O(\Delta t^2)$.

e. Numerical testing (1): Advection

Let us perform numerical tests using a *slow solution* for which $u = 0$, and $\bar{f}v = \partial \phi/\partial x$, and the analytical solution transports these fields with a velocity \bar{u} without changing their shape.

With this solution we mirror what we would be attempting with the full three-dimensional system of equations in an operational context. There, the analysis and initialization produces a well-balanced initial state; the slow initial state mirrors this. On the boundary we want the meteorological fields to enter the domain without corruption from the gravity waves; here we model this by imposing the slow solution on the boundary.

For ϕ we use the same bell curve as in section 2; thus

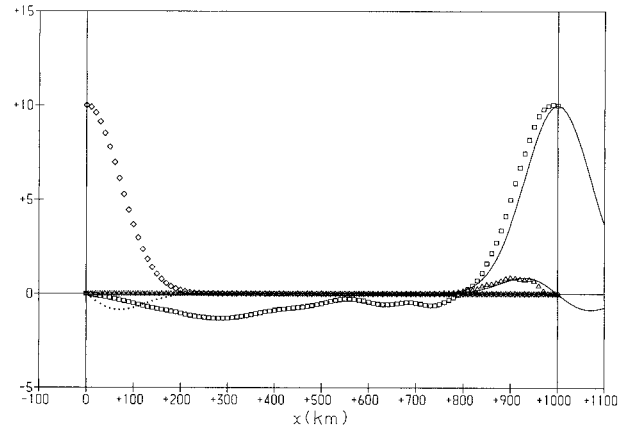


FIG. 7. The solution of Eqs. (3.14)–(3.17) with initial and boundary conditions given by Eq. (3.39) and ϕ imposed at both boundaries and v at inflow. We use the trajectory truncation scheme. At time zero ϕ is shown by the diamonds, u by the plus signs, and v by the dots. The analytical solutions at time T for all three fields are shown by symbol-free lines. The result of integrating with $\Delta t = 416.667$ s, $\bar{c} = 300$ m s $^{-1}$, $\bar{u} = 100$ m s $^{-1}$, and $\alpha = 4.166 67$ is displayed as squares for ϕ , \times 's for u , and triangles for v .

$$\phi(x, t) = \hat{\phi} \exp \left\{ -\left[\frac{(x - \bar{u}t)}{(L/10)} \right]^2 \right\},$$

$$v(x, t) = \frac{-2(x - \bar{u}t)}{\bar{f}(L/10)^2} \phi(x, t), \quad u(x, t) = 0. \quad (3.39)$$

To crudely mirror the meteorological situation of a “front” entering the area through the western boundary and exiting through the eastern boundary we will begin the integration by placing the bell curve ϕ solution with its maximum on the western boundary. We will then impose boundary conditions consistent with the bell curve moving in the positive x direction with a velocity \bar{u} .

We use $L = 1000$ km and $\Delta x = 10$ km. In addition, $\bar{f} = 10^{-4}$. In the tests that follow the length of the integration (T) will always be chosen such that in the analytical solution the bell curve moves *exactly* a distance L .

Let us impose the boundary conditions on $\phi(0, t)$ and $\phi(L, t)$ and $v(0, t)$ given by Eq. (3.39). The initial state, which can be seen in Fig. 7 is given by Eq. (3.39) with $t = 0$.

First, the forecast was run using trajectory truncation. That is, Eqs. (3.14)–(3.17) were solved with R^u , R^v , and R^ϕ computed as described in section 3b when the departure point was adjacent to the boundary. The result is shown in Fig. 7. The positive aspects of this forecast are (a) that the front has entered and exited the area reasonably accurately, (b) that there are no false reflections at the boundary, and (c) that there is no short-wave noise generated. The negative aspects are, of course, (a) the “tail,” which the ϕ solution has attained, and (b) the slowing down of the ϕ and v solutions, just as in section 2.

Compare this with the forecast obtained using time

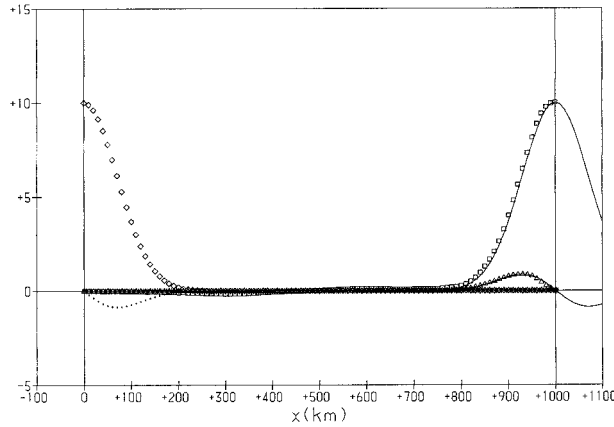


FIG. 8. Same as Fig. 7 but now using time interpolation on the boundary instead of trajectory truncation.

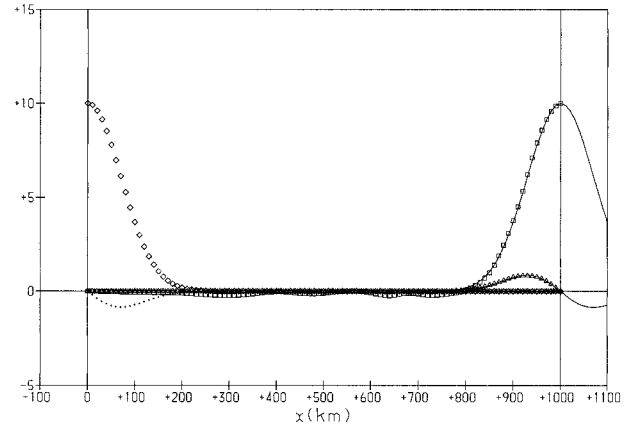


FIG. 9. Same as Fig. 7 but now using a well-posed buffer zone instead of trajectory truncation.

interpolation on the boundary. That is, Eqs. (3.14)–(3.17) were solved when the departure point was within the boundary, and Eqs. (3.31)–(3.33) when the departure point was outside the boundary. The result is shown in Fig. 8. The positive aspects of the solution are retained and the negative ones eliminated. The solution is almost as good as the analytical one.

The well-posed buffer zone also gives stable forecasts, better than trajectory truncation. In the region $x < 800$ km the forecast is not quite as good as the time interpolation forecast; for $x > 800$ km it is marginally better; see Fig. 9.

f. Numerical testing (2): Adjustment

The analysis of Oliger and Sundström (1978) does not derive a *unique* set of fields required for well posedness. There are many choices. Obviously, “transparent” boundaries, that is, those that allow waves to pass through them without reflection, are the most desirable. In this section we demonstrate that imposing p/q on the boundary allows the gravity waves to pass through with little or no reflection, making it a more attractive option than imposing ϕ , which causes the gravity waves to be reflected.

Let us model the steady-state solution to the Rossby adjustment problem, described, for example, in Gill (1982, section 7.2.2). The initial state is given by

$$\begin{aligned} \phi(x, 0) &= \begin{cases} \hat{\phi} & \text{for } x \leq L/2 \\ -\hat{\phi} & \text{for } x > L/2; \end{cases} & u(x, 0) = 0 & \text{and} \\ v(x, 0) &= 0 & \text{for } 0 \leq x \leq L. \end{aligned} \quad (3.40)$$

If we choose $\bar{f} = 10^{-4}$ and $\sqrt{\bar{\phi}} = 300 \text{ m s}^{-1}$, then $L = 30\,000 \text{ km}$ is 10 times the Rossby deformation radius ($\sqrt{\bar{\phi}/|\bar{f}|}$). Thus, the geostrophically balanced large-scale flow at the boundaries will remain steady with time. We set $\bar{u} = 1 \text{ m s}^{-1}$ so that we know we must impose v at $x = 0$. Last, $\Delta x = 100 \text{ km}$, $\Delta t = 10 \text{ min}$, and $\hat{\phi} = 10$. Because the boundary fields are constant

and α is minuscule we will use trajectory truncation for both tests.

First let us examine what happens if we impose ϕ on both boundaries and v at inflow:

$$\phi(0, t) = \hat{\phi}; \quad \phi(L, t) = -\hat{\phi}; \quad v(0, t) = 0. \quad (3.41)$$

The 10-day forecast is shown in Fig. 10. It is badly corrupted by the gravity waves, which have been reflected from the boundaries.

Next, let us impose our boundary values on the fields corresponding to the ingoing characteristics valid at the ϕ points, that is, p_0 and q_{i_L} . How can we proceed? Let us assume that the following equations are valid at $i = 0$ and $i = i_L$: (3.11), (3.13), and (3.14); also Eq. (3.16) at outflow, ($i = i_L$). Now use the imposed boundary conditions to determine $u_{-1/2}$ and $u_{i_L+1/2}$:

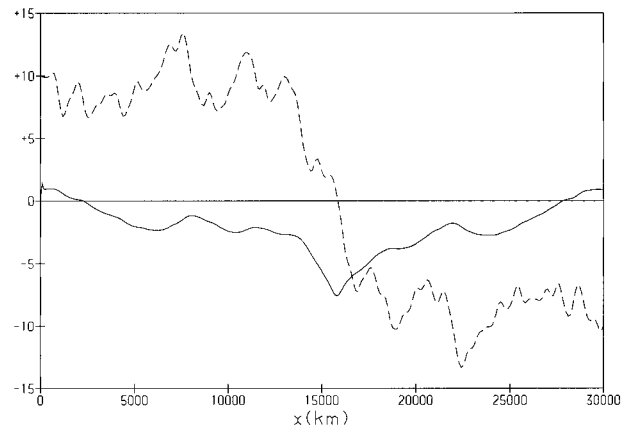


FIG. 10. The solution of Eqs. (3.14)–(3.17) after 10 days. The dashed line represents the ϕ solution; the solid line the v solution multiplied by \bar{c} . The initial conditions are given by Eq. (3.40). On the boundaries we hold ϕ constant and impose v at inflow as described in Eq. (3.41).

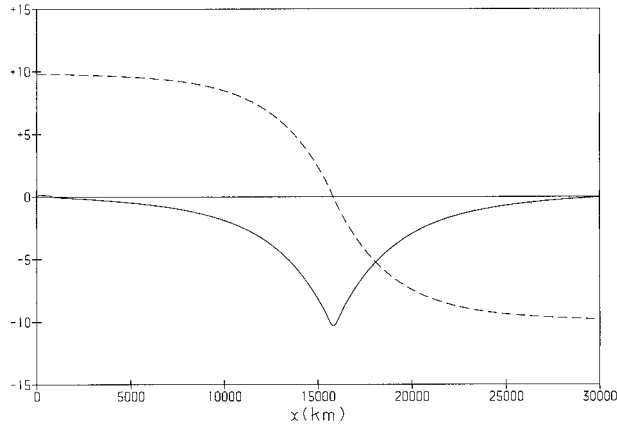


FIG. 11. Same as Fig. 10 except that on the boundaries, while still imposing v at inflow, we now hold p and q constant, as described in Eq. (3.45).

$$\begin{aligned}
 u_{-(1/2)} &= \frac{2}{c}(p_0 - \phi_0) - u_{1/2} \\
 u_{i_L+(1/2)} &= \frac{2}{c}(\phi_{i_L} - q_{i_L}) - u_{i_L-(1/2)}. \quad (3.42)
 \end{aligned}$$

Then substituting for $u_{1/2}$ and $u_{i_L-(1/2)}$ from Eq. (3.15) into Eq. (3.14) yields the following two equations for ϕ_0 and ϕ_{i_L} , which closes the system of equations (3.17):

$$\begin{aligned}
 &\left(1 + \frac{\bar{\phi}\Delta t}{\bar{c}\Delta x} + \frac{\bar{\phi}}{2\Delta x^2}\frac{\Delta t^2}{\bar{c}}\right)\phi_0^{(k+1)} - \frac{\bar{\phi}}{2\Delta x^2}\phi_1^{(k+1)} \\
 &= R_0^\phi - \frac{\bar{\phi}}{\Delta x}\left(\tilde{R}_{1/2}^u - \frac{p_0^{n+1}}{\bar{c}}\right), \quad (3.43)
 \end{aligned}$$

$$\begin{aligned}
 &\left(1 + \frac{\bar{\phi}\Delta t}{\bar{c}\Delta x} + \frac{\bar{\phi}}{2\Delta x^2}\frac{\Delta t^2}{\bar{c}}\right)\phi_{i_L}^{(k+1)} - \frac{\bar{\phi}}{2\Delta x^2}\phi_{i_L-1}^{(k+1)} \\
 &= R_{i_L}^\phi + \frac{\bar{\phi}}{\Delta x}\left(\frac{q_{i_L}^{n+1}}{\bar{c}} + \tilde{R}_{i_L-(1/2)}^u\right). \quad (3.44)
 \end{aligned}$$

The trajectories are truncated as follows: $R^u = (X^u)_{1/2}^n$, when $x_* < \Delta x/2$; $R^u = (X^u)_{i_L-1/2}^n$, when $x_* > L - \Delta x/2$; $R^\phi = (X^\phi)_0^n$, when $x_* < 0$; $R^\phi = (X^\phi)_{i_L}^n$, when $x_* > L$; $R^v = (X^v)_0^n$, when $x_* < 0$; and $R^v = (X^v)_{i_L}^n$, when $x_* > L$.

We are now ready to do our second test. We impose v and p at $x = 0$, and q at $x = L$,

$$p(0, t) = \hat{\phi}; \quad q(L, t) = -\hat{\phi}; \quad v(0, t) = 0, \quad (3.45)$$

and take care of the resulting complications as described in Eqs. (3.42)–(3.44). The resulting 10-day forecast is no longer seriously contaminated by gravity waves, as can be seen from Fig. 11.

4. Stability

When the departure point is outside the integration area an added complication is introduced into the discretized initial-boundary-value problem. We have seen that our three solutions to this complication, trajectory truncation, a well-posed buffer zone, and time interpolation, yielded stable integrations in practical tests. In this section we make a first attempt at confirming these results analytically by using the energy method to demonstrate the stability of the one-dimensional advection equation with constant velocity $\bar{u} > 0$:

$$\begin{aligned}
 \frac{\partial \phi(x, t)}{\partial t} + \bar{u}\frac{\partial \phi(x, t)}{\partial x} &= 0; \quad \phi(x, 0) = f(x); \\
 \phi(0, t) &= g(t), \quad (4.1)
 \end{aligned}$$

where $0 \leq x \leq L$ and $t \geq 0$.

Using a linear interpolation to find the departure point value of the field the semi-Lagrangian discretization of Eq. (4.1) gives

$$\begin{aligned}
 \phi_i^{n+1} &= (1 - \hat{\alpha})\phi_{i-p}^n + \hat{\alpha}\phi_{i-p-1}^n \quad \phi_i^0 = f_i; \\
 \phi_0^n &= g^n. \quad (4.2)
 \end{aligned}$$

Squaring both sides and rearranging the terms on the right-hand side results in

$$\begin{aligned}
 (\phi_i^{n+1})^2 &= (\phi_{i-p}^n)^2 - (\hat{\alpha} - \hat{\alpha}^2)(\phi_{i-p}^n - \phi_{i-p-1}^n)^2 \\
 &\quad - \hat{\alpha}(\phi_{i-p}^n)^2 + \hat{\alpha}(\phi_{i-p-1}^n)^2. \quad (4.3)
 \end{aligned}$$

a. Trajectory truncation

This assumption implies that for the first p points Eq. (4.2) is replaced with

$$\phi_1^{n+1} = \phi_2^{n+1} = \dots = \phi_p^{n+1} = g^n. \quad (4.4)$$

Therefore we get the following when we sum over the grid points:

$$\begin{aligned}
 \sum_{i=1}^{i_L} (\phi_i^{n+1})^2 &= (p + \hat{\alpha})(g^n)^2 + \sum_{i=1}^{i_L-p} (\phi_i^n)^2 \\
 &\quad - (\hat{\alpha} - \hat{\alpha}^2) \sum_{i=1}^{i_L-p} (\phi_i^n - \phi_{i-1}^n)^2 \\
 &\quad - \hat{\alpha}(\phi_{i_L-p}^n)^2. \quad (4.5)
 \end{aligned}$$

Recall that the stability of the semi-Lagrangian scheme is dictated by the condition $0 \leq \hat{\alpha} \leq 1$. Therefore $0 \leq (\hat{\alpha} - \hat{\alpha}^2) \leq 1$, enabling us to write

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq (p + \hat{\alpha})(g^n)^2 + \sum_{i=1}^{i_L} (\phi_i^n)^2. \quad (4.6)$$

Using this equation again to substitute for ϕ_i^n on the right-hand side we get

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq (p + \hat{\alpha})[(g^n)^2 + (g^{n-1})^2] + \sum_{i=1}^{i_L} (\phi_i^{n-1})^2. \quad (4.7)$$

Repeating this process we finally see that the solution is bounded by the initial condition and the boundary condition, thus establishing stability:

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq (p + \hat{\alpha}) \sum_{m=0}^n (g^m)^2 + \sum_{i=1}^{i_L} (f_i)^2. \quad (4.8)$$

b. Time interpolation

For the first p points,

$$\phi_\mu^{n+1} = \left(1 - \frac{\mu}{\alpha}\right) \phi_0^{n+1} + \frac{\mu}{\alpha} \phi_0^n, \quad \mu = 1, p. \quad (4.9)$$

Squaring this, summing from 1 to p , and rearranging the terms we arrive at

$$\sum_{\mu=1}^p (\phi_\mu^{n+1})^2 = p(\phi_0^{n+1})^2 - \frac{p(p+1)}{2\alpha} \left[1 - \frac{(2p+1)}{3\alpha}\right] \times (\phi_0^n - \phi_0^{n+1})^2 - \frac{p(p+1)}{2\alpha} [(\phi_0^{n+1})^2 - (\phi_0^n)^2]. \quad (4.10)$$

Since we only “time interpolate” when $\alpha > p \geq 1$, we can write

$$\sum_{\mu=1}^p (\phi_\mu^{n+1})^2 \leq p(\phi_0^{n+1})^2 + \frac{p(p+1)}{2\alpha} (\phi_0^n)^2. \quad (4.11)$$

Once the departure point is inside the integration area we switch to the standard space interpolation scheme, which means the remainder of the derivation is the same as above, resulting in

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq \sum_{i=1}^{i_L} (f_i)^2 + \sum_{m=0}^n \left[p(g^{m+1})^2 + \left(\hat{\alpha} + \frac{p(p+1)}{2\alpha}\right) (g^m)^2 \right]. \quad (4.12)$$

Thus the solution is bounded by the initial condition and the boundary condition, establishing stability.

c. Well-posed buffer zone

For the points outside the area we define the fields in terms of the imposed boundary condition:

$$\phi_i^n = \left(g + \frac{i\Delta x}{\bar{u}} \frac{dg}{dt} + \frac{(i\Delta x)^2}{2\bar{u}^2} \frac{d^2g}{dt^2} + \dots \right)^n. \quad (4.13)$$

Thus, in Eq. (4.3) the fields outside the area are defined in terms of the boundary condition and when we sum over all the grid points we get

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 = \sum_{i=-p+1}^{i_L-p} (\phi_i^n)^2 - (\hat{\alpha} - \hat{\alpha}^2) \sum_{i=-p+1}^{i_L-p} (\phi_i^n - \phi_{i-1}^n)^2 + \hat{\alpha}(\phi_{-p}^n)^2 - \hat{\alpha}(\phi_{i_L-p}^n)^2. \quad (4.14)$$

Since $(\hat{\alpha} - \hat{\alpha}^2) \leq 1$ we can write

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq \sum_{i=1}^{i_L} (\phi_i^n)^2 + \sum_{i=-p+1}^0 (\phi_i^n)^2 + \hat{\alpha}(\phi_{-p}^n)^2. \quad (4.15)$$

As before we can continue this procedure until we arrive at

$$\sum_{i=1}^{i_L} (\phi_i^{n+1})^2 \leq \sum_{i=1}^{i_L} (f_i)^2 + \sum_{m=0}^n \left[\hat{\alpha}(\phi_{-p}^m)^2 + \sum_{i=-p+1}^0 (\phi_i^m)^2 \right]. \quad (4.16)$$

Using Eq. (4.13), we finally see that the solution is bounded by the initial condition and the boundary condition, thus establishing stability, provided g is sufficiently smooth.

5. Discussion

As we have seen, inventing stable boundaries even for simple one-dimensional systems is a nontrivial task. When we move to two (or three) dimensions the complications multiply. As is discussed in Durran (1999, section 8.2), there are fundamental difficulties encountered when we try to extend to two dimensions the idea of controlling the gravity waves as we did in section 3f. Nevertheless, Engquist and Majda (1977) have established a method constructing boundary conditions that are well posed and theoretically perfectly absorbing at least for waves striking at angles sufficiently close to normal incidence. For waves impinging at other angles of incidence (except tangential) reflections at the boundary can be reduced to a minimum by their method. The plan is to use their ideas and those discussed in this paper to construct a two-dimensional semi-implicit semi-Lagrangian shallow water model with boundaries that are “as transparent as possible.”

Extrapolations from the interior were found to be unavoidable. However, they did not lead to instabilities in our tests. This is consistent with the findings of Gustafsson et al. (1972), who proved that first-order extrapolations are stable for the Lax–Wendroff scheme (which is the same as a semi-Lagrangian scheme with quadratic interpolation and $|\alpha| < 1$).

We have seen that both the time interpolation and well-posed buffer zone schemes overcome the accuracy limitation of trajectory truncation for large $|\alpha|$. Looking ahead, it is interesting to ask what are the limitations of the former schemes in the context of an operational numerical weather prediction model? There is a limi-

tation that applies to both; that is, their accuracy depends on how often the boundary is updated. For instance, if this is done every 3 h, neither scheme will be much more accurate than trajectory truncation. Each scheme has a particular limitation, at least as we have implemented them here. Convergence may be slow for an iterative implementation of the time interpolation scheme. Also, for our particular implementation of the well-posed buffer zone, looking at Eqs. (3.36) and (3.37) we see that it cannot be used when $|\bar{u}| = |\bar{c}|$, which can happen for internal gravity waves. Last, solving the two-dimensional equivalent of Eq. (3.17) will require the inversion of a sparse matrix. Thus any iterative procedures will be potentially expensive. This does not condemn either method but warns us that these preliminary implementations are not without possible weaknesses.

Although these are early days it is still nice to have a “fallback” position. One is to use trajectory truncation and accept the associated inaccuracy when $|\alpha|$ is large on the boundary. The second is based on an idea of J. E. Haugen (1999, personal communication). If $u\Delta t/\Delta x < 0.5$ on the western boundary, say, we need never invoke the time interpolation nor well-posed buffer zone schemes. To accomplish this we surround the uniformly discretized area with a buffer zone in which we use stretched coordinates, each Δx being, say, 15% bigger than its neighbor until this criterion is fulfilled for the maximum advecting velocity we expect perpendicular to the boundary. We choose this as our boundary and apply trajectory truncation there as an insurance against instability for unanticipated large inflow velocities.

Finally, although we have concentrated on looking at well-posed boundaries for semi-Lagrangian schemes in this paper, Eulerian systems are equally problematic. Consider an Eulerian approach using fourth-order spatial differencing. This will, in principle, require knowledge of fields outside the boundaries. Thus, stable extrapolations will be required to update the fields in the vicinity of the boundary. A possible solution is to adopt the upstream ideas inherent in the semi-Lagrangian approach. See also, however, Elvius and Sundström (1973) for an alternative approach.

Acknowledgments. Suggestions for improving the original manuscript by an anonymous reviewer and Dale Durran are gratefully acknowledged. In particular, the latter’s comment that the Rossby adjustment problem would provide an interesting challenge inspired section 3f.

REFERENCES

- Bates, J. R., and A. McDonald, 1982: Multiply-upstream, semi-Lagrangian advective schemes: Analysis and application to a multi-level primitive equation model. *Mon. Wea. Rev.*, **110**, 1831–1842.
- Cats, G. J., and O. Åkesson, 1983: An investigation into a marked difference between two successive ECMWF forecasts of September, 1982. *Beitr. Phys. Atmos.*, **56**, 440–451.
- Davies, H. C., 1976: A lateral boundary formulation for multi-level prediction models. *Quart. J. Roy. Meteor. Soc.*, **102**, 405–418.
- Durran, D. R., 1999: *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag, 465 pp.
- Elvius, T., and A. Sundström, 1973: Computationally efficient schemes and boundary conditions for a fine mesh barotropic model based on the shallow water equations. *Tellus*, **25**, 132–156.
- Engquist, B., and A. Majda, 1977: Absorbing boundary conditions for the numerical simulation of waves. *Math. Comput.*, **31**, 629–651.
- Falcone, M., and R. Ferretti, 1998: Convergence analysis for a class of high-order semi-Lagrangian advection schemes. *SIAM J. Numer. Anal.*, **35**, 909–940.
- Gill, A. E., 1982: *Atmosphere–Ocean Dynamics*. Academic Press, 662 pp.
- Gustafsson, B., H.-O. Kreiss, and A. Sundström, 1972: Stability theory of difference approximations for mixed initial boundary value problems. *Math. Comput.*, **26**, 649–686.
- Gustafsson, N., E. Källén, and S. Thorsteinsson, 1998: Sensitivity of forecast errors to initial and lateral boundary conditions. *Tellus*, **50A**, 167–185.
- Le Dimet, F. X., and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, **38**, 97–110.
- McDonald, A., 1984: Accuracy of multiply-upstream, semi-Lagrangian advective schemes. *Mon. Wea. Rev.*, **112**, 1267–1275.
- , 1998: Conservation of mass in HIRLAM. *HIRLAM Newsletter*, Vol. 32, 39–44. [Available from A. McDonald, Irish Meteorological Service, Glasnevin Hill, Dublin 9, Ireland.]
- , 1999a: A review of lateral boundary conditions for limited area forecast models. *Proc. Ind. Nat. Sci. Acad.*, **65**, 91–105.
- , 1999b: Well-posed boundary conditions for semi-Lagrangian schemes: The one-dimensional case. HIRLAM Tech. Rep. 43, HIRLAM Group, 26 pp. [Available from A. McDonald, Irish Meteorological Service, Glasnevin Hill, Dublin 9, Ireland.]
- , 2000: Well-posed boundary conditions for semi-Lagrangian schemes: The one-dimensional case; part II. HIRLAM Tech. Rep. 44, HIRLAM Group, 19 pp. [Available from A. McDonald, Irish Meteorological Service, Glasnevin Hill, Dublin 9, Ireland.]
- Oliger, J., and A. Sundström, 1978: Theoretical and practical aspects of some initial boundary value problems in fluid dynamics. *SIAM J. Appl. Math.*, **35**, 419–446.
- Robert, A., and E. Yakimiw, 1986: Identification and elimination of an inflow boundary computational solution in limited area model integrations. *Atmos.–Ocean*, **24**, 369–385.
- Skålin, R., and I. Lie, 1995: On the stability of semi-Lagrangian advection with interpolation in time. DNMI, 24 pp. [Available from Ivar Lie, DNMI, Box 43, Blindern, N-0313 Oslo, Norway.]
- Sundström, A., and T. Elvius, 1979: Computational problems related to limited area modelling. *Numerical Methods Used in Atmospheric Models*, Vol. 2, GARP Publ. Series, No. 17, 379–416.