

# Modeling Sea Ice Transport Using Incremental Remapping

William H. Lipscomb \*

Elizabeth C. Hunke

*Group T-3, Los Alamos National Laboratory*

*Monthly Weather Review*, in press 2004

December 4, 2003

## Abstract

Sea ice models contain transport equations for the area, volume, and energy of ice and snow in various thickness categories. These equations typically are solved with first-order accurate upwind schemes, which are very diffusive; with second-order accurate centered schemes, which are highly oscillatory; or with more sophisticated second-order schemes that are computationally costly if many quantities must be transported (e.g., MPDATA). Here an incremental remapping scheme, originally designed for horizontal transport in ocean models, is adapted for sea ice transport. This scheme has several desirable features: it preserves the monotonicity of both conserved quantities and tracers; it is second-order accurate except where the accuracy is reduced locally to preserve monotonicity; and it efficiently solves the large number of equations in sea ice models with multiple thickness categories and tracers. Remapping outperforms the first-order upwind scheme and basic MPDATA scheme in several simple test problems. In realistic model runs, remapping is less

diffusive than the upwind scheme and about twice as fast as MPDATA.

## 1. Introduction

Sea ice has long been recognized as important for climate because of its high albedo, strong insulating effect, and potential sensitivity to greenhouse warming. Many global models predict that reductions in sea ice thickness and extent will amplify climate change at high latitudes. For many years sea ice was represented fairly crudely in large-scale models. More sophisticated approaches were available—for example, the models of Maykut and Untersteiner (1971) for sea ice thermodynamics, Hibler (1979) for dynamics, and Thorndike et al. (1975) for the ice thickness distribution—but were considered too complex and expensive for global climate simulations. Recently, however, sea ice models have become more realistic as computing power has increased. Large-scale models now are more likely to include a realistic ice rheology (Hunke and Dukowicz 1997), multilayer thermodynamics (Bitz and Lipscomb 1999), and a multi-category thickness distribution (Bitz et al. 2001; Lipscomb 2001).

Sea ice models transport area, mass, and in-

---

\* *Corresponding author address:* Dr. William H. Lipscomb, MS B216, Los Alamos National Laboratory, Los Alamos, NM 87545. E-mail: *Lipscomb@lanl.gov*

ternal energy in horizontal  $(x, y)$  and thickness  $(h)$  space. The basic equation for the transport of ice area is (Thorndike et al. 1975)

$$\frac{\partial g}{\partial t} + \nabla \cdot (g\mathbf{u}) + \frac{\partial}{\partial h}(gf) = \psi, \quad (1)$$

where  $\nabla = (\partial/\partial x, \partial/\partial y)$ ,  $\mathbf{u}$  is the horizontal velocity,  $f$  is the rate of thermodynamic ice growth,  $\psi$  is a mechanical redistribution function, and  $g$  is the ice thickness distribution function. We define  $g(\mathbf{x}, t, h) dh$  as the fractional area covered by ice in the thickness range  $(h, h + dh)$  at a given time and location. If open water is classified as ice of zero thickness, then it follows from the definition of  $g$  that

$$\int_0^\infty g(h) dh = 1. \quad (2)$$

The source term  $\psi$  creates or destroys open water and builds ridges as necessary to satisfy (2) as the ice pack deforms. Similar equations describe the transport of the mass and energy of the ice and overlying snow.

Equation (1) typically is solved in several stages by operator splitting, with all but one forcing term set to zero in each stage. Here we consider the equation for horizontal transport of ice area:

$$\frac{\partial a_n}{\partial t} + \nabla \cdot (a_n \mathbf{u}) = 0, \quad (3)$$

which is obtained from (1) when thermodynamics and mechanical redistribution are neglected and the ice thickness distribution is discretized. The ice is partitioned into  $N_C$  categories, each with fractional area  $a_n$  and thickness boundaries  $H_{n-1}$  and  $H_n$ . That is,

$$a_n = \int_{H_{n-1}}^{H_n} g(h) dh. \quad (4)$$

Equation (3) is solved for  $n = 0$  to  $N_C$ , where  $a_0$  denotes the open water fraction. Area is conserved globally in each category. The corresponding equations for volume and internal energy are

$$\frac{\partial(a_n h_n)}{\partial t} + \nabla \cdot (a_n h_n \mathbf{u}) = 0, \quad (5)$$

$$\frac{\partial(a_n h_n q_n)}{\partial t} + \nabla \cdot (a_n h_n q_n \mathbf{u}) = 0, \quad (6)$$

where  $h_n$  is the ice or snow thickness and  $q_n$  is the enthalpy, defined as the energy needed to melt a unit volume and raise its temperature to  $0^\circ\text{C}$ . These equations are solved for  $n = 1$  to  $N_C$ , possibly in several vertical layers. For simplicity, ice and snow are assumed to have constant densities, so that volume conservation is equivalent to mass conservation.

There are many equations of the form (3), (5), and (6) in sea ice models with multiple thickness categories and vertical layers. The Los Alamos sea ice model, CICE (Hunke and Lipscomb 2001), is typically run with five thickness categories, enough to simulate accurately the annual cycles of ice thickness, ice strength, and surface fluxes (Bitz et al. 2001; Lipscomb 2001). Each category is characterized by a fractional area, a surface temperature, the thicknesses of ice and snow, and the enthalpies of four ice layers and one snow layer. The quantities conserved under horizontal transport for each category are the ice area  $a$ , the area-weighted surface temperature  $aT_s$ , the ice and snow volumes  $v = ah$ , and the internal energies  $e = a \Delta h q$ , where  $\Delta h$  is the thickness of a particular layer. (Henceforth we drop the subscript  $n$  for compactness.) In standard CICE runs there are 46 transported fields—nine in each category, plus the open water area. As models become more realistic, additional fields will likely need to be transported: for example, ice salinity, snow properties in multiple layers, and melt pond fraction and depth.

There are many numerical methods for solving transport equations (LeVeque 1992). An ideal method for sea ice transport would have the following features:

- The method is conservative. Conservation may be ensured by writing the equations in terms of fluxes across cell edges.
- The method is stable. For many schemes, numerical stability requires that

the time step  $\Delta t$  satisfy a Courant-Friedrichs-Lewy (CFL) condition of the form  $\max |\mathbf{u}| \Delta t / \Delta x \leq 1$ .

- The method is at least second-order accurate in space, since first-order schemes are very diffusive. Second-order accuracy in time is less important, because  $|\mathbf{u}| \Delta t / \Delta x \ll 1$  in most grid cells most of the time.
- The method preserves monotonicity of the conserved fields and tracers. Monotonicity is discussed in section 2.
- The method is computationally efficient for systems of many equations.

No method previously used in global sea ice models combines all these features.

In this paper we adapt the incremental remapping method of Dukowicz and Baumgardner (2000) (henceforth “DB”) for sea ice transport. Section 2 discusses useful properties of sea ice transport schemes and describes several potential schemes, including remapping. Section 3 discusses incremental remapping in detail, emphasizing new features developed to implement the scheme in a global sea ice model. Section 4 compares remapping to the MPDATA and upwind schemes in three simple test problems. In section 5 we compare these schemes using a global sea ice model with realistic atmospheric and ocean forcing, and we conclude with general remarks in section 6.

## 2. Sea ice transport schemes

The transport equations (3), (5), and (6) have several properties we would like to preserve when finding a numerical solution. For example, these equations are sign-preserving; any field that is nonnegative at an initial time remains so at future times. For the special case of a

nondivergent velocity field, (3) becomes a simple advection equation:

$$\frac{da}{dt} \equiv \frac{\partial a}{\partial t} + \mathbf{u} \cdot \nabla a = 0, \quad (7)$$

and similarly for (5) and (6). In this case the conserved quantities are unchanged following the motion. In any location the value of a conserved quantity at time  $t + \Delta t$  is bounded by the values in the local neighborhood, defined as the set of points within a distance  $c\Delta t$ , where  $c$  is the fluid speed. If a numerical scheme maintains this property—if it does not create new maxima or minima in conserved quantities under nondivergent transport—it is said to preserve monotonicity. Divergent velocity fields may, of course, create new extrema. For divergent flow a scheme is monotonicity-preserving if it reproduces the physical extrema without generating spurious numerical extrema. Numerical schemes violate monotonicity to varying degrees. For example, oscillatory schemes create unwanted ripples in initially smooth fields, typically in regions with sharp gradients. Nonoscillatory schemes can still violate monotonicity; they may preserve the number of extrema but allow existing extrema to grow beyond their initial values.

From (3), (5), and (6) we can derive advection equations for thickness and enthalpy:

$$\frac{dh}{dt} \equiv \frac{\partial h}{\partial t} + \mathbf{u} \cdot \nabla h = 0, \quad (8)$$

$$\frac{dq}{dt} \equiv \frac{\partial q}{\partial t} + \mathbf{u} \cdot \nabla q = 0, \quad (9)$$

which are valid for any velocity field. These quantities are unchanged following the motion; at each point the value at  $t + \Delta t$  is bounded by nearby values at  $t$ . If a numerical scheme preserves this property for any tracer obeying an equation of the form (8) or (9), it is said to be compatible (Schär and Smolarkiewicz 1996). Tracer advection equations usually are not solved explicitly in sea ice models. Instead, equations (3), (5), and (6) are solved to obtain

the new area, volume, and energy. Then the volume is divided by area to find the new thickness, and energy is divided by volume to find the new enthalpy. For this reason a numerical scheme that preserves monotonicity for the conserved quantities ( $a$ ,  $v$ ,  $e$ ) is not necessarily compatible for the tracers ( $h$ ,  $q$ ). Compatibility is highly desirable for sea ice transport, since noncompatible schemes produce spurious thickness and temperature changes. Most conservative transport schemes, however, are not compatible.

The simplest method commonly used for sea ice transport is the first-order upwind, or donor cell, scheme. This method is conservative, stable, fast, monotonicity-preserving, and compatible (see Schär and Smolarkiewicz (1996) for a proof of its compatibility), but also very diffusive. First-order upwind schemes are used for sea ice transport in several climate models, including those of the Canadian Centre for Climate Modelling and Analysis (G. M. Flato 2001, personal communication), the Geophysical Fluid Dynamics Laboratory (Winton 2001), the Hadley Centre (Cattle and Crossley 1995), and the Max Planck Institute for Meteorology (U. Mikolajewicz 2001, personal communication).

The simplest second-order accurate methods are centered-in-space schemes. The centered-in-time, centered-in-space scheme is used in many ocean models (Semtner 1986) but is highly oscillatory and is not sign-preserving. Hibler (1979) used a scheme that is centered-in-space and modified-Euler in time, with harmonic and bi-harmonic diffusion terms added to suppress oscillations. While this scheme gives reasonable results, the artificial diffusion reduces its accuracy.

In order to limit diffusion while retaining smooth, nonnegative fields, the MPDATA (multidimensional positive-definite advection transport algorithm) scheme (Smolarkiewicz 1984) was chosen for the original version of CICE (Hunke 1998). MPDATA also has been used in the Community Sea Ice Model (CSIM) (Briegleb

et al. 2003), the sea ice component of the Community Climate System Model (CCSM) of the National Center for Atmospheric Research (NCAR). The method consists of a series of upwind steps. The first step uses the physical velocity, and subsequent steps use “antidiffusive” velocities that reduce the truncation error. In its basic form MPDATA is conservative, second-order accurate, and sign-preserving. It is weakly oscillatory and therefore is neither monotonicity-preserving nor compatible. However, several enhanced versions of MPDATA have been developed. For example, Schär and Smolarkiewicz (1996) developed a compatible flux-corrected transport scheme using MPDATA, and Margolin and Smolarkiewicz (1998) introduced a third-order accurate MPDATA that preserves monotonicity.

MPDATA is expensive compared to upwind and centered-in-space differencing. Its expense was not an issue for the original CICE, which, with two thickness categories, transported only five fields: ice area for the thicker category plus ice and snow volume for both. In the current version, however, 46 fields are transported. Since the cost per field is constant, the computational cost of transport with MPDATA increased by an order of magnitude compared to the earlier model, to about 40% of the total run time. Switching from the basic MPDATA to monotonicity-preserving or compatible versions would increase the cost further. For example, the scheme of Margolin and Smolarkiewicz (1998) is about twice as expensive as the basic MPDATA.

Merryfield and Holloway (2003) recently applied the second-order moment (SOM) scheme of Prather (1986) to sea ice transport. SOM is a modified upwind scheme that reduces diffusion by transporting not only the mean fields, but also their first- and second-order moments—a total of six fields in two dimensions. This scheme very nearly preserves monotonicity for conserved fields (ice area and volume), though not for tracers (ice thickness). It is third-order

accurate in space except where limited to prevent overshoots and undershoots. SOM is relatively inexpensive when used to transport just a few fields but, like MPDATA, probably would prove costly if applied to large numbers of fields as in CICE. Russell and Lerner (1981) developed a similar method, the linear upstream or “slopes” scheme, which is used for sea ice transport in the Goddard Institute for Space Studies climate model. This scheme transports first-order but not second-order moments and thus is second-order accurate in space.

All of these methods are conservative and reasonably stable, but none combines the desired features of spatial accuracy, monotonicity, compatibility, and computational efficiency. For this reason we have implemented the 2D incremental remapping scheme of DB, originally designed for horizontal transport in isopycnic ocean models. The transport equation is solved by projecting model grid cells backward in time along Lagrangian trajectories. Scalar fields at time  $t$  are reconstructed over the grid, integrated over the Lagrangian departure regions, and remapped onto the grid at time  $t + \Delta t$ . When the conserved fields are constructed appropriately, remapping has all the desirable features listed above. It is conservative by virtue of being written in flux form. Moreover, integrals of the reconstructed fields recover the total mass and tracer in each grid cell. The spatial accuracy depends on the accuracy of the reconstruction. If the fields are constant within each grid cell, remapping is a first-order scheme, but if the fields vary linearly in  $x$  and  $y$ , as assumed here, the scheme is formally second-order accurate. Quadratic reconstructions would give third-order accuracy, but with much added complexity. Remapping preserves monotonicity when the field gradients are limited appropriately; this limiting may reduce the accuracy locally to first-order. Another key property of remapping is that fluxes of the various conserved quantities are computed in a way that ensures tracer compatibility.

Remapping is relatively expensive for trans-

porting a single field but very efficient for multiple categories and tracers. Much of the work is geometrical and is performed just once per time step instead of being repeated for each conserved field. For nondivergent velocity fields the time step is limited by the requirement that trajectories are confined to neighboring grid cells; this is what is meant by incremental as opposed to general remapping. This requirement leads to a CFL-like condition for the velocities,  $\max |\mathbf{u}| \Delta t / \Delta x \leq 1$ . If the velocity field is divergent, the time step is limited by the more stringent condition  $\max |\mathbf{u}| \Delta t / \Delta x \leq 0.5$  to ensure that trajectories do not cross. These restrictions are the same as for a one-dimensional upwind scheme.

We modified the remapping scheme of DB to make it more suitable for sea ice transport. For example, the original scheme has no analog to the energy conservation equation (6). Energy is proportional to the product of two tracers, thickness and enthalpy, and DB did not consider tracer products. Also, the original scheme was designed for grids that are approximately rectangular, unlike the CICE grid, which is highly curved near the poles. With the appropriate changes, incremental remapping gives excellent results, as shown below.

### 3. Sea ice transport using incremental remapping

In this section we describe the remapping algorithm, focusing on features not found in DB. Readers interested in more details may refer to DB or the online CICE documentation at <http://climate.lanl.gov/Models/CICE/index.htm>.

The algorithm proceeds as follows:

1. Given mean values of the ice area and tracer fields in each grid cell, construct linear approximations of these fields. Limit the field gradients to preserve monotonicity.
2. Given ice velocities at grid cell corners,

identify departure regions for the fluxes across each cell edge. Divide these departure regions into triangles and compute the coordinates of the triangle vertices.

3. Integrate each field over the departure triangles to obtain the area, volume, and energy fluxes across each cell edge.
4. Transfer the fluxes across cell edges and update the state variables.

Since all scalar fields are transported by the same velocity field, step (2) is done only once per time step. The other three steps are repeated for each field in each thickness category. Remapping is intrinsically a B-grid scheme (Arakawa and Lamb 1977); it assumes that velocities are defined at cell corners and scalars at cell centers. The overall time-stepping scheme is first-order accurate. Given the area and tracers at time  $t$ , the momentum equation is solved for the time  $t$  velocities, which are then used to remap the area and tracers to time  $t + \Delta t$ .

### *a. Reconstructing Area and Tracer Fields*

First, using the known values of the state variables, the ice area and tracer fields are reconstructed in each grid cell as linear functions of  $x$  and  $y$ . For each field we compute the value at the cell center (i.e., at the origin of a 2D Cartesian coordinate system defined for that grid cell) along with gradients in the  $x$  and  $y$  directions. The gradients are then limited to preserve monotonicity. When integrated over a grid cell, the reconstructed fields must have mean values equal to the known state variables, denoted by  $\bar{a}$  for fractional area,  $\tilde{h}$  for thickness, and  $\hat{q}$  for enthalpy. The mean values are not, in general, equal to the values at the cell center. For example, the mean ice area must equal the value at the centroid, which may not lie at the cell center.

Consider first the fractional ice area, the analog to fluid density in DB. For each thickness category we construct a field  $a(\mathbf{r})$  whose mean is  $\bar{a}$ , where  $\mathbf{r} = (x, y)$  is the position vector relative to the cell center. That is, we require

$$\int_A a dA = \bar{a} A, \quad (10)$$

where  $A = \int_A dA$  is the grid cell area. Equation (10) is satisfied if  $a(\mathbf{r})$  has the form

$$a(\mathbf{r}) = \bar{a} + \alpha_a \langle \nabla a \rangle \cdot (\mathbf{r} - \bar{\mathbf{r}}), \quad (11)$$

where  $\langle \nabla a \rangle$  is a centered estimate of the area gradient within the cell,  $\alpha_a$  is a limiting coefficient that enforces monotonicity, and  $\bar{\mathbf{r}}$  is the cell centroid:

$$\bar{\mathbf{r}} = \frac{1}{A} \int_A \mathbf{r} dA. \quad (12)$$

It follows from (11) that the ice area at the cell center ( $\mathbf{r} = 0$ ) is

$$a_c = \bar{a} - a_x \bar{x} - a_y \bar{y}, \quad (13)$$

where  $a_x = \alpha_a(\partial a / \partial x)$  and  $a_y = \alpha_a(\partial a / \partial y)$  are the limited gradients in the  $x$  and  $y$  directions, and the components of  $\bar{\mathbf{r}}$ ,  $\bar{x} = \int_A x dA / A$  and  $\bar{y} = \int_A y dA / A$ , are evaluated using the triangle integration formulas in section 3c. These means, along with higher-order means such as  $\overline{x^2}$ ,  $\overline{xy}$ , and  $\overline{y^2}$ , are computed once and stored.

Next, consider the ice and snow thickness and enthalpy fields. Thickness is analogous to the tracer concentration in DB, but there is no analog in DB to the enthalpy. The reconstructed thickness  $h(\mathbf{r})$  and enthalpy  $q(\mathbf{r})$  must satisfy

$$\int_A a h dA = \bar{a} \tilde{h} A, \quad (14)$$

$$\int_A a h q dA = \bar{a} \tilde{h} \hat{q} A. \quad (15)$$

Equations (14) and (15) are satisfied when  $h(\mathbf{r})$  and  $q(\mathbf{r})$  are given by

$$h(\mathbf{r}) = \tilde{h} + \alpha_h \langle \nabla h \rangle \cdot (\mathbf{r} - \bar{\mathbf{r}}), \quad (16)$$

$$q(\mathbf{r}) = \hat{q} + \alpha_q \langle \nabla q \rangle \cdot (\mathbf{r} - \bar{\mathbf{r}}), \quad (17)$$

where  $\alpha_h$  and  $\alpha_q$  are limiting coefficients,  $\tilde{\mathbf{r}}$  is the center of ice area:

$$\tilde{\mathbf{r}} = \frac{1}{\bar{a} A} \int_A a \mathbf{r} dA, \quad (18)$$

and  $\hat{\mathbf{r}}$  is the center of ice or snow volume:

$$\hat{\mathbf{r}} = \frac{1}{\bar{a} \tilde{h} A} \int_A a h \mathbf{r} dA. \quad (19)$$

Evaluating the integrals, we find that the components of  $\tilde{\mathbf{r}}$  are

$$\tilde{x} = \frac{a_c \bar{x} + a_x \bar{x}^2 + a_y \bar{x} \bar{y}}{\bar{a}}, \quad (20)$$

$$\tilde{y} = \frac{a_c \bar{y} + a_x \bar{x} \bar{y} + a_y \bar{y}^2}{\bar{a}}, \quad (21)$$

and the components of  $\hat{\mathbf{r}}$  are

$$\hat{x} = \frac{c_1 \bar{x} + c_2 \bar{x}^2 + c_3 \bar{x} \bar{y} + c_4 \bar{x}^3 + c_5 \bar{x}^2 \bar{y} + c_6 \bar{x} \bar{y}^2}{\bar{a} \tilde{h}} \quad (22)$$

$$\hat{y} = \frac{c_1 \bar{y} + c_2 \bar{x} \bar{y} + c_3 \bar{y}^2 + c_4 \bar{x}^2 \bar{y} + c_5 \bar{x} \bar{y}^2 + c_6 \bar{y}^3}{\bar{a} \tilde{h}} \quad (23)$$

where

$$\begin{aligned} c_1 &\equiv a_c h_c, & c_2 &\equiv a_c h_x + a_x h_c, \\ c_3 &\equiv a_c h_y + a_y h_c, & c_4 &\equiv a_x h_x, \\ c_5 &\equiv a_x h_y + a_y h_x, & c_6 &\equiv a_y h_y. \end{aligned}$$

From (16) and (17) the thickness and enthalpy at the cell center are given, respectively, by

$$h_c = \tilde{h} - h_x \tilde{x} - h_y \tilde{y}, \quad (24)$$

$$q_c = \hat{q} - q_x \hat{x} - q_y \hat{y}, \quad (25)$$

where  $h_x$ ,  $h_y$ ,  $q_x$  and  $q_y$  are the limited gradients of thickness and enthalpy. The surface temperature is treated like ice or snow thickness but has no associated enthalpy.

Monotonicity is enforced by Van Leer limiting (Van Leer 1979). That is, the gradients are reduced, if necessary, to ensure that the reconstructed fields contain no values outside the range of the mean values in the cell and its neighbors. Details may be found in DB.

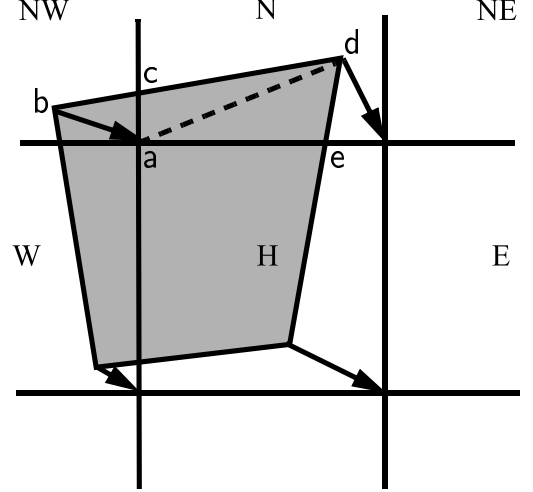


Figure 1: In incremental remapping, conserved quantities are remapped from the shaded departure region, a quadrilateral formed by connecting the backward trajectories from the four cell corners, to the grid cell labeled  $H$ . The region fluxed across the north edge of cell  $H$  consists of a triangle ( $abc$ ) in the  $NW$  cell and a quadrilateral (two triangles,  $acd$  and  $ade$ ) in the  $N$  cell.

### *b. Locating Departure Triangles*

The locating of departure triangles is described in detail by DB and is illustrated in Figures 1–3 and Table 1 of their paper. Here we emphasize the changes made when implementing remapping in CICE.

The basic idea is illustrated in Figure 1, which shows a shaded quadrilateral departure region whose contents are transported to the target or home grid cell, labeled  $H$ . The neighboring grid cells are labeled by compass directions:  $NW$ ,  $N$ ,  $NE$ ,  $W$ , and  $E$ . The four vectors point along the velocity field at the cell corners, and the departure region is formed by joining the starting points of these vectors. Instead of integrating over the entire departure region, it is convenient to compute fluxes across cell edges. We identify departure regions for the north and east edges of each cell, which are also the south and west edges of neighboring cells. Consider the north edge of the home cell, across which there are

fluxes from the neighboring *NW* and *N* cells. The contributing region from the *NW* cell is a triangle with vertices *abc*, and that from the *N* cell is a quadrilateral that can be divided into two triangles with vertices *acd* and *ade*. Focusing on triangle *abc*, we first determine the coordinates of vertices *b* and *c* relative to the cell corner (vertex *a*), using Euclidean geometry to find vertex *c*. Then we translate the three vertices to a coordinate system centered in the *NW* cell. This translation is needed in order to integrate fields (section 3c) in the coordinate system where they have been reconstructed (section 3a). Repeating this process for the north and east edges of each grid cell, we compute the vertices of all the departure triangles associated with each cell edge.

This scheme was designed for rectangular grids. Grid cells in a sea ice model actually lie on the surface of a sphere and must be projected onto a plane. Many such projections are possible. The projection used in CICE, illustrated in Figure 2, approximates spherical grid cells as quadrilaterals in the plane tangent to the sphere at a point inside the cell. The quadrilateral vertices are  $(N/2, E/2)$ ,  $(-N/2, W/2)$ ,  $(-S/2, -W/2)$ , and  $(S/2, -E/2)$ , where *N*, *S*, *E*, and *W* are the lengths of the cell edges on the spherical grid. The quadrilateral area,  $(N + S)(E + W)/4$ , is a good approximation to the true spherical area. However, cell edges in this projection are not orthogonal (i.e., they do not meet at right angles) as on the spherical grid. This means that when vectors are translated from cell corners to cell centers, we must take care that the departure points in the cell-center coordinate system lie inside the grid cell contributing the flux. Otherwise, monotonicity may be violated, because the Van Leer limiting of section 3a does not apply outside the grid cell.

Figure 2 illustrates the difficulty. At the cell center we define orthogonal basis vectors  $\hat{i}$  and  $\hat{j}$  that point toward the midpoints of the cell edges. Similarly, at each cell corner we define a coordinate system whose basis vectors,  $\hat{i}'$  and

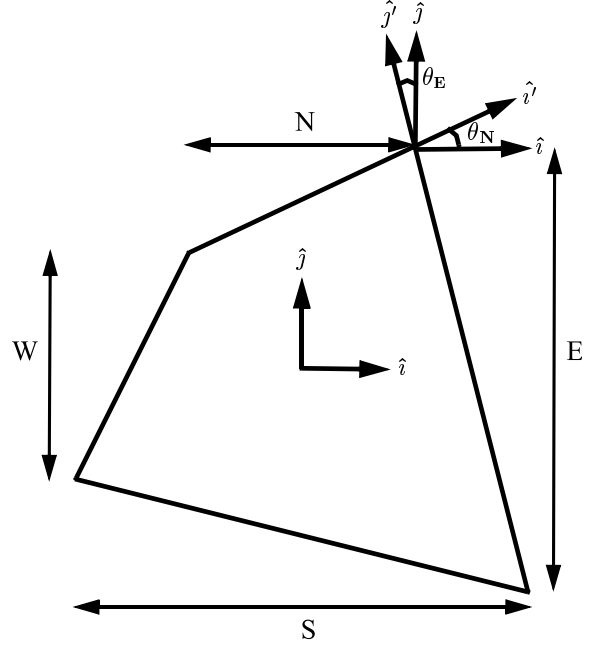


Figure 2: A grid cell on the surface of a sphere with unequal sides of length *N*, *S*, *E*, and *W* is approximated as a quadrilateral lying in the tangent plane at the cell center. The quadrilateral vertices are  $(N/2, E/2)$ ,  $(-N/2, W/2)$ ,  $(-S/2, -W/2)$ , and  $(S/2, -E/2)$ . The basis vectors  $(\hat{i}', \hat{j}')$  at the northeast cell corner have been projected into the cell-center coordinate system and are different from the cell-center basis vectors  $(\hat{i}, \hat{j})$ . The angles  $\theta_N$  and  $\theta_E$  relating the two bases are defined in the text.



$\hat{j}'$ , point along cell edges. The vectors  $\hat{i}'$  and  $\hat{j}'$  are orthogonal in the cell-corner reference frame, but not when projected into the reference frame of the neighboring cell center. For this reason a simple transformation is needed to preserve monotonicity when vectors are translated from corners to centers. Consider a vector  $(x'\hat{i}' + y'\hat{j}')$  in the cell-corner basis. We make the approximation that this vector has the same coordinates when  $\hat{i}'$  and  $\hat{j}'$  are non-orthogonal projections of the cell-corner basis vectors into the cell-center tangent plane, as in Figure 2. Then we transform from the  $(\hat{i}', \hat{j}')$  basis to the  $(\hat{i}, \hat{j})$  basis. In the cell-center coordinate system,  $\hat{i}'$  is obtained by a rotation of  $\hat{i}$  through an angle  $\theta_N$ , where

$$\theta_N = \arctan\left(\frac{E - W}{2N}\right). \quad (26)$$

Similarly,  $\hat{j}'$  is obtained by a rotation of  $\hat{j}$  through  $\theta_E$ , where

$$\theta_E = \arctan\left(\frac{S - N}{2E}\right). \quad (27)$$

Vector are transformed between basis sets using

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \theta_N & -\sin \theta_E \\ \sin \theta_N & \cos \theta_E \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix}, \quad (28)$$

which can be verified by inspection, alternately setting  $x' = 0$  and  $y' = 0$ . Similar transformations are used at the other three cell corners. These transformations guarantee that the grid cell in which a given departure point is located does not change under a change in coordinate systems.

Most grids cells are nearly rectangular, unlike the distorted cell shown in Figure 1. On the  $1^\circ$  displaced-pole grid often used for CICE runs, the maximum angle in (26) and (27) is about  $1^\circ$ . Vector transformations may therefore be omitted on most grids with little loss of accuracy. We have retained them, however, because they ensure exact monotonicity at negligible added cost.

We made one other change in the DB scheme for locating triangles. In their paper, departure

points are defined by projecting cell corner velocities directly backward. That is,

$$\mathbf{x}'_D = -\mathbf{u}' \Delta t, \quad (29)$$

where  $\mathbf{x}'_D$  is the location of the departure point relative to the cell corner and the primes denote vectors defined in the cell-corner basis. This approximation is only first-order accurate in time. The accuracy can be increased to second-order by correcting the velocity with a midpoint approximation before finding the departure point. That is, we first estimate the midpoint of the backward trajectory,  $\mathbf{x}'_M = \mathbf{x}'_D/2$ , then interpolate bilinearly from the four nearest cell corners to estimate the velocity at  $\mathbf{x}'_M$ , and finally use the corrected velocity in (29) to find the departure point. Details may be found in the online CICE documentation.

With this correction the departure points for a linearly varying velocity are nearly exact. Model results are improved significantly for fluid motions in which the CFL number, defined as  $|\mathbf{u}|\Delta t/\Delta x$ , is of order 0.1 or greater. This is true for the test problem below in section 4b, but generally not for real model runs, in which the CFL number typically is of order 0.01 for most grid cells. (The model time step is limited by relatively large velocities in a few grid cells.) Thus the midpoint correction may be omitted for most sea ice simulations, giving about a 3% reduction in the cost of remapping for standard CICE runs.

### c. Integrating Fluxes

Next, we integrate the reconstructed fields over the departure triangles to find the total fluxes of area, volume, and energy across each cell edge. Ice area fluxes are easy to compute since the area is linear in  $x$  and  $y$ . Given a triangle with vertices  $\mathbf{x}_i = (x_i, y_i)$ ,  $i \in \{1, 2, 3\}$ , the triangle area is

$$A_T = \frac{1}{2} |(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)|. \quad (30)$$

The integral  $I_1$  of any linear function  $f(\mathbf{r})$  over a triangle is given by

$$I_1 = A_T f(\mathbf{x}_0), \quad (31)$$

where  $\mathbf{x}_0 = (x_0, y_0)$  is the triangle midpoint,

$$\mathbf{x}_0 = \frac{1}{3} \sum_{i=1}^3 \mathbf{x}_i. \quad (32)$$

To compute the area flux we evaluate the area at the midpoint,

$$a(\mathbf{x}_0) = a_c + a_x x_0 + a_y y_0, \quad (33)$$

and multiply by  $A_T$ . By convention, northward and eastward fluxes are positive, while southward and westward fluxes are negative.

Equation (31) cannot be used for volume fluxes, because the reconstructed volumes are quadratic functions of position. (They are products of two linear functions, area and thickness.) The integral of a quadratic polynomial over a triangle requires function evaluations at three points,

$$I_2 = \frac{A_T}{3} \sum_{i=1}^3 f(\mathbf{x}'_i), \quad (34)$$

where  $\mathbf{x}'_i = (\mathbf{x}_0 + \mathbf{x}_i)/2$  are points lying halfway between the midpoint and the three vertices. DB use this formula to compute fluxes of the tracer mass, which is analogous to ice volume. Equation (34) does not work for ice and snow energies, which are cubic functions—products of area, thickness, and enthalpy. Integrals of a cubic polynomial over a triangle can be evaluated using a four-point formula (Stroud 1971):

$$I_3 = A_T \left[ -\frac{9}{16} f(\mathbf{x}_0) + \frac{25}{48} \sum_{i=1}^3 f(\mathbf{x}''_i) \right] \quad (35)$$

where  $\mathbf{x}''_i = (3\mathbf{x}_0 + 2\mathbf{x}_i)/5$ .

To evaluate functions at specific points, we must compute many products of the form  $a(\mathbf{x})h(\mathbf{x})$  and  $a(\mathbf{x})h(\mathbf{x})q(\mathbf{x})$ , where each term in the product is the sum of a cell-center value and two displacement terms. This computation can be sped up by storing and reusing terms that appear in the expressions for more than one flux.

#### d. Updating State Variables

Finally, we use the fluxes to compute new values of the state variables in each ice category and grid cell. The new fractional ice area  $a_{\text{new}}$  in grid cell  $(i, j)$  is given by

$$a_{\text{new}}(i, j) = a(i, j) + \frac{F_E(i-1, j) - F_E(i, j) + F_N(i, j-1) - F_N(i, j)}{A(i, j)} \quad (36)$$

where  $F_E(i, j)$  and  $F_N(i, j)$  are area fluxes across the east and north edges, respectively, of cell  $(i, j)$ , and  $A(i, j)$  is the grid cell area. Since all fluxes added to one cell are subtracted from a neighboring cell, (36) conserves global ice area.

The new ice volumes and energies are computed analogously. New thicknesses are given by the ratio of volume to area, and new enthalpies by the ratio of energy to volume. Compatibility is ensured because the new-time thickness  $h_{\text{new}}$  and enthalpy  $q_{\text{new}}$  are given by

$$h_{\text{new}} = \frac{\int_A a h dA}{\int_A a dA}, \quad (37)$$

$$q_{\text{new}} = \frac{\int_A a h q dA}{\int_A a h dA}, \quad (38)$$

where the terms in the numerators and denominators on the right are integrals of the old-time area, volume, and energy over a Lagrangian departure region with area  $A$ . That is, the new-time thickness and enthalpy are weighted averages of old-time values with non-negative weights  $a$  and  $ah$ , respectively. Thus the new-time values must lie between the maximum and minimum of the old-time values.

## 4. Test problems with exact solutions

We now apply the incremental remapping scheme to three test problems with known solutions. Each test problem is an initial-boundary value problem of the form

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (39)$$

$$\rho(x, y, 0) = \rho_0(x, y), \quad (40)$$

with specified initial conditions on a square domain. We first consider uniform advection of a square mesa in a straight line, both along and diagonal to the grid lines. We next examine solid-body rotation of a cylinder, a test used by DB. Finally, we perform a compatibility test with a convergent velocity field, as suggested by Schär and Smolarkiewicz (1996).

For comparison, we also apply the first-order upwind and MPDATA schemes. The MPDATA scheme tested here is the one used in earlier versions of CICE; it is the basic scheme of Smolarkiewicz (1984) with an upwind step followed by three corrective iterations. The upwind scheme is equivalent to MPDATA without any corrective iterations. There are other potential schemes we did not test: for example, the linear upstream and second-order moment schemes and the enhanced versions of MPDATA. Some of these schemes would likely perform at least as well as remapping in the following tests. However, these schemes probably would be no less expensive than the basic MPDATA, which already is undesirably expensive.

#### *a. Uniform advection in a straight line*

First, we study uniform advection in a straight line on a square grid. The grid has dimensions  $128 \times 128$  with cells of unit length and width. We advect a square mesa that has initial height  $h = 1$  and is surrounded by grid cells with  $h = 0$ . The lower left corner of the mesa is located at  $(x, y) = (20, 20)$ , and the sides have length  $L = 10$  or  $20$ . The velocity field is directed either eastward along the  $x$  axis,  $\mathbf{u} = (1, 0)$ , or northeastward at a 45-degree angle to the  $x$  axis,  $\mathbf{u} = (1, 1)$ . The model is stepped forward 72 time units. At  $t = 72$  the exact solution is a square mesa displaced by 72 units in the  $x$  direction and by 0 or 72 units in the  $y$  direction, depending on whether the velocity is eastward or northeastward. We apply each transport scheme

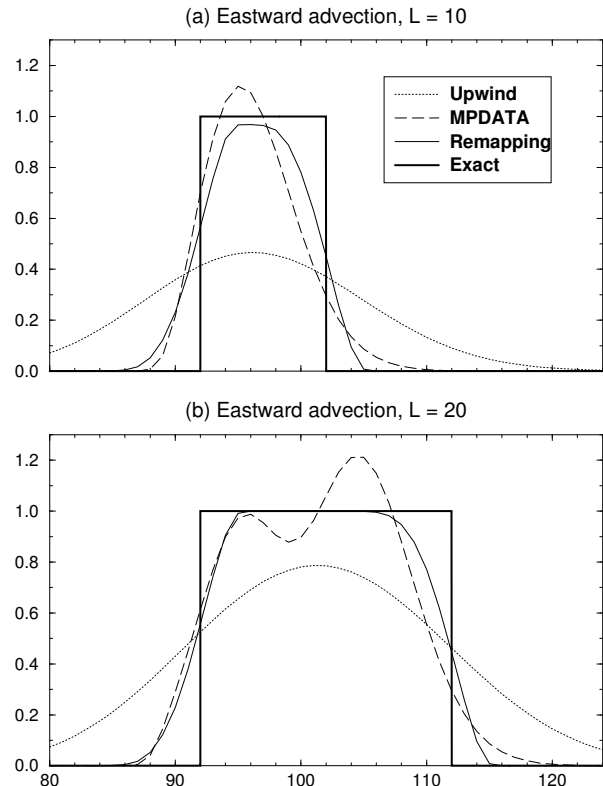


Figure 3: Cross section of the height at time  $t = 72$  of an initially square mesa with height  $h = 1.0$ , advected eastward with uniform velocity (CFL number = 0.1). The initial mesa has sides of length (a)  $L = 10$  and (b)  $L = 20$ . The upwind scheme is highly diffusive, and MPDATA violates monotonicity. The remapping solution is the best approximation to the exact solution.

and compare the resulting numerical solutions to the exact solution.

Figure 3 shows cross sections of the numerical solutions at  $t = 72$  for the case of eastward advection with a CFL number given by  $|\mathbf{u}|\Delta t/\Delta x = 0.1$  (i.e.,  $\Delta t = 0.1$ ), for  $L = 10$  and  $L = 20$ . In each plot the remapping solution is closest to the exact solution, while the upwind plots are very diffuse and the MPDATA plots have overshoots. Table 1 gives the peak values of the numerical solutions. The rms errors for  $L = 10$  ( $L = 20$ ) are 0.027 (0.036) for remapping, 0.027 (0.043) for MPDATA, and

0.052 (0.070) for upwind. The remapping and upwind solutions improve as the mesa width increases. With  $L = 20$  remapping gives virtually no peak clipping over a width of about 10 grid cells. Surprisingly, the MPDATA solution does not improve as  $L$  increases. With  $L = 20$  the overshoot is larger than with  $L = 10$ , and the solution is bimodal. When the velocity field is at a 45-degree angle to the grid lines, the solutions are qualitatively similar, though with more peak clipping for remapping and upwind and with larger overshoots for MPDATA.

If the CFL number is increased the remapping and upwind solutions become less diffuse. With CFL number 0.9 (i.e.,  $\Delta t = 0.9$ ), all three schemes are stable for eastward advection, but the MPDATA and upwind schemes are unstable for northeastward advection. The upwind scheme is unstable for diagonal flow with CFL numbers larger than 0.5 because fluxes across the east and north cell edges are computed independently. If each flux includes more than half the grid cell, the height can become negative. MPDATA has a more stringent CFL limit than the upwind scheme (Smolarkiewicz 1984); for this test problem the solution is unstable for northeastward flow with CFL numbers greater than 0.39. Remapping is stable for CFL numbers up to 1.0, regardless of the flow direction, because it is fully two-dimensional. In a real sea ice model the CFL number in most grid cells is small, and stability is less of a concern than spatial accuracy. In grid cells near the ice edge, however, the velocities may be relatively large, in which case remapping allows a longer time step than the other two schemes. The upwind scheme could, however, be run with the same time step as remapping if the fields were updated twice per time step, first in the east-west direction and then in the north-south direction.

### b. Solid body rotation

Next, we perform the rotating cylinder test found in DB. The grid is the same as in sec-

tion 4a, and the velocity field produces uniform clockwise rotation about the center of the grid. The speed is 1.0 at the midpoints of the sides of the grid, and  $\Delta t$  is chosen so that the rotation period is 1000 time steps, giving a maximum CFL number of about 0.4. The initial condition is a cylinder of radius  $r = 10$  and height  $h = 1$ , centered 42 units north of the center of rotation. The surrounding grid cells initially have  $h = 0$ .

Figure 4 shows height contours after one rotation. Since the flow is nondivergent, a perfect numerical scheme would preserve the shape of the cylinder and return it exactly to its starting location after 1000 steps, as in Figure 4a. Again, remapping gives the closest approximation to the exact solution. It maintains a peak height of 0.999 and does the best job of preserving the cylinder shape. The upwind solution is very diffuse, with a peak height of just 0.317. MPDATA has numerical overshoots, giving a peak height of 1.317. The rms errors are 0.047 for remapping, 0.053 for MPDATA, and 0.110 for upwind.

### c. Compatibility test

The first two tests analyzed the transport of a height field, analogous to sea ice area, focusing on two properties: spatial accuracy and monotonicity. We now turn our attention to tracer transport and a third desirable property, compatibility, following Schär and Smolarkiewicz (1996). We have transport equations for area and volume,

$$\frac{\partial a}{\partial t} + \nabla \cdot (a\mathbf{u}) = 0, \quad (41)$$

$$\frac{\partial v}{\partial t} + \nabla \cdot (v\mathbf{u}) = 0, \quad (42)$$

which together imply advection of thickness:

$$\frac{\partial h}{\partial t} + \mathbf{u} \cdot \nabla h = 0, \quad (43)$$

where  $h = v/a$ . A compatible numerical scheme is one that preserves the monotonicity of  $h$ .

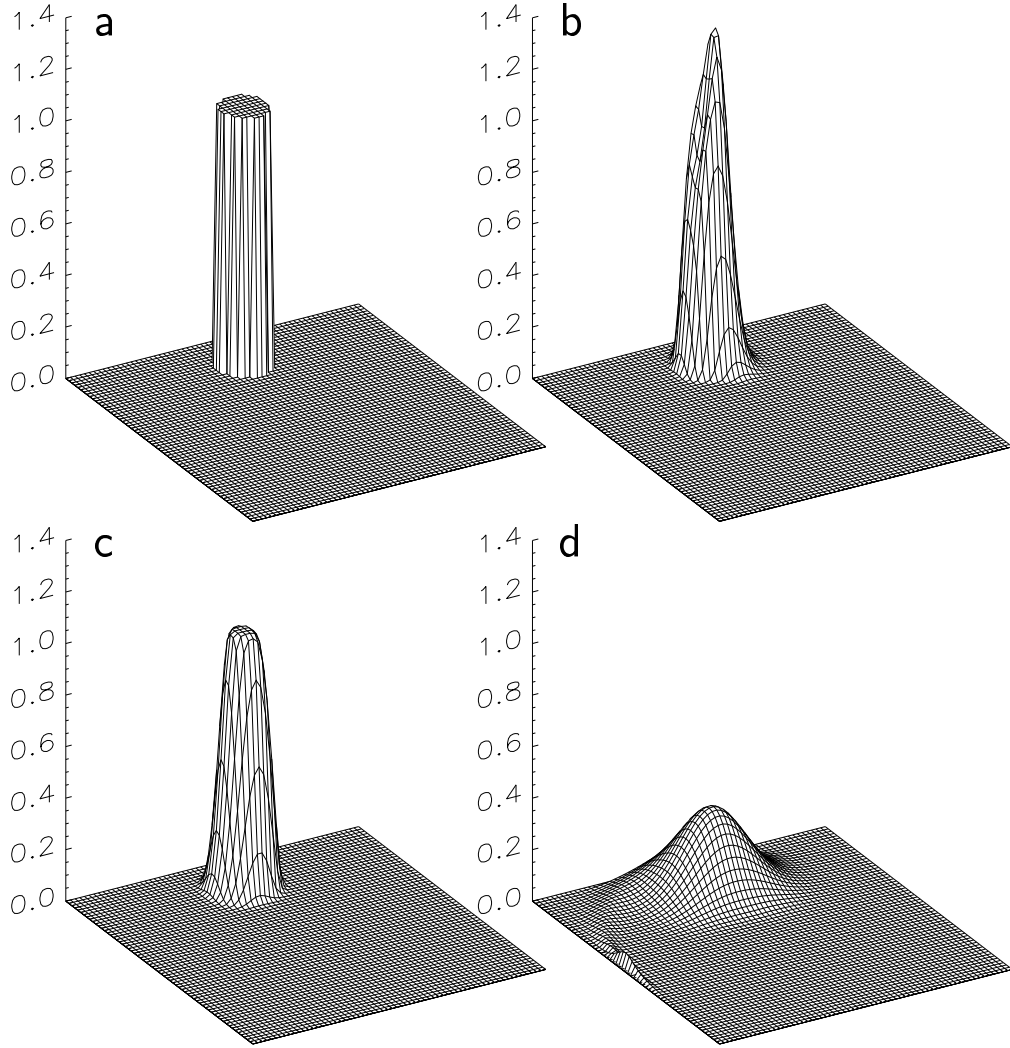


Figure 4: Contours of fractional ice area of a cylinder after one solid-body rotation: (a) initial condition ( $\max = 1.0$ ), (b) MPDATA ( $\max = 1.317$ ), (c) remapping ( $\max = 0.999$ ), (d) upwind ( $\max = 0.317$ ). Remapping preserves monotonicity, unlike MPDATA, and is less diffusive than the upwind scheme.

The prescribed velocity field is one-dimensional and convergent:  $\mathbf{u}(x, y) = (-x, 0)$ . For arbitrary initial conditions  $a_0(x)$  and  $v_0(x)$ , the solution at time  $t$  is

$$a(x, t) = e^t a_0(xe^t), \quad (44)$$

$$v(x, t) = e^t v_0(xe^t), \quad (45)$$

$$h(x, t) = h_0(xe^t). \quad (46)$$

The initial conditions for this test problem, as shown in Figure 5, are

$$a_0(x) = \begin{cases} 1+x & \text{for } -1 \leq x \leq 0, \\ 1 & \text{for } 0 \leq x \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad (47)$$

$$h_0(x) = \begin{cases} 1 & \text{for } |x| \geq 0.75, \\ 0.2 & \text{otherwise.} \end{cases} \quad (48)$$

The grid spacing is  $\Delta x = 0.05$  and the time step is  $\Delta t = 0.025$ , giving a CFL number of 0.5 at the edge of the ice-covered domain. The model is integrated for 40 steps to  $t = 1$ .

Figure 6 shows the final distributions of area, volume, and thickness for the three schemes. This figure corresponds to Figure 2 in Schär and Smolarkiewicz (1996). The solutions are scaled (the abscissa is  $xe^t$ , and the area and volume are divided by  $e^t$ ) so that if the numerical scheme were perfect, the distributions would not change in time. As in previous tests the upwind solutions are the most diffuse; the area plateau erodes, and the peak volume decreases from 1.0 to 0.460 (Figure 6a). However, the upwind scheme is compatible (Figure 6b). At first glance the area and volume plots for MPDATA and remapping (Figures 6c and 6e) look similar. MPDATA, however, gives a larger volume-to-area ratio near  $xe^t = -1$  and  $xe^t = 1$ , resulting in spurious extrema in  $h$  (Figure 6d); the maximum thickness is  $h = 2.566$  at  $xe^t = -1.16$ . These results are found with three corrective iterations per time step. With fewer iterations the overshoot is smaller—for example, the maximum thickness is 1.25 with a single corrective iteration—but the solution is more diffuse. As

in the other two test problems, remapping gives the best overall results. Not only does it satisfy compatibility (Figure 6f), but it is less diffusive than the upwind scheme. The rms errors in area (volume), based on the difference between the scaled model solutions and the initial conditions, are 0.065 (0.059) for remapping, 0.065 (0.063) for MPDATA, and 0.078 (0.071) for upwind. We obtained similar results with a smaller time step.

## 5. Tests in a global sea ice model

### a. Model Results

To study differences among transport schemes in global sea ice models, we ran three 10-year simulations of the NCAR CCSM to a quasi-equilibrium state. CSIM, the sea ice component of CCSM, is physically very similar to CICE. We ran the model in the “M” configuration, using atmospheric forcing data based on 1979-1988 observations and ocean forcing data from a fully coupled CCSM run. The atmospheric forcing fields were daily 10 m temperature, wind velocity, specific humidity, and air density from the National Centers for Environmental Prediction (NCEP)/NCAR reanalysis project (Kalnay et al. 1996), monthly downward shortwave radiation and cloud fraction from the European Centre for Medium-Range Weather Forecasts (ECMWF) reanalysis project (Gibson et al. 1997), and monthly precipitation from the Xie-Arkin data set (Xie and Arkin 1996) with improvements for the Arctic (Serreze and Hurst 2000). These fields were interpolated to the model grid, a displaced-pole grid with a horizontal resolution of approximately  $1^\circ$  (Figure 7). The ice model includes a simple slab ocean mixed layer model to improve the surface energy budget. The initial ice conditions were derived from a previous model run.

Figure 8 shows the annual cycles of total

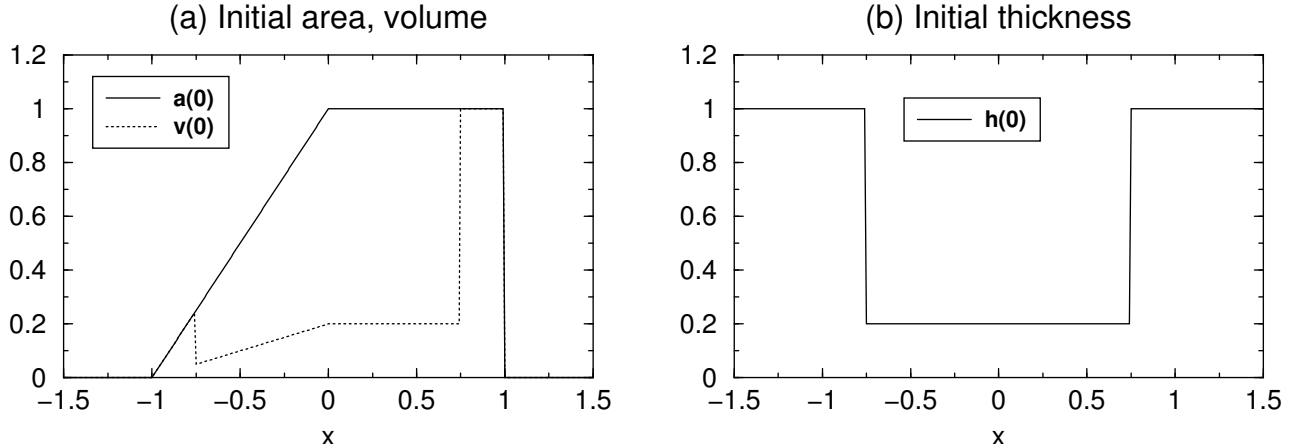


Figure 5: Initial conditions for the compatibility test problem: (a) area and volume, (b) thickness.

ice area and volume in the Arctic and Antarctic during the tenth year of the simulation. These cycles agree fairly well with observations, given the uncertainties in the forcing data. As observed, the model produces thick ( $> 2$  m) perennial ice in the central Arctic and thinner ( $< 1$  m) seasonal ice in the peripheral Arctic seas and in much of the Antarctic. Satellite measurements (Cavalieri et al. 1997) suggest that the model somewhat underestimates the extent of Antarctic ice, which ranges from about  $0.4 \times 10^7$  km<sup>2</sup> in summer to  $1.8 \times 10^7$  km<sup>2</sup> in winter, and also underestimates the seasonal change in Arctic ice extent, which varies from about  $0.7 \times 10^7$  km<sup>2</sup> in summer to  $1.5 \times 10^7$  km<sup>2</sup> in winter. Remapping and MPDATA produce nearly identical cycles of ice area and volume. The upwind scheme gives about the same area as the other two schemes, except during the Antarctic summer, but yields less volume in both hemispheres throughout the year. The remapping and MPDATA volumes exceed the upwind volumes by  $0.07 - 0.11 \times 10^4$  km<sup>3</sup> in the Arctic and by  $0.08 - 0.13 \times 10^4$  km<sup>3</sup> in the Antarctic during the last five years of the simulation. These differences represent 2-3% of the annual mean simulated ice volume in the Arctic and 10-15% in the Antarctic.

Figure 9 compares the mean simulated ice thickness (including open water) during Decem-

ber of year 10 for each of the three transport schemes. The remapping-MPDATA differences (Figures 9e and 9f) are small, generally  $< 0.25$  m, with an average difference near zero. The remapping-upwind differences are positive, on average, and in some regions exceed 0.5 m (Figures 9c and 9d). The differences are largest where the modeled ice is thickest, along the Canadian and Greenland coasts in the Arctic and in the Ross and Weddell Seas in the Antarctic. The differences in ice area between upwind and the other two schemes (not shown) are much less pronounced than the volume differences.

These results are consistent with the large diffusiveness of upwind transport. Compared to the other two schemes, the upwind scheme tends to spread the ice over a larger area. The ice extent, however, is highly constrained by the ocean mixed layer temperature, which depends mainly on the atmosphere and ocean forcing and is not strongly influenced by the transport scheme. For this reason, much of the ice diffused equatorward by the upwind scheme quickly melts. The ice volume is reduced, but the ice area is nearly unchanged.

The remapping and MPDATA results are much more similar for the realistic model runs than for the test problems of section 4. The similarity can be explained by noting that the modeled fields are much smoother than the dis-

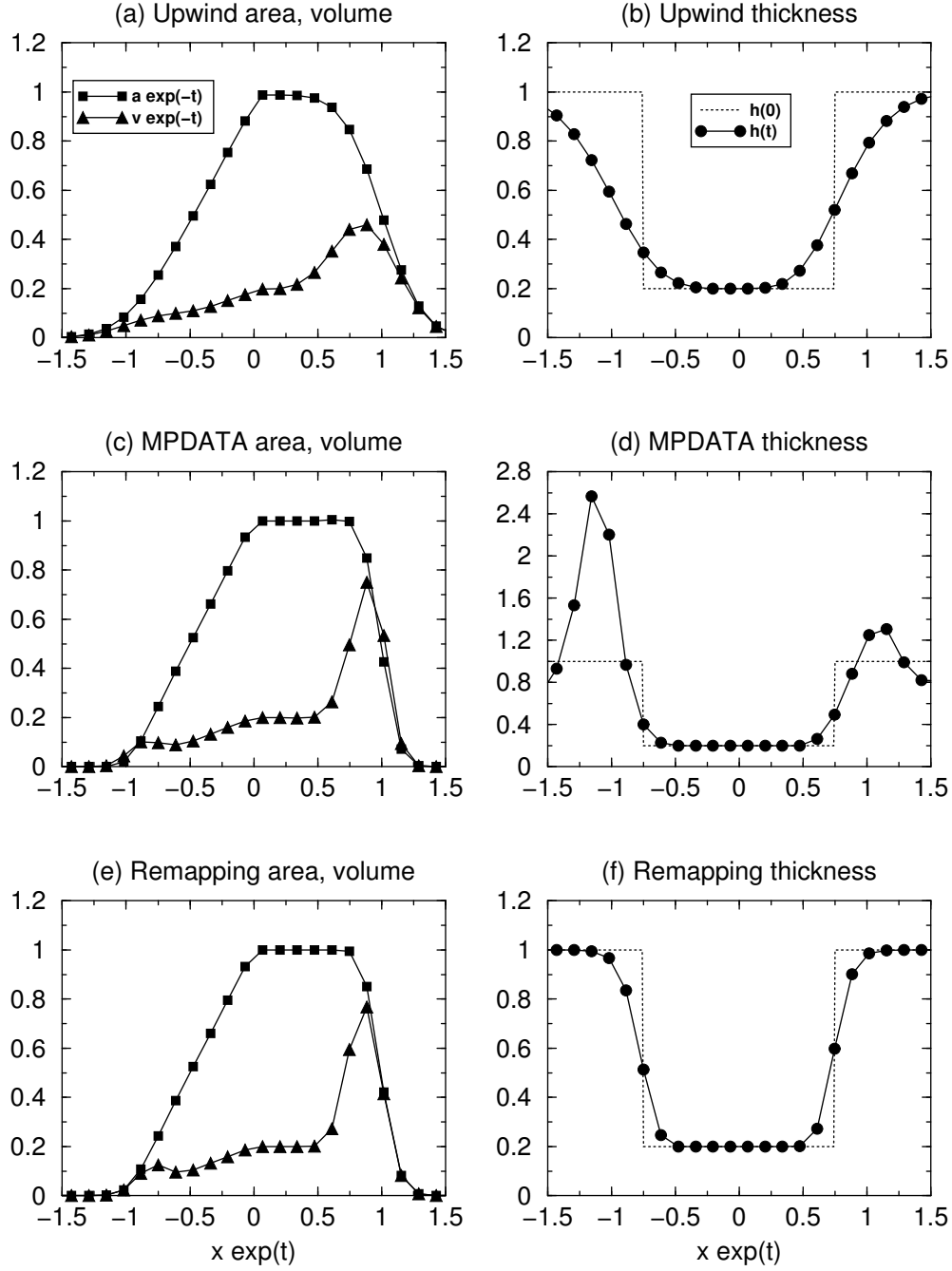


Figure 6: Numerical solutions given by the upwind, MPDATA, and remapping schemes at  $t = 1$  for the compatibility test problem. The scales are stretched so that a perfect numerical scheme would yield plots identical to the initial conditions in Figure 5. Note the different vertical scale in (d).



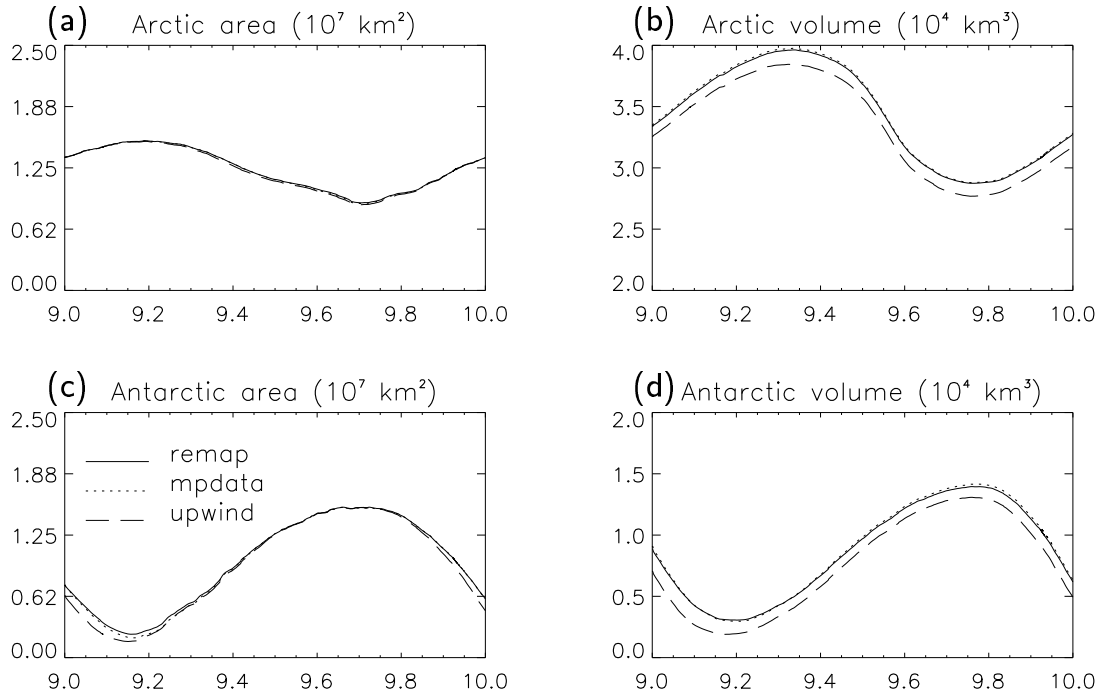


Figure 8: Total ice area and volume in the Arctic and Antarctic in the tenth year of global sea ice simulations: remapping (solid line), MPDATA (dotted line), and upwind (dashed line).

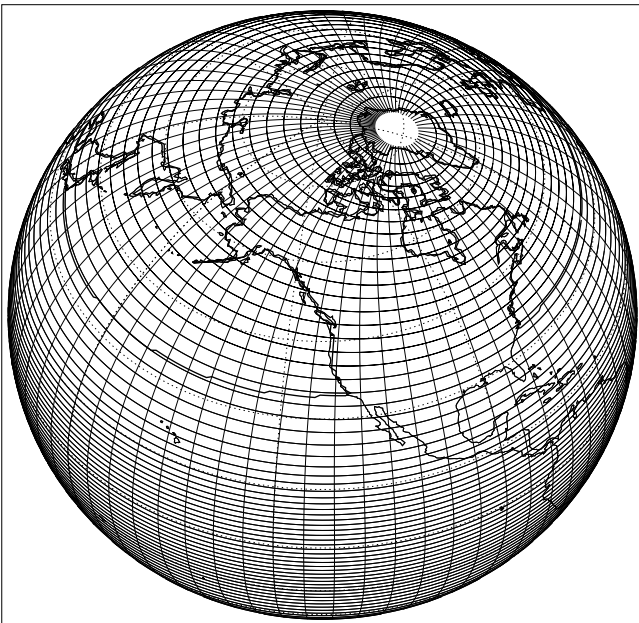


Figure 7:  $320 \times 384$  displaced-pole grid used for global sea ice simulations. The grid resolution is approximately  $1^\circ$ ; every fifth grid line is plotted.

continuous fields in the test cases. Also, negative feedbacks in the model reduce the effects of numerical overshoots. For example, a positive overshoot in the ice area produces additional ridging, which strengthens the ice and reduces the likelihood of further convergence and overshoots. Nevertheless, numerical overshoots can cause problems in model simulations. In CCSM simulations using MPDATA it is possible for a succession of enthalpy overshoots in a grid cell to drive internal ice temperatures below absolute zero. The other major disadvantage of MPDATA compared to remapping is its cost, as discussed in the next section.

### b. Model Performance

We evaluated the performance of the three transport schemes on several model grids. Here we present results using CICE on the  $1^\circ$  grid shown in Figure 7. Each scheme was optimized to the best of our abilities. The model was

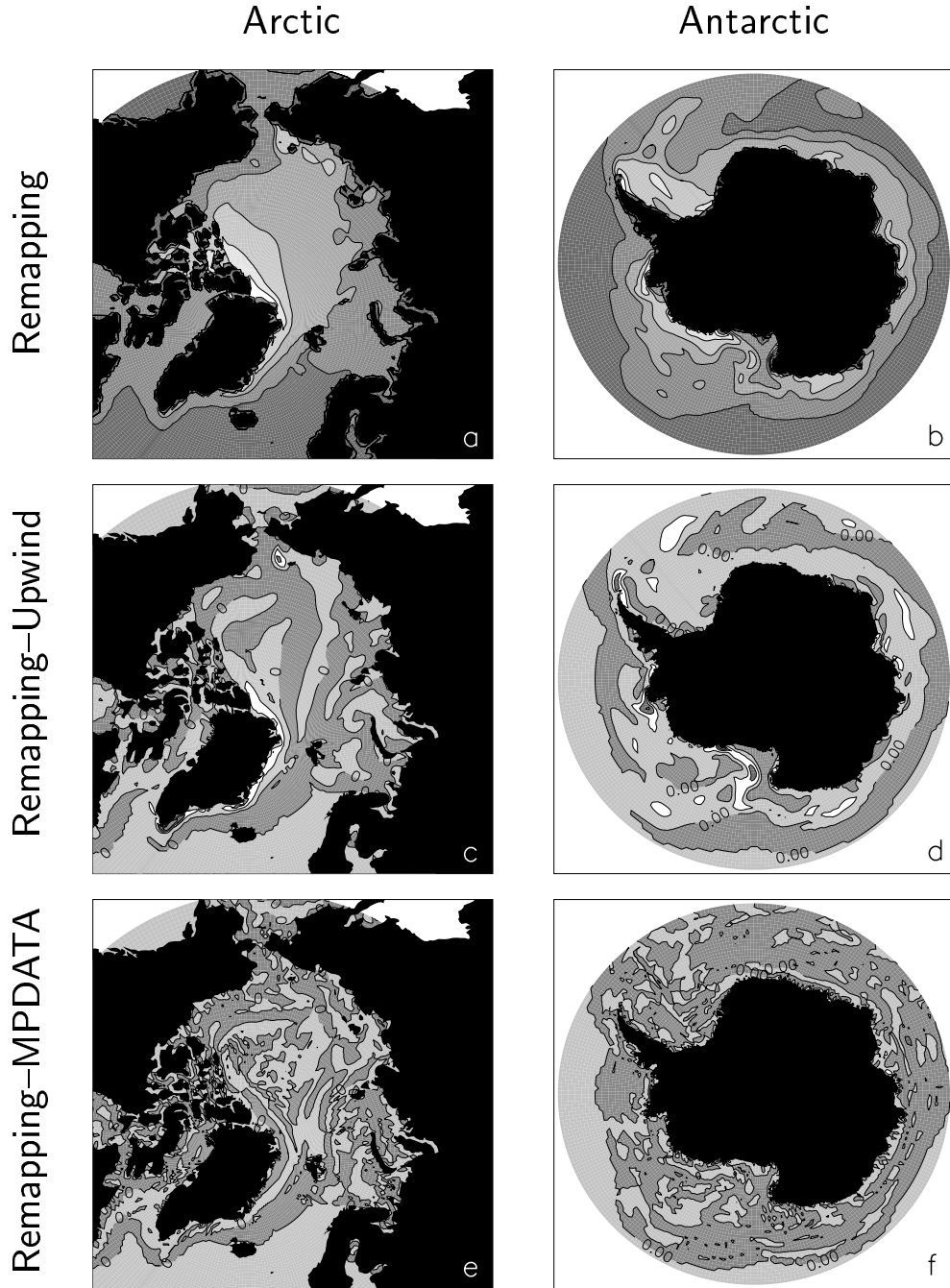


Figure 9: Mean ice thickness in December of year 10 for global sea ice simulations. Panels (a) and (b) show the ice thickness using the remapping scheme, with lighter shades corresponding to thicker ice. Panels (c)-(f) show thickness differences; negative values are shaded dark grey, positive values light grey, and large positive values white. The contour intervals are 2 m for (a), 1 m for (b) and (c), 0.5 m for (d), and 0.25 m for (e) and (f).

run in ice-only mode with a simple mixed layer using atmospheric forcing similar to that described in section 5a, but without ocean forcing. (These differences do not significantly affect the time required for ice transport.) The model was parallelized using the Message Passing Interface (MPI) and run on an SGI Origin 3000 with 4, 8, 16, 32, and 64 processors. We first spun up the model with ten years of forcing data, then ran for 720 time steps (30 days) using the remapping, MPDATA, and upwind schemes. For each processor number and transport scheme, we ran the model twice and averaged the times, which are reproducible to within about 1%.

Figure 10 shows the time required for ice transport using each scheme. The cost decreases almost linearly as the number of processors increases. As expected, the upwind scheme is least expensive, four to five times cheaper than remapping. MPDATA, on the other hand, is about twice as expensive as remapping. The ratio of the MPDATA to remapping cost ranges from 1.87 for 8 processors to 2.35 for 64 processors. MPDATA accounts for 26-38% of the total run time for the ice model, compared to 15-24% for remapping and 3.7-7.3% for upwind. The relative cost of transport goes down as the number of processors goes up, since all three schemes parallelize more efficiently than the model as a whole. On a coarser  $3^\circ$  grid (results not shown), the costs of the three schemes relative to each other and to the entire model are similar to those on the  $1^\circ$  grid.

These times are for the standard model configuration with five ice thickness categories, each with nine conserved quantities. In simple models with just one or two equations, MPDATA is faster than remapping, but remapping is faster when there are more than about five transported fields. There are two reasons that remapping becomes more efficient as the number of categories and tracers is increased. First, the most complex part of the algorithm is locating triangle vertices (section 3b). Since this computation is done only once per time step, its cost per field

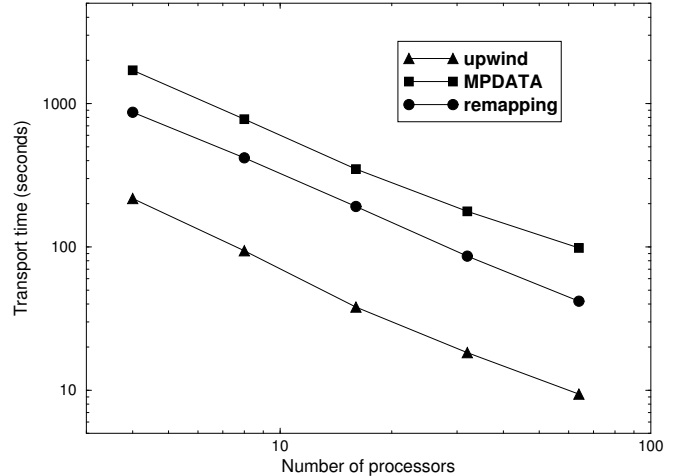


Figure 10: Cost in seconds for the three transport schemes, running CICE on a  $1^\circ$  grid for 720 time steps. All three schemes scale almost linearly with processor number from 4 to 64 processors. Remapping is four to five times slower than the upwind scheme but about twice as fast as MPDATA.

decreases as the number of fields increases. Second, many of the quantities needed to integrate fluxes (section 3c), such as sums over triangle points of  $a$ ,  $ax$ , and  $ay$ , can be reused for additional tracers.

Table 2 illustrates the marginal costs of additional fields for the three schemes. The model was first spun up on the  $1^\circ$  grid for one year, then timed for a 30-day run with 32 processors using each transport scheme. In the control run we transported the standard number of fields: one area, three thickness-type fields (including surface temperature), and five enthalpies for each of five ice categories. In the first sensitivity run we transported a sixth ice category with the standard number of tracers. In the second run five categories were transported, but with an additional thickness-like tracer in each category. The third run is like the second, except that the added tracer is an enthalpy instead of a thickness. The marginal costs are reproducible only to within about 10%, since they are obtained by subtracting two larger numbers. For

each scheme an extra thickness category, corresponding to nine new equations, is roughly twice as expensive as an extra tracer, which requires five new equations. MPDATA is about twice as expensive as remapping overall, but the cost of a new category or tracer is almost three times greater.

We did not test the linear upstream and SOM schemes, but these schemes probably are comparable in cost to MPDATA. For the two-dimensional rotating cylinder test, SOM is about 12 times slower than upwind transport (W. J. Merryfield 2003, personal communication). We found MPDATA to be about 10 times slower than upwind in both the rotating cylinder test and in global simulations, whereas remapping is 12 times slower than upwind for the rotating cylinder test but only 4 times slower in a global sea ice model. SOM, like MPDATA, scales linearly with the number of equations. It would be competitive with remapping in models with a small number of transport equations, but not in CICE.

## 6. Conclusions

The tests in sections 4 and 5 suggest that incremental remapping is an ideal scheme for solving the transport equation in sea ice models with multiple thickness categories and vertical layers. Remapping satisfies several important criteria: it is monotonicity-preserving, compatible, second-order accurate (except where gradients are limited to preserve monotonicity), and efficient. Other schemes used to model sea ice transport fail to meet at least one of these criteria.

The upwind scheme has several desirable properties: it is monotonicity-preserving, compatible, simple, and fast. On the other hand, it is very diffusive, as illustrated by the test problems in section 4. In global simulations, upwind transport thins the ice pack by diffusing ice equatorward into warm water where it melts. However, the differences between remap-

ping and upwind are fairly small compared to other model uncertainties. These differences might be larger in coupled models with more feedbacks present. On finer grids, remapping would be better than upwind at resolving small-scale features because of its greater accuracy.

MPDATA, like remapping, is second-order accurate. In global simulations it produces ice area and volume fields similar to those given by remapping. Nonetheless, the version of MPDATA we tested has two important disadvantages. First, it takes roughly twice as long as remapping to solve the 46 transport equations per time step in the standard version of CICE, and three times as long for each additional field. Second, this version of MPDATA does not preserve the monotonicity of conserved fields and tracers. Numerical overshoots in these fields reduce the realism of the model and can lead to pathologies such as unphysically low temperatures. Enhanced monotonicity-preserving versions of MPDATA would be more realistic and robust, but also more expensive.

The linear upstream and second-order moment schemes would provide some of the same advantages as remapping. They are spatially accurate (especially SOM, which is third-order accurate) and can be limited to preserve monotonicity for conserved fields. These schemes do not, however, satisfy compatibility, and they require storage and transport of higher-order moments. Also, the cost of these schemes scales linearly with the number of equations. Although these schemes might be faster than remapping in models with a small number of transported fields, they would be less efficient in models such as CICE with many transport equations.

Incremental remapping should have useful applications apart from sea ice transport. For example, it may prove cost-effective in ocean models with many tracers. One of us (WHL) recently implemented a remapping transport scheme in a shallow-water model on the spherical geodesic grid developed at Colorado State University (Randall et al. 2002). This scheme

Table 1: Peak solution values at  $t = 72$  for uniform advection of an initially square mesa of length  $L$  and unit height. The exact solution retains a height of 1.0.

CFL no.	direction	$L$	remapping	upwind	MPDATA
0.1	east	10	0.968	0.466	1.118
0.1	east	20	1.000	0.786	1.211
0.1	northeast	10	0.940	0.218	1.227
0.1	northeast	20	1.000	0.619	1.467
0.9	east	10	1.000	0.938	1.193
0.9	east	20	1.000	1.000	1.193
0.9	northeast	10	1.000	Unstable	Unstable
0.9	northeast	20	1.000	Unstable	Unstable

Table 2: Cost in seconds for additional transport equations using the three transport schemes in the Los Alamos sea ice model, CICE. Runs were performed on the  $1^\circ$  grid for 30 model days with 32 processors.

Configuration	remapping	upwind	MPDATA
control	85.6	18.2	176.4
+1 category	13.9	4.4	38.5
+1 thickness	7.2	1.8	20.2
+1 enthalpy	7.1	2.0	20.7

will later be tested in an isopycnic ocean GCM on a geodesic grid. Also, it may be possible to combine the 2D scheme described here with a vertical remapping scheme to make a 3D scheme suitable for tracer transport in  $z$ -level ocean models. These efforts will be reported in future publications.

## Acknowledgments

We could not have written this paper without the inspiration and guidance of John Dukowicz and John Baumgardner. We are grateful to Julie Schramm for assistance in implementing and running the remapping scheme in CCSM. We also thank Bruce Briegleb, Greg Flato, Bill Merryfield, Gary Russell, and an anonymous reviewer for helpful discussions and comments. This work was supported by the Department of Energy Climate Change Prediction Program (CCPP) and Scientific Discovery through Advanced Computing (SciDAC) program.

## References

- Arakawa, A. and V. R. Lamb, 1977: Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics*, J. Chang, Ed., Vol. 17, Academic Press, 173–265.
- Bitz, C. M., M. M. Holland, A. J. Weaver, and M. Eby, 2001: Simulating the ice-thickness distribution in a coupled climate model. *J. Geophys. Res.*, **106**, 2441–2463.
- Bitz, C. M. and W. H. Lipscomb, 1999: An energy-conserving thermodynamic sea ice model for climate study. *J. Geophys. Res.*, **104**, 15,669–15,677.
- Briegleb, B. P., C. M. Bitz, E. C. Hunke, W. H. Lipscomb, and J. L. Schramm, cited 2003: Description of the Community Climate System Model Version 2 Sea Ice Model
- Cattle, H. and J. Crossley, 1995: Modelling Arctic climate change. *Phil. Trans. R. Soc. Lond. A*, **352**, 201–213.
- Cavalieri, D. J., P. Gloersen, C. L. Parkinson, J. C. Comiso, and H. J. Zwally, 1997: Observed hemispheric asymmetry in global sea ice changes. *Science*, **278**, 1104–1106.
- Dukowicz, J. K. and J. R. Baumgardner, 2000: Incremental remapping as a transport/advection algorithm. *J. Comput. Phys.*, **160**, 318–335.
- Gibson, J. K., P. Kalberg, S. Uppala, A. Nomura, A. Hernandez, and E. Serrano, 1997: ERA description. Technical Report series 1, ECMWF Re-Analysis Project, 72 pp.
- Hibler, W. D., 1979: A dynamic thermodynamic sea ice model. *J. Phys. Oceanogr.*, **9**, 817–846.
- Hunke, E. C., 1998: CICE: The Los Alamos Sea Ice Model, Documentation and Software. LA-CC-98-16, Los Alamos National Laboratory, Los Alamos, New Mexico.
- Hunke, E. C. and J. K. Dukowicz, 1997: An elastic-viscous-plastic model for sea ice dynamics. *J. Phys. Oceanogr.*, **27**, 1849–1867.
- Hunke, E. C. and W. H. Lipscomb, 2001: CICE: the Los Alamos Sea Ice Model, Documentation and Software, version 3. LA-CC-98-16 v.3, Los Alamos National Laboratory, Los Alamos, New Mexico.
- LeVeque, R. J., 1992: *Numerical Methods for Conservation Laws*. Birkhauser Verlag, 214 pp.
- Lipscomb, W. H., 2001: Remapping the thickness distribution in sea ice models. *J. Geophys. Res.*, **106**, 13,989–14,000.

- Margolin, L. and P. K. Smolarkiewicz, 1998: Antidiffusive velocities for multipass donor cell advection. *SIAM J. Sci. Comput.*, **20**, 907–929.
- Maykut, G. A. and N. Untersteiner, 1971: Some results from a time dependent thermodynamic model of sea ice. *J. Geophys. Res.*, **76**, 1550–1575.
- Merryfield, W. J. and G. Holloway, 2003: Application of an accurate advection algorithm to sea-ice modelling. *Ocean Modelling*, **5**, 1–15.
- Prather, M. J., 1986: Numerical advection by conservation of second-order moments. *J. Geophys. Res.*, **91**, 6671–6681.
- Randall, D. A., T. D. Ringler, R. Heikes, P. Jones, and J. Baumgardner, 2002: Climate modeling with spherical geodesic grids. *Computing in Science and Engineering*, **4**, 32–41.
- Russell, G. L. and J. A. Lerner, 1981: A new finite-difference scheme for the tracer transport equation. *J. Appl. Meteorol.*, **12**, 1483–1498.
- Schär, C. and P. K. Smolarkiewicz, 1996: A synchronous and iterative flux-correction formalism for coupled transport. *J. Comput. Phys.*, **128**, 101–120.
- Semtner, A. J., 1986: Finite-difference formulation of a world ocean model. *Advanced Physical Oceanography Numerical Modeling*, J. J. O’Brien, Ed., D. Reidel, 187–202.
- Serreze, M. C. and C. M. Hurst, 2000: Representation of mean Arctic precipitation from NCEP-NCAR and ERA reanalysis. *J. Climate*, **13**, 182–201.
- Smolarkiewicz, P. K., 1984: A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. *J. Comput. Phys.*, **54**, 325–362.
- Stroud, A. H., 1971: *Approximate Calculation of Multiple Integrals*. Prentice-Hall, 431 pp.
- Thorndike, A. S., D. A. Rothrock, G. A. Maykut, and R. Colony, 1975: The thickness distribution of sea ice. *J. Geophys. Res.*, **80**, 4501–4513.
- Van Leer, B., 1979: Towards the ultimate conservative difference scheme. V. a second-order sequel to Godunov’s method. *J. Comput. Phys.*, **32**, 101–136.
- Winton, M., 2001: FMS sea ice simulator. GFDL Technical Document, Geophysical Fluid Dynamics Laboratory, Princeton, New Jersey.
- Xie, P. and P. A. Arkin, 1996: Analyses of global monthly precipitation using gauge observations, satellite estimates and numerical model prediction. *J. Climate*, **9**, 840–858.