

First- and Second-Order Conservative Remapping Schemes for Grids in Spherical Coordinates

PHILIP W. JONES

Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico

15 September 1998 and 14 December 1998

ABSTRACT

Coupling atmosphere, ocean, sea ice, and land surface models requires a means for remapping fields between grids in an accurate and conservative manner. A method is described here for computing interpolation weights for first- and second-order conservative remappings. The method is completely general and can be used for any grid on a sphere.

1. Introduction

In order to study the earth's climate system, climate models are often created by coupling individual component models that simulate the atmosphere, land surface, ocean, and sea ice. Component models are often developed as stand-alone models or perhaps as pairs of components (e.g., atmosphere–land or ocean–ice) and utilize grids that work best for that particular component. For example, ocean circulation models have begun to utilize grids that cover the polar regions effectively (Smith et al. 1995) or that can utilize local mesh refinement (Levin et al. 1997; Taylor et al. 1997). Similarly, atmosphere models have begun to use icosahedral grids or triangular tessellations for horizontal discretizations (Schättler and Krenzien 1997; Heikes and Randall 1995).

In a fully coupled climate model, state variables and heat and water fluxes must be transferred between models periodically, and such fields must be remapped from one component grid to another. Fluxes in particular must be remapped in a conservative manner in order to maintain the energy and water budgets of the coupled climate system. Currently, many coupled models utilize an area-weighted remapping where the interpolation weights are based on the fractional area-overlap of the source and destination grid cells (Bryan et al. 1996). Such a scheme is automatically conservative, but is only first-order accurate. Also, it has been primarily applied to situations where both grids are latitude–longitude rectangular grids. We present here a method for computing remap-

ping weights that will work for any general grid and will provide weights for both first- and second-order accurate remappings. The method was originally introduced by Dukowicz and Kodis (1987) but will be derived for spherical coordinates in the next section. The third section will describe practical considerations and problems encountered in spherical coordinates.

2. Conservative remapping

To compute a flux on a new (destination) grid that results in the same energy or water exchange as a flux f on an old (source) grid, the destination flux F at a destination grid cell k must satisfy

$$\bar{F}_k = \frac{1}{A_k} \int_{A_k} f \, dA, \quad (1)$$

where \bar{F} is the area-averaged flux and A_k is the area of cell k . Satisfying (1) will automatically satisfy the global conservation condition

$$\int_A F \, dA = \int_A f \, dA, \quad (2)$$

where the integral is over the entire grid surface A . Because the integral in (1) is over the area of the destination grid cell, only those cells on the source grid that are covered at least partly by the destination grid cell contribute to the value of the flux on the destination grid. If cell k overlaps N cells on the source grid, the remapping can be written as

$$\bar{F}_k = \frac{1}{A_k} \sum_{n=1}^N \int_{A_{nk}} f_n \, dA, \quad (3)$$

where A_{nk} is the area of the source grid cell n covered

Corresponding author address: Dr. Philip W. Jones, Los Alamos National Laboratory, T-3, Fluid Dynamics, P.O. Box 1663, Mail Stop B216, Los Alamos, NM 87545.
E-mail: pwjones@lanl.gov

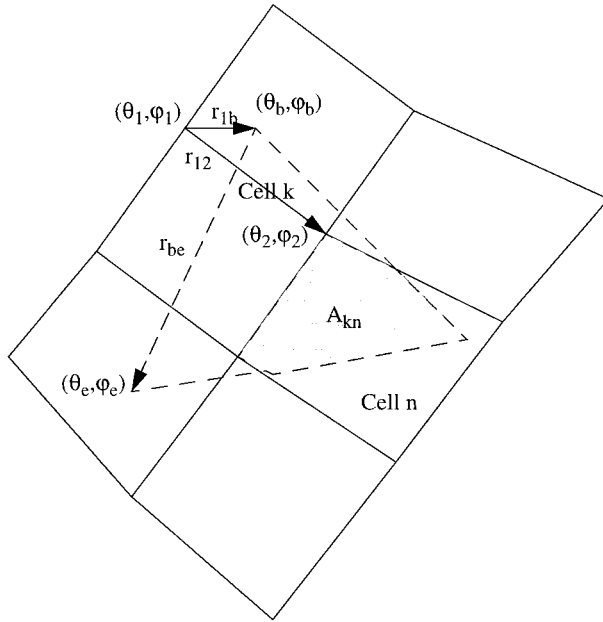


FIG. 1. An example of a triangular destination grid cell k shown by the dashed outline overlapping a quadrilateral source grid. The region A_{kn} , where cell k overlaps the quadrilateral cell n , is shaded. Vectors used by search and intersection routines are also labeled.

by the destination grid cell k , and f_n is the local value of the flux in the source grid cell (see Fig. 1).

Assuming f_n is constant across a source grid cell, (3) would lead to the first-order area-weighted schemes used in current coupled models. A more accurate form of the remapping is obtained by using a more accurate approximation of the flux on the source grid. For a conservative scheme, the flux must satisfy

$$\bar{f}_n A_n = \int_{A_n} f_n dA. \quad (4)$$

Dukowicz and Kodis (1987) approximate the source flux by

$$f_n = \bar{f}_n + \nabla_n f \cdot (\mathbf{r} - \mathbf{r}_n), \quad (5)$$

where $\nabla_n f$ is the gradient of the flux in cell n and \mathbf{r}_n is the centroid of cell n defined by

$$\mathbf{r}_n = \frac{1}{A_n} \int_{A_n} \mathbf{r} dA. \quad (6)$$

Such a distribution satisfies the conservation constraint and is equivalent to the first terms of a Taylor series expansion of f around \mathbf{r}_n . The remapping is thus second-order accurate if $\nabla_n f$ is at least a first-order approximation to the gradient.

The remapping can now be expanded in spherical coordinates as

$$\bar{F}_k = \sum_{n=1}^N \left[\bar{f}_n w_{1nk} + \left(\frac{\partial f}{\partial \theta} \right)_n w_{2nk} + \left(\frac{1}{\cos \theta} \frac{\partial f}{\partial \phi} \right)_n w_{3nk} \right], \quad (7)$$

where θ is latitude, ϕ is longitude, and the three remapping weights are

$$w_{1nk} = \frac{1}{A_k} \int_{A_{nk}} dA, \quad (8)$$

$$\begin{aligned} w_{2nk} &= \frac{1}{A_k} \int_{A_{nk}} (\theta - \theta_n) dA \\ &= \frac{1}{A_k} \int_{A_{nk}} \theta dA - \frac{w_{1nk}}{A_n} \int_{A_n} \theta dA, \end{aligned} \quad (9)$$

and

$$\begin{aligned} w_{3nk} &= \frac{1}{A_k} \int_{A_{nk}} \cos \theta (\phi - \phi_n) dA \\ &= \frac{1}{A_k} \int_{A_{nk}} \phi \cos \theta dA - \frac{w_{1nk}}{A_n} \int_{A_n} \phi \cos \theta dA. \end{aligned} \quad (10)$$

Again, if the gradient is zero, (7) reduces to a first-order area-weighted remapping. The second-order weights are an area-weighted distance from the source cell centroid.

Computing the area integrals in Eqs. (8)–(10) is simple for very simple grids, but can be difficult for general grids where overlap regions may be irregular in shape. To compute the area integrals for more general grids, Dukowicz and Kodis (1987) convert the area integrals into line integrals using the divergence theorem. Computing line integrals around the overlap regions is much simpler; one simply integrates first around every grid cell on the source grid, keeping track of intersections with destination grid lines, and then one integrates around every grid cell on the destination grid in a similar manner. After the sweep of each grid, all overlap regions have been integrated.

In spherical coordinates, the divergence theorem is

$$\int_A \nabla \cdot \mathbf{G} dA = \oint_C [G_\phi d\theta - G_\theta \cos \theta d\phi], \quad (11)$$

where C is the curve containing the area A . The selection of the function G is arbitrary as long as $\nabla \cdot \mathbf{G}$ is chosen to be the integrand of the above area integrals. If $G_\phi = 0$, the integrals in Eqs. (8)–(10) become

$$\int_{A_{nk}} dA = \oint_{C_{nk}} -\sin \theta d\phi, \quad (12)$$

$$\int_{A_{nk}} \theta dA = \oint_{C_{nk}} [-\cos \theta - \theta \sin \theta] d\phi, \quad (13)$$

and

$$\int_{A_{nk}} \phi \cos \theta dA = \oint_{C_{nk}} -\frac{\phi}{2} [\sin \theta \cos \theta + \theta] d\phi, \quad (14)$$

where C_{nk} is the counterclockwise path around the region A_{nk} . Computing these three line integrals during

the sweeps of each grid provides all the information necessary for computing the remapping weights.

3. Practical considerations

As mentioned in the previous section, the algorithm for computing the remapping weights is relatively simple. The process amounts to finding the location of the end point of a segment and then finding the next intersection with the other grid. The line integrals are then computed and summed according to which grid cells are associated with that particular subsegment. Despite this relative simplicity, the practical application of this method requires avoiding some spherical coordinate problems and optimizing search and intersection algorithms.

a. Search algorithms

The most time-consuming portion of the algorithm described above is finding which cell on one grid contains an end point from the other grid. Optimal search algorithms can be written when the grid is well structured and regular. However, if one requires a search algorithm that will work for any general grid, a hierarchy of search algorithms appears to work best. In such a case, a preliminary sorting of grid cells based on latitude bins or bounding boxes can be used to restrict a more robust but expensive algorithm.

Once the search has been restricted, a robust algorithm that works for most cases is a cross-product test. In this test, a cross product is computed between the vector corresponding to a cell side (\mathbf{r}_{12} in Fig. 1) and a vector extending from the beginning of the cell side to the search point (\mathbf{r}_{1b}). If

$$\mathbf{r}_{12} \times \mathbf{r}_{1b} > 0, \tag{15}$$

the point lies to the left of the cell side. If (15) holds for every cell side, the point is enclosed by the cell. This test is not completely robust and will fail for grid cells that are nonconvex.

b. Intersections

Once the location of an initial end point is found, it is necessary to check to see if the segment intersects with the cell side. If the segment is parametrized as

$$\theta = \theta_b + s_1(\theta_e - \theta_b) \quad \phi = \phi_b + s_1(\phi_e - \phi_b), \tag{16}$$

and the cell side as

$$\theta = \theta_1 + s_2(\theta_2 - \theta_1) \quad \phi = \phi_1 + s_2(\phi_2 - \phi_1), \tag{17}$$

where $\theta_1, \phi_1, \theta_2, \phi_2, \theta_b,$ and θ_e are end points as shown in Fig. 1, the intersection of the two lines occurs when θ and ϕ are equal. The linear system

$$\begin{bmatrix} (\theta_e - \theta_b) & (\theta_1 - \theta_2) \\ (\phi_e - \phi_b) & (\phi_1 - \phi_2) \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} (\theta_1 - \theta_b) \\ (\phi_1 - \phi_b) \end{bmatrix}, \tag{18}$$

is then solved to determine s_1 and s_2 at the intersection point. If s_1 and s_2 are between zero and one, an intersection occurs with that cell side.

It is important also to compute identical intersections during the sweeps of each grid. To ensure that this will occur, the entire line segment should be used to compute intersections rather than using a previous or next intersection as an end point.

c. Coincidences

Often, pairs of grids will share common lines (e.g., the equator). When this is the case, the method described above will double count the contribution of these line segments. Coincidences can be detected when computing cross products for the search algorithm described above. If the cross product is zero in this case, the end point lies on the cell side. A second cross product between the line segment and the cell side can then be computed. If the second cross product is also zero, the lines are coincident. Once a coincidence has been detected, the contribution of the coincident segment can be computed during the first sweep and ignored during the second sweep.

d. Spherical coordinates

Some aspects of the spherical coordinate system introduce additional problems for the method described above.

1) MULTIPLE-VALUED COORDINATE

Longitude is multiple valued on one line on the sphere, and this branch cut may be chosen differently by different grids. Care must be taken when calculating intersections and line integrals to ensure that the proper longitude values are used. A simple method is to always check to make sure the longitude is in the same interval as the source grid cell center.

2) POLE

Another problem with computing weights in spherical coordinates is the treatment of the pole. First, note that although the pole is physically a point, it is a line in latitude–longitude space and has a nonzero contribution to the weight integrals. If a grid does not contain the pole explicitly as a grid vertex, the pole contribution must be added to the appropriate cells. The pole contribution can be computed analytically.

The pole also creates problems for the search and intersection algorithms described above. For example, a grid cell that overlaps the pole can result in a non-convex cell in latitude–longitude coordinates. The cross-

product test described above will fail in this case. In addition, segments near the pole typically exhibit large changes in longitude even for very short segments. In such a case, the linear parametrizations used above result in inaccuracies for determining the correct intersections.

To avoid these problems, a coordinate transformation can be used poleward of a given threshold latitude (typically within one degree of the pole). A possible transformation is the Lambert equivalent azimuthal projection

$$\begin{aligned}
 X &= 2 \sin\left(\frac{\pi}{4} - \frac{\theta}{2}\right) \cos\phi \\
 Y &= 2 \sin\left(\frac{\pi}{4} - \frac{\theta}{2}\right) \sin\phi
 \end{aligned}
 \tag{19}$$

for the North Pole. The transformation for the South Pole is similar. This transformation is used only to compute intersections; line integrals are still computed in latitude–longitude coordinates. Because intersections computed in the transformed coordinates can be different from those computed in latitude–longitude coordinates, line segments that cross the latitude threshold must be treated carefully. To compute the intersections consistently for such a segment, intersections with the threshold latitude are detected and used as a normal grid intersection to provide a clean break between the two coordinate systems.

e. Gradients

For a second-order remapping, a gradient must be supplied. Methods for computing gradients can vary depending on the grid and numerical algorithm employed, and the remapping method above requires only that the gradient be at least first-order accurate. If a field to be remapped has strong gradients, numerical approximations to the gradient may be too steep. In such cases, (5) may overshoot the expected value of the field, resulting in spurious oscillations in the resulting remapped field. Avoiding such oscillations requires limiting the gradient, and sample methods for limiting the gradient are discussed in Dukowicz and Kodis (1987). When gradient limiters are employed, formal second-order accuracy is lost, but the remappings can be made monotone.

It should also be noted that (5) implies that little increase in accuracy is gained using the second-order remappings to remap from a fine grid to a coarse grid. Remapping from a fine grid to a coarse grid consists of averaging over cells on the fine grid; the gradient components in (5) will average to zero when averaging over an entire cell.

f. Mask conflicts

While computing the remapping weights, the fraction of a grid cell participating in a remapping is also computed. For an ocean-to-atmosphere mapping, this means that the land fraction on the atmosphere grid is computed automatically. This land fraction and the associated land mask may differ from that used by the atmosphere in stand-alone mode. In this case, one grid must be chosen as a “master” grid for determining land–ocean boundaries. The problem is further compounded if all components use different grids, and in this case all components must conform to the same “master” grid.

g. Optimizations

Throughout the descriptions above, the most general algorithms have been presented that are robust enough to work for any grid on a sphere. Such robustness comes at a cost in computational time, but the weights only need be computed once for each grid pair and stored in a file for later use. Computational cost is typically less of an issue than robustness and accuracy. However, the algorithms can be highly optimized for grids that are highly structured. For example, it is very easy to determine the location of a point in a simple latitude–longitude grid using an arithmetic calculation without using a full search algorithm. Also, for logically rectangular grids, it is simple to determine which cell you are entering after an intersection, so a search is unnecessary.

For grids in which cells share a common side (e.g., logically rectangular grids), the line integral contribution can be computed once for both grid cells on each side of the segment; the contribution to the cell on the right of the segment has the opposite sign of the contribution to the cell on the left. For a logically rectangular grid, all of the line integrals can be performed by integrating once along the *i*-axis and then once along the *j*-axis rather than integrating around each cell individually.

Finally, note that lines of constant longitude do not contribute to the integrals and can be ignored in the weight calculations. For some grids, this results in a large savings in computational work.

4. Results

The conservative first- and second-order remappings have been implemented and tested for a variety of grid combinations described below. Tests were conducted using several analytic fields. The first field is a cosine hill $f = 2 + \cos(\pi r/L)$ where *r* is the distance from the center of the hill and *L* is a length scale. Such a function is useful for determining the effects of repeated applications as shown below. The other two fields are representative of spherical harmonic wavefunctions. A relatively smooth function $f = 2 + \cos^2\theta \cos(2\phi)$ is

TABLE 1. Results of remappings from a T42 grid to a 1° latitude–longitude grid.

Field	Method	ϵ_{\max}	ϵ_{mean}
Hill	Bilinear	7.0×10^{-3}	1.7×10^{-4}
Hill	Bicubic	1.8×10^{-3}	2.0×10^{-5}
Hill	1st-order conservative	5.9×10^{-2}	1.1×10^{-3}
Hill	2d-order conservative	4.8×10^{-3}	8.4×10^{-5}
Y_2^2	Bilinear	1.7×10^{-3}	2.2×10^{-4}
Y_2^2	Bicubic	1.0×10^{-5}	2.4×10^{-6}
Y_2^2	1st-order conservative	1.8×10^{-2}	3.9×10^{-3}
Y_2^2	2d-order conservative	3.7×10^{-4}	1.2×10^{-4}
Y_{32}^{16}	Bilinear	8.0×10^{-2}	4.3×10^{-3}
Y_{32}^{16}	Bicubic	9.0×10^{-3}	5.9×10^{-4}
Y_{32}^{16}	1st-order conservative	1.4×10^{-1}	1.1×10^{-2}
Y_{32}^{16}	2d-order conservative	4.7×10^{-2}	2.1×10^{-3}

similar to a spherical harmonic with $l = 2$ and $m = 2$, where l is the spherical harmonic order and m is the azimuthal wave number. The function $f = 2 + \sin^{16}(2\theta) \cos(16\phi)$ is similar to a spherical harmonic with $l = 32$ and $m = 16$ and is useful for testing a field with relatively high spatial frequency and rapidly changing gradients. The latter two functions will be referred to as Y_2^2 and Y_{32}^{16} , respectively, although they are not true spherical harmonic functions.

A first test compared the conservative remappings to bilinear and bicubic interpolations for two simple grids. While the bilinear and bicubic interpolations are not conservative, such a test is useful to gauge the accuracy of the remappings. The grids in this case were a Gaussian latitude–longitude grid used for T42 spectral truncations and a latitude–longitude grid with one-degree spatial resolution. The results of a remapping from the coarse T42 grid to the finer one-degree grid are shown in Table 1. The errors shown in the table are relative errors $\epsilon = |(f_{\text{computed}} - f_{\text{analytic}})/f_{\text{analytic}}|$ and both the maximum and spatially averaged values are shown. For all of the test fields, the second-order conservative remappings resulted in errors somewhere between the bilinear

and bicubic results. The second-order mappings used analytic gradients, so a more typical case with numerically approximated gradients would likely be closer to the bilinear result. Such a result is consistent with the assumption of constant gradients across a cell for the second-order remappings. The first-order remappings assume a piecewise constant field and are much less accurate than the bilinear and bicubic cases. Such a piecewise constant field is a very poor approximation for fields with high spatial frequency exhibited by the Y_{32}^{16} test field, yet this is the method commonly used in current coupled models where fields must be conserved. Use of the second-order conservative mappings would greatly improve the accuracy of fluxes between coupled model components.

In order to show the robustness and generality of the method presented here, the remappings were tested using more general grids. Results for combinations of four different grids are shown in Table 2. The icosahedral (ICOS) grid (Heikes and Randall 1995) is made up of cells with either five or six sides and has an average resolution of 4.5° . It is used to provide a good test of the generality of the method. A second grid is a high-resolution (27 km) ice grid (ICE; Washington and Weatherly 1997) that provides a good test of a grid with incomplete coverage (the grid only covers the polar regions) and of the coordinate transformation utilized in the polar regions. The third grid is a displaced-pole grid used by the Los Alamos Parallel Ocean Program (POP) and is a general grid with a land mask. The final grid is a standard Gaussian latitude–longitude grid at T42 resolution. Only the conservative mapping results are shown as the bilinear and bicubic interpolations cannot be applied to some of these grids. Where bilinear interpolations can be applied, the second-order conservative results are again comparable to the bilinear results for accuracy.

For all remappings from a coarse grid to a fine grid,

TABLE 2. Results of remappings between various grid pairs.

Source grid	Destination grid	Field	1st-order conservative		2nd-order conservative	
			ϵ_{\max}	ϵ_{mean}	ϵ_{\max}	ϵ_{mean}
T42	POP	Y_2^2	1.5×10^{-2}	2.7×10^{-3}	1.3×10^{-3}	1.2×10^{-4}
POP	T42	Y_2^2	3.1×10^{-3}	4.6×10^{-4}	4.4×10^{-4}	8.9×10^{-5}
ICOS	POP	Y_2^2	3.0×10^{-2}	5.9×10^{-3}	2.2×10^{-3}	3.0×10^{-4}
POP	ICOS	Y_2^2	2.2×10^{-3}	3.9×10^{-4}	1.8×10^{-3}	2.4×10^{-4}
POP	ICE	Y_2^2	1.8×10^{-2}	2.2×10^{-3}	1.1×10^{-3}	3.7×10^{-5}
ICE	POP	Y_2^2	1.4×10^{-4}	1.4×10^{-5}	1.3×10^{-4}	1.1×10^{-5}
T42	POP	Y_{32}^{16}	1.2×10^{-1}	9.3×10^{-3}	5.1×10^{-2}	1.8×10^{-3}
POP	T42	Y_{32}^{16}	6.0×10^{-2}	3.4×10^{-3}	2.7×10^{-2}	1.2×10^{-3}
ICOS	POP	Y_{32}^{16}	4.5×10^{-1}	2.7×10^{-2}	2.4×10^{-1}	8.6×10^{-3}
POP	ICOS	Y_{32}^{16}	1.4×10^{-1}	9.0×10^{-3}	1.0×10^{-1}	7.0×10^{-3}
POP	ICE	Y_{32}^{16}	1.8×10^{-1}	1.5×10^{-2}	6.0×10^{-2}	1.9×10^{-3}
ICE	POP	Y_{32}^{16}	2.1×10^{-2}	9.2×10^{-4}	2.1×10^{-2}	8.6×10^{-4}
T42	POP	Hill	2.7×10^{-2}	5.0×10^{-4}	2.9×10^{-3}	4.5×10^{-5}
POP	T42	Hill	1.0×10^{-2}	3.5×10^{-4}	1.7×10^{-3}	7.2×10^{-5}
ICOS	POP	Hill	6.7×10^{-2}	1.1×10^{-3}	1.1×10^{-2}	1.5×10^{-4}
POP	ICOS	Hill	7.8×10^{-3}	5.1×10^{-4}	5.1×10^{-3}	3.1×10^{-4}

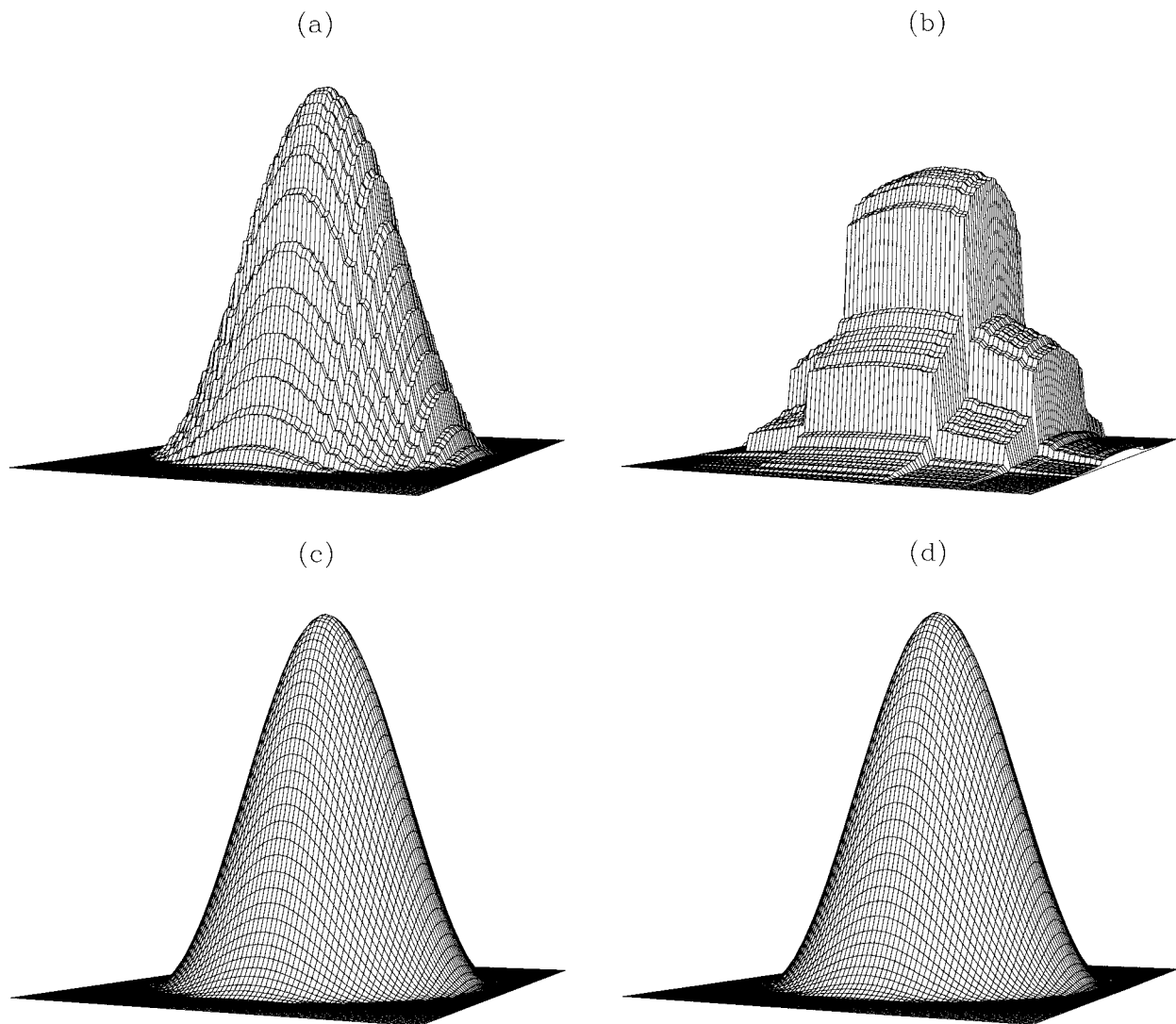


FIG. 2. Remapping of a cosine hill function after (a) one remapping at first order, (b) 100 remappings at first order, (c) one remapping at second order, and (d) 100 remappings at second order.

the second-order scheme results in an order-of-magnitude improvement over the first-order conservative remapping. The first-order mappings are particularly bad for the high-frequency Y_{32}^{16} field on the coarse grids considered here. The disadvantage of the second-order remapping is the computational cost; a factor of 3 more computations are required and two components of the gradient must be computed. Remappings from a fine grid to a coarse grid are an averaging operation, so there is virtually no advantage to using the second-order method in such cases, as can be seen in Table 2.

The conservation properties of the remappings were tested by computing global area integrals of the analytic field on the source grid and of the remapped field on the destination grid. The resulting integrals agreed to one part in 10^{-15} , so the remappings are conservative to machine accuracy.

The conservative remappings presented here have been used in production coupled model simulations, including the work of Washington and Weatherly (1997) and a spinup of the Los Alamos POP model in preparation of a full coupled model simulation. In both of these cases, the remappings have worked well and no biases have been detected associated with the remappings. However, to try and assess the effects of repeated remappings, a test was performed mapping the hill function back and forth repeatedly between a T42 grid and the 1° latitude–longitude grid. Figure 2 shows the result after one mapping and after 100 repeated mappings of the hill function for both the first-order and second-order methods. The second-order method retains the shape quite well, while the first-order method is quite diffusive and begins to show the pattern of the coarse grid on the fine grid. It should be noted that such a pattern does not

appear in actual coupled model simulations where the identical field is never mapped repeatedly.

5. Conclusions

The method described here provides a mechanism for computing remapping weights for both first- and second-order remappings. Use of these weights provides remappings that are up to second-order accurate and conservative to machine accuracy. Because the method is completely general, weights can be computed for any type of grid on a sphere, allowing developers of component models in a coupled model context to use whatever grid is appropriate for a given component and not be constrained by compatibility with other component model grids.

Software implementing the methods described in this paper is freely available from the author (<http://www.acl.lanl.gov/lanlclimate/SCRIP>).

Acknowledgments. The author would like to thank J. Dukowicz for many useful conversations regarding his method and A. Craig and T. Ringler for helping to test the method on various grids. This research is supported

by the Department of Energy under Contract W-7405-ENG-36.

REFERENCES

- Bryan, F. O., B. G. Kauffman, W. G. Large, and P. R. Gent, 1996: The NCAR CSM flux coupler. NCAR Tech. Note 424, 50 pp. [Available from NCAR, Boulder, CO 80307.]
- Dukowicz, J. K., and J. W. Kodis, 1987: Accurate conservative remapping (rezoning) for arbitrary Lagrangian-Eulerian computations. *SIAM J. Sci. Stat. Comput.*, **8**, 305–321.
- Heikes, R., and D. A. Randall, 1995: Numerical integration of the shallow-water equations on a twisted icosahedral grid. Part I: Basic design and results. *Mon. Wea. Rev.*, **123**, 1862–1880.
- Levin, J. G., M. Iskandarani, and D. B. Haidvogel, 1997: A spectral filtering procedure for eddy-resolving simulations with a spectral element ocean model. *J. Comput. Phys.*, **137**, 130–154.
- Schättler, U., and E. Krenzien, 1997: Model development for parallel computers at DWD. *Making Its Mark*, G. R. Hoffmann and N. Kreitz, Eds., World Scientific, 83–100.
- Smith, R. D., S. Kortas, B. Meltz, 1995: Curvilinear coordinates for global ocean models. Los Alamos Unclassified Rep. 95-1146, 51 pp.
- Taylor, M., J. Tribbia, and M. Iskandarani, 1997: The spectral element method for the shallow water equations on the sphere. *J. Comput. Phys.*, **130**, 92–108.
- Washington, W. M., and J. W. Weatherly, 1997: Simulations with a climate model with high-resolution ocean and sea ice. Polar Processes and Global Climate, Draft Summary Report from the Conference on Polar Processes and Global Climate, WMO/ICSU/IOC World Climate Research Programme. 250–252.