Geochemistry Geophysics Geosystems G3
AN ELECTRONIC JOURNAL OF THE EARTH SCIENCES
Published by AGU and the Geochemical Society

Technical Brief
Volume 7, Number 4
27 April 2006
Q04014, doi:10.1029/2005GC001073
ISSN: 1525-2027

A semi-Lagrangian Crank-Nicolson algorithm for the numerical solution of advection-diffusion problems

Marc Spiegelman
Department of Applied Physics and Applied Mathematics, Columbia University, Palisades, New York 10965, USA (mspieg@ldeo.columbia.edu)

Department of Earth and Environmental Sciences, Columbia University, Palisades, New York 10965, USA

Richard F. Katz
Lamont-Doherty Earth Observatory, Columbia University, Palisades, New York 10965, USA

[1] We present a hybrid method for the numerical solution of advection-diffusion problems that combines two standard algorithms: semi-Lagrangian schemes for hyperbolic advection-reaction problems and Crank-Nicolson schemes for purely diffusive problems. We show that the hybrid scheme is identical to the two end-member schemes in the limit of infinite and zero Peclet number and remains accurate over a wide range of Peclet numbers. This scheme does not have a CFL stability criterion allowing the choice of time step to be decoupled from the spatial resolution. We compare numerical results with an analytic solution and test both an operator split version of our method and a combined version that solves advection and diffusion simultaneously. We also compare results of simple explicit and implicit numerical schemes and show that the semi-Lagrangian Crank-Nicolson (SLCN) scheme is both faster and more accurate on the same problem. We then apply the combined SLCN scheme to a more geologically relevant benchmark for calculating the thermal structure of a subduction zone. This problem demonstrates that the SLCN scheme can remain stable and accurate at large Courant numbers even in flows with highly curved streamlines. Finally, we introduce a variable order interpolation scheme for the semi-Lagrangian schemes that reduces interpolation artifacts for sharp fronts without introducing numerical diffusion.

Components: 5043 words, 7 figures.

Keywords: advection; Lagrangian; method of characteristics; numerical.

Index Terms: 0560 Computational Geophysics: Numerical solutions (4255); 3225 Mathematical Geophysics: Numerical approximations and analysis (4260).

Received 15 July 2005; **Revised** 16 December 2005; **Accepted** 26 January 2006; **Published** 27 April 2006.

Spiegelman, M., and R. F. Katz (2006), A semi-Lagrangian Crank-Nicolson algorithm for the numerical solution of advection-diffusion problems, *Geochem. Geophys. Geosyst.*, *7*, Q04014, doi:10.1029/2005GC001073.

1. Introduction

[2] Advection-diffusion problems are ubiquitous in physical sciences and engineering and can pose significant challenges for accurate numerical solution. Typical problems arise where the Peclet number varies strongly across the domain as, for example, in convection problems that are predominantly advective but develop narrow diffusive boundary layers, or for problems where advective stirring significantly steepens concentration gradients enhancing diffusion. These problems can also prove difficult for operator splitting approaches when advection modifies gradients on timescales comparable to diffusion timescales. We have also encountered problems in advection-diffusion-reaction problems [e.g., *Spiegelman et al.*, 2001; *Spiegelman and Kelemen*, 2003] (see http://

www.ldeo.columbia.edu/~mspieg/SolFlow/) where the three processes are so closely balanced that small amounts of excess numerical diffusion (or even operator splitting) can change the numerical results. Ideally we would like a efficient numerical solver that can handle a wide range of Peclet numbers and solve both advection and diffusion simultaneously.

[3] Here, we describe a numerical scheme with these properties that we have found useful for solving chemical transport in reactive media. This scheme is a hybrid that combines Crank-Nicolson schemes for diffusion and semi-Lagrangian schemes for advection, drawing on advantages of both algorithms. We show that this scheme reduces identically to the two end-member schemes in the limits of zero and infinite Peclet number and is accurate for a wide range of Peclet numbers in between. Furthermore, SLCN has no inherent stability criterion and therefore it decouples the resolution in space from the resolution in time, reducing the computational cost of increasing the grid size. We present numerical tests of the scheme against an analytic solution that couples advection and diffusion. We show that the combined scheme is more accurate and faster than simple explicit or implicit schemes. We also consider a more geologically relevant problem of calculating the thermal structure in subduction zones and demonstrate the utility of being able to decouple spatial and temporal resolution. We discuss existing short-comings of this scheme and some approaches to correcting them.

2. Algorithms

[4] Before developing the hybrid semi-Lagrangian Crank-Nicolson scheme it is worth briefly reviewing the component schemes and their behavior.

2.1. Basic Algorithms

2.1.1. Crank-Nicolson Scheme

[5] A d-dimensional Crank-Nicolson scheme for solution of the diffusion equation

$$\frac{\partial u}{\partial t} = \nabla^2 u \tag{1}$$

can be written as

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \frac{1}{2}\left[\mathcal{L}\mathbf{u}^{n+1} + \mathcal{L}\mathbf{u}^n\right] \tag{2}$$

where $\mathbf{u}^n$ is the discrete solution to equation (1) at time step n and $\mathcal{L}$ is the discrete diffusion operator (Laplacian). Rearranging equation (2) yields

$$\left[I - \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^{n+1} = \left[I + \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^n \tag{3}$$

which is a sparse linear system (assuming $\mathcal{L}$ is a linear operator) for $\mathbf{u}^{n+1}$ which we solve rapidly using a standard geometric multi-grid solver [e.g., *Briggs et al.*, 2000] for cartesian geometries. The Crank-Nicolson scheme is unconditionally stable for all time steps.

2.1.2. Semi-Lagrangian Schemes

[6] Semi-Lagrangian schemes solve the advection-reaction problem

$$\frac{\partial u}{\partial t} + \mathbf{v}\cdot\nabla u = f(u, \mathbf{x}, t) \tag{4}$$

using the method of characteristics for *each* point in a regular mesh. Unlike fully Lagrangian (particle tracking schemes), which take an initially regular mesh and distort it, semi-Lagrangian schemes are effectively particle tracking schemes between two regular meshes that preserve the regularity of the mesh from time step to time step. *Staniforth and Cote* [1991] provide a useful introduction and details for the basic schemes. More complex variants (including conservative, parallel and 3-D unstructured mesh implementations) are given by *Bermejo and Conde* [2002], *Malevsky and Thomas* [1997], *Giraldo et al.* [2003], *Kaazempur-mofrad and Ethier* [2002], and *Kaazempur-mofrad et al.* [2003]. The essential idea of semi-Lagrangian schemes is that they actually solve the equation

$$\frac{Du}{Dt} = f(u, \mathbf{x}, t) \tag{5}$$

as an ordinary differential equation along the trajectory that connects from some take-off point $\mathbf{x}_*$ at time t to the *regular grid point* $\mathbf{x}$ at time $t + \Delta t$. In general they can be written as

$$\mathbf{u}^{n+1} = \mathbf{u}^n_* + \int_t^{t+\Delta t} f(\mathbf{u}(\tau), \mathbf{x}(\tau), \tau)\, d\tau \tag{6}$$

where $\mathbf{u}^{n+1}$ is the solution on the regular grid at time step $n + 1$. $\mathbf{u}^n_* = \mathbf{u}^n(t, \mathbf{x}_*)$ is the value of the solution at the take-off point (which is generally not on the grid). The final term is the line integral of the source terms along the trajectory. The take-

off point $\mathbf{x}_*$ is found by solving the ODE particle tracking problem

$$\frac{d\mathbf{x}}{dt} = -\mathbf{v} \tag{7}$$

starting at the grid point $\mathbf{x}$ and moving back in time. Once the take-off point is located, the value of the function and any associated source terms are found at this point by interpolation from the *regular* mesh $\mathbf{u}^n$. Semi-Lagrangian schemes have considerable flexibility in their choice of particle tracking and interpolation schemes [e.g., *Martin and Gorelick*, 2005; *Kaazempur-mofrad and Ethier*, 2002]. Here we use the simplest second-order, two-level scheme which uses an iterated midpoint scheme to find the take-off point and bicubic interpolation (in 2-D) to interpolate values at time step n (see *Staniforth and Cote* [1991] for details). Lower-order interpolants such as bilinear interpolation introduce systematic errors similar to the numerical diffusion introduced by low-order up-wind schemes and are not acceptable [e.g., see *Bermejo and Conde*, 2002]. However, higher-order interpolants can produce systematic overshoots near sharp edges that remain localized to the edges but grow with time. The overshoot problem can be handled by variable order interpolation schemes, for example, the QMSL scheme of *Bermejo and Conde* [2002]. Section 5 describes another hybrid interpolation scheme that preserves high-order interpolation for smooth fields but prevents overshoots near discontinuities.

[7] To complete equation (6) requires an approximation to the final integral. For the second-order two-level scheme, we use a trapezoidal rule so that the full algorithm can be written

$$\mathbf{u}^{n+1} = \mathbf{u}^n_* + \frac{\Delta t}{2}\left[\mathbf{f}^n_* + \mathbf{f}^{n+1}\right] \tag{8}$$

where $\mathbf{f}^n_* = f(\mathbf{x}_*,t)$ is the value of the source term (usually interpolated) at the take-off point.

[8] For pure advection problems with no source terms, the semi-Lagrangian scheme reduces to $\mathbf{u}^{n+1} = \mathbf{u}^n_*$, i.e., the value of the function remains constant on the characteristic and the old value is simply copied into its new position on the regular grid. As long as the take-off point and the old value can be found accurately, there is no stability limit to the length of the time step. In practice, the second-order accuracy of the midpoint scheme restricts the time step to $\sim$4–5 times the Courant number although for simpler flow fields, time steps greater than 10 times the Courant condition are possible.

2.2. Hybrid Schemes

[9] Given these two schemes for advection and diffusion, we combine them to solve the scaled, constant diffusivity advection-diffusion problem

$$\frac{\partial u}{\partial t} + \mathbf{v}\cdot\nabla u = \nabla^2 u \tag{9}$$

where the maximum scaled velocity $\|\mathbf{v}\|_{\max}$ is order the Peclet number, $\mathrm{Pe} = \frac{Lv_0}{\kappa}$.

2.2.1. Operator Splitting OS-SLCN

[10] The first approach simply uses the semi-Lagrangian scheme to advect the quantity for a time Δt such that $\mathbf{u}^{n'} = \mathbf{u}^n_*$ and then diffuses for a time Δt using the CN scheme. The operator split semi-Lagrangian Crank-Nicolson scheme for equation (9) is then

$$\left[I - \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^{n+1} = \left[I + \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^{n'} \tag{10}$$

[11] Note that $\mathbf{u}^{n'}$ has first been moved onto the regular grid before the diffusion operator is applied. A more complex operator split-scheme for 3-D unstructured meshes is presented by *Kaazempur-mofrad et al.* [2003] combining "characteristic Galerkin" schemes for advection [*Kaazempur-mofrad and Ethier*, 2002] together with standard Galerkin treatment of diffusion. For small time steps or for flows that do not significantly distort gradients over a time step, the operator split SLCN scheme can produce accurate solutions for large time steps (see section 3). However, for strongly distorting flows where advection and diffusion operate simultaneously, we find that a small modification to this scheme is more reliable.

2.2.2. Operator Combined SLCN

[12] In the operator combined SLCN scheme, we consider the problem slightly differently as

$$\frac{Du}{Dt} = \nabla^2 u \tag{11}$$

and treat it in the same manner as equation (5) with the diffusion as a "source term." Using the same discretization that leads to equation (8) we write the SLCN scheme as

$$\mathbf{u}^{n+1} = \mathbf{u}^n_* + \frac{\Delta t}{2}\left[(\mathcal{L}\mathbf{u}^n)_* + \mathcal{L}\mathbf{u}^{n+1}\right] \tag{12}$$

where $(\mathcal{L}\mathbf{u}^n)_*$ is the diffusion operator acting at the *take-off point* $\mathbf{x}_*$ before advection has distorted the gradients. If $\mathcal{L}$ is the discrete Laplacian operator and $\mathcal{C}_*$ is the bicubic interpolation operator at point $\mathbf{x}_*$ then it is straightforward to show, for a regular cartesian grid, that the order of interpolation and diffusion is interchangeable, i.e., $(\mathcal{L}\mathbf{u}^n)_* = \mathcal{L}\mathcal{C}_*\mathbf{u}^n = \mathcal{C}_*\mathcal{L}\mathbf{u}^n$. Thus, in practice, it is usually easier to apply the Laplacian on the regular grid at time step n and then interpolate the resulting field. With these definitions equation (12) can be rearranged to yield

$$\left[I - \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^{n+1} = \mathbf{u}^n_* + \frac{\Delta t}{2}(\mathcal{L}\mathbf{u}^n)_* \tag{13}$$

or

$$\left[I - \frac{\Delta t}{2}\mathcal{L}\right]\mathbf{u}^{n+1} = \mathcal{C}_*\left[\left(I + \frac{\Delta t}{2}\mathcal{L}\right)\mathbf{u}^n\right] \tag{14}$$

[13] In the limit of no motion (Pe = 0 or $\mathbf{v} = \mathbf{0}$), equation (14) is identical to a Crank-Nicolson scheme (equation (3)) as $\mathcal{C}_* = I$ (or $\mathbf{u}^n_* = \mathbf{u}^n$ and $(\mathcal{L}\mathbf{u}^n)_* = \mathcal{L}\mathbf{u}^n$). It is less obvious that in the limit of no diffusion (Pe $\rightarrow \infty$), this scheme reduces identically to the semi-Lagrangian scheme.

[14] To see this, we first note that although none of these schemes have stability criteria, we still have to choose a time step Δt based on some accuracy criterion. For mixed advection-diffusion problems we choose a time step based on the fastest process and an accuracy criterion based on either the maximum number of grid points we want to move in a time step or the smallest wavelength that we want to decay accurately. For example we set

$$\Delta t = \min\left(\frac{\alpha \Delta x}{\| \mathbf{v} \|_{\max}}, (\beta \Delta x)^2\right) \tag{15}$$

where Δx is a measure of the grid spacing, α is the maximum number of grid points to move in a time step (i.e., the maximum Courant number) and $\lambda = \beta \Delta x$ is the minimum wavelength to resolve for diffusion at long times. In the limit of large velocities (Pe $\rightarrow \infty$), the time it takes to move α grid points is negligible (i.e., $\Delta t = O(\epsilon)$). In the limit $\Delta t \rightarrow 0$ equation (13) reduces to $\mathbf{u}^{n+1} = \mathbf{u}^n_*$ which is just the semi-Lagrangian advection scheme.

[15] For strongly advection dominated problems, it usually makes more sense to scale by the advection time so that the dimensionless velocities are order 1, in which case the problem becomes

$$\frac{Du}{Dt} = \frac{1}{\mathrm{Pe}}\nabla^2 u \tag{16}$$

and the combined SLCN scheme becomes

$$\left[I - \frac{\Delta t}{2\mathrm{Pe}}\mathcal{L}\right]\mathbf{u}^{n+1} = \mathbf{u}^n_* + \frac{\Delta t}{2\mathrm{Pe}}(\mathcal{L}\mathbf{u}^n)_* \tag{17}$$

and the Peclet number enters into the operator. With this scaling, it is clear that as Pe $\rightarrow \infty$ the problem reduced identically to the non-diffusive semi-Lagrangian scheme.

3. An Analytic Test Problem

[16] Strictly speaking, the method of characteristics does not work for advection-diffusion problems because the diffusion operator moves information between particle trajectories such that characteristics do not exist for all times. However, in practice, this numerical scheme works well because it only considers the particle trajectories for a single time step and just provides a more accurate method to map a uniform grid $\mathbf{u}^n$ at time t to another uniform grid $\mathbf{u}^{n+1}$ at time $t + \Delta t$. We have not proved this rigorously, but below we describe a useful test of the algorithm for an analytic solution that combines both advection and diffusion in a non-separable way.

[17] This problem considers the solution of equation (9) in an infinite domain with velocity field

$$\mathbf{v}(\mathbf{x}, t) = U(y) = \mathrm{Pe}y\mathbf{i} \tag{18}$$

and initial condition

$$u(\mathbf{x}, 0) = e^{i\mathbf{k}\cdot\mathbf{x}} \tag{19}$$

Physically, this problem corresponds to advection-diffusion of a plane wave in a shearing flow field (see Figure 1).

[18] In the absence of diffusion, this problem can be solved by characteristics for any initial condition $u(\mathbf{x}, 0) = f(\mathbf{x})$ such that $u(\mathbf{x}, t) = f(x - \mathrm{Pe}yt, y)$. In the case of a plane wave initial condition, the solution is

$$u(\mathbf{x}, t) = e^{i\left[k_x(x - \mathrm{Pe}yt) + k_y y\right]} \tag{20}$$

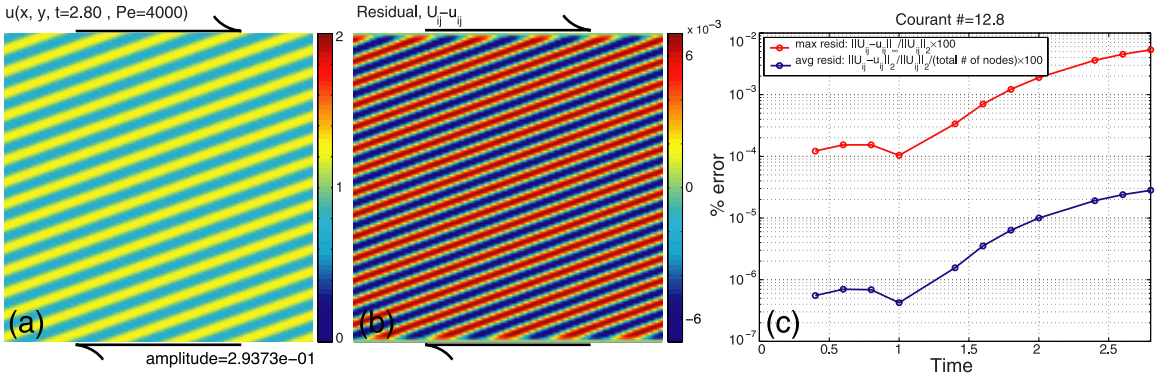

Figure 1. Results of an example calculation using the operator combined SLCN scheme. (a) The u field at model time $t = 2.8$ with a grid spacing of 25 points per initial wavelength. (b) The residual, analytic minus numerical, at the same time. (c) The residual over model time. The red curve shows the maximum residual normalized by the L2 norm of the discrete analytic solution. The blue curve is the grid averaged L2 norm of the residual, normalized in the same way.

which can be written more physically as $u(\mathbf{x}, t) = e^{i\mathbf{k}(t)\cdot\mathbf{x}}$ where

$$\mathbf{k}(t) = k_x\mathbf{i} + (k_y - k_x \mathrm{Pe} t)\mathbf{j}$$

is a time dependent wave-number. Thus any initial plane wave remains a plane wave but changes wavelength with time. For an initially vertical plane wave ($k_y = 0$), shear increases the wave-number with time and should enhance diffusion.

[19] An analytic solution to the full advection-diffusion problem can be found by seeking solutions of the form

$$u(\mathbf{x}, t) = e^{i(\mathbf{k}(t)\cdot\mathbf{x}) + s(t)} \tag{21}$$

Substituting equations (18) and (21) into equation (9) and solving for $s(t)$ gives

$$s(t) = -\left[k^2 t - \mathrm{Pe} k_x k_y t^2 + (\mathrm{Pe} k_x)^2 t^3/3\right] \tag{22}$$

where $k^2 = k_x^2 + k_y^2$. In the absence of advection (Pe = 0), this solution is simply the exponential decay of a static plane wave. When Pe > 0 and initially $k_y = 0$, the shear enhances the decay rate of the amplitude as expected.

[20] We test both the operator split and combined versions of the SLCN scheme against the real part of this solution for a computational domain $x \in [-1/2, 1/2]$, $y \in [-1/2, 1/2]$. Boundary conditions are periodic in the x-direction and Dirichlet in y; we prescribe the analytical solution at $y = \pm 1/2$ for all x. The initial condition that we use is $k_y = 0$, $k_x = 10\pi$ for five full cycles in the x-direction.

[21] In addition to the two schemes described above, we also compare results with numerical solutions using a Forward-time centered-space (FTCS) scheme as well as a semi-implicit Crank-Nicolson centered-space (CNCS) scheme. In both schemes, the advection term is discretized as

$$\mathrm{Pe} y \frac{\partial u}{\partial x} \sim \frac{\mathrm{Pe} y}{2\Delta x}\left[u^n_{i+1,j} - u^n_{i-1,j}\right] = A\mathbf{u}^n$$

and therefore the explicit FTCS scheme can be written as

$$\mathbf{u}^{n+1} = [I + \Delta t(-A + \mathcal{L})]\mathbf{u}^n \tag{23}$$

and the CNCS scheme as

$$\left[I + \frac{\Delta t}{2}(A - \mathcal{L})\right]\mathbf{u}^{n+1} = \left[I - \frac{\Delta t}{2}(A - \mathcal{L})\right]\mathbf{u}^n \tag{24}$$

The linear system in equation (24) is solved rapidly using multigrid. The FTCS is a straw-man scheme: its CFL condition limits it to very small time steps and even under these conditions its accuracy and efficiency are poor. However, it provides a useful baseline for the simplest explicit scheme. The CNCS scheme is better than FTCS however it too has a stability criterion that limits the time step size.

4. Results

[22] For each run of the benchmark problem we measure the accuracy of the solution and the

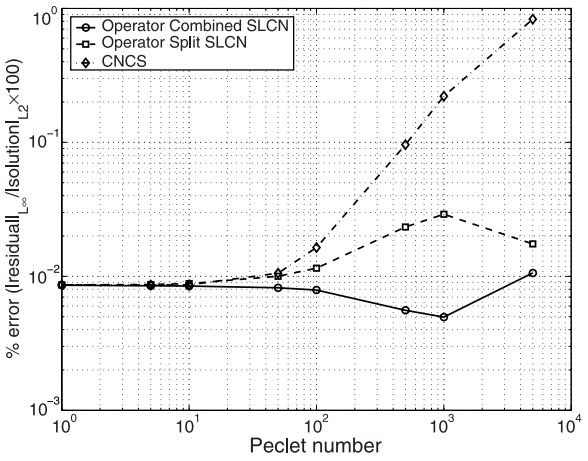

Figure 2. L∞ percent error as a function of Peclet number for CNCS, operator split SLCN, and operator combined SLCN. The time step is set according to equation (15), with $\beta = 0.7$ and $\alpha = 6$ for the semi-Lagrangian algorithms and $\alpha = 1$ for the CNCS scheme.

process time. Accuracy is measured by comparing the numerical solution to a discretized version of the exact solution $\mathbf{u}^n_{true}$:

$$\%\text{error} = \frac{\| \mathbf{u}^n_{true} - \mathbf{u}^n \|_{2\,\text{or}\,\infty}}{\| \mathbf{u}^n_{true} \|_2} \times 100 \qquad (25)$$

Timing results are relative to one "work unit" which is the time it takes for 1200 FTCS sweeps on a 100 × 100 grid (which at one instant in computational history was conveniently equal to 1 second). One FTCS sweep is a good measure of a standard stencil operation.

[23] Figure 1 shows the results of an example calculation using operator combined SLCN for the scaling given by equations (16) and (17). In this example the Peclet number is 4000 and final shear strain is 0.625. This combination of parameters results in the decay of the signal to 30% of its initial amplitude. The L2 norm percent error is parallel to the infinity norm in our calculations; for the rest of the paper, percent error is reported in terms of the infinity norm.

[24] Figure 2 shows the comparison of accuracy as a function of Peclet number for the SLCN scheme in operator split and combined versions with the CNCS scheme. Each simulation is run to approximately one e-folding time in amplitude, thus the maximum amount of shear increases with Peclet number. For small Peclet number (Pe $\lesssim$ 10), all the schemes give the same results. This is as expected: they all reduce to pure Crank-Nicolson for Pe $\rightarrow$ 0. For large Peclet number runs where the maximum shear is larger, SLCN performs considerably better than CNCS, with a significant advantage for the operator combined version of SLCN.

[25] Figure 3 shows the results of a suite of simulations similar to Figure 1 but with different Δx and Δt. All calculations have been run to one e-folding time in amplitude. It is clear that even for time steps an order of magnitude longer, SLCN is more accurate than the other schemes. For example, in Figure 3c, with 12 grid intervals per initial wavelength and a Courant number of about 10, the error is 0.03%, around an order of magnitude smaller than the CNCS scheme can achieve.

[26] The computational time required for each of the calculations performed is shown in Figure 4. Computation time for the CNCS scheme and the SLCN schemes are comparable for a given Courant number as they use the same multigrid scheme for solving the respective linear systems. However, the SLCN scheme is stable and accurate to much higher Courant number, so it is capable of accelerating the simulation significantly. For the example given above, the SLCN scheme is about five times faster than the CNCS scheme for an order of magnitude improvement in accuracy.

[27] A comparison of operator split and operator combined SLCN shows that while the operator combined SLCN scheme has a wide stability range where accuracy is approximately constant, the operator split version has two regimes of accuracy dependence on Courant number. In the first regime, at lower Courant numbers, the performance of the scheme improves with increasing time step size. The second regime shows a rapid degeneration of the performance as the time step size continues to increase.

5. Discussion

[28] The utility of the shearing, diffusing plane wave as a benchmark problem is that it couples advection and diffusion in a non-trivial way while still allowing for an analytic solution. This test clearly distinguishes the behavior of the different schemes and shows that the SLCN scheme is significantly more accurate and efficient than the other schemes considered, particularly at high Pe. The performance of the operator split SLCN scheme is the same as the operator combined version, but its accuracy is highly variable and overall, the combined scheme appears to be more

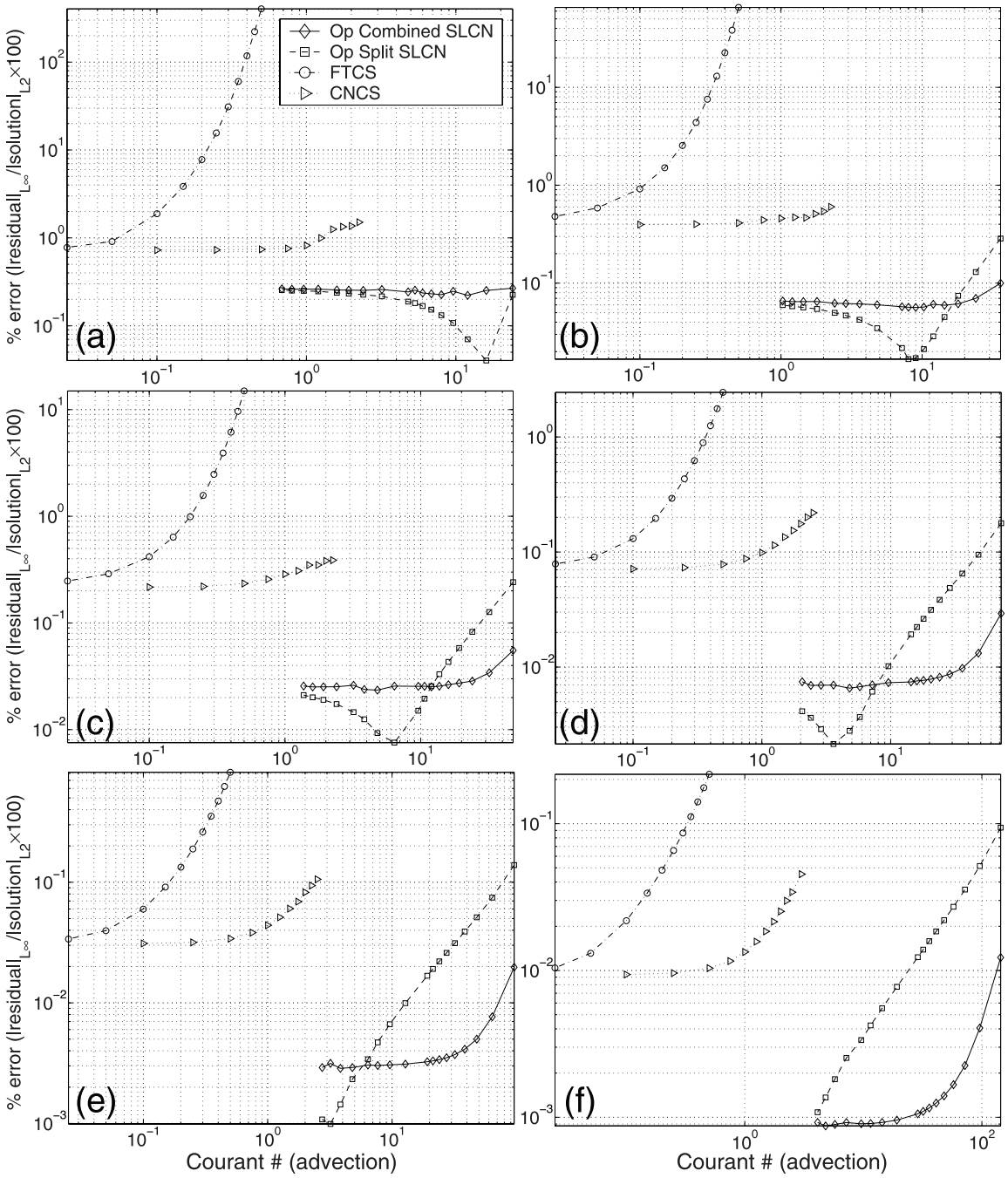

Figure 3. L∞ percent error (as defined in equation (25)) as a function of Courant number, $\frac{\mathrm{Pe}v_{\max}\Delta t}{\Delta x}$ at the constant $\mathrm{Pe}v_{\max}$ used in Figure 1. Each graph represents a different grid spacing Δx, which is described in terms of grid points per wavelength of the initial u field (Figure 1): (a) 6 grid points per wavelength (i.e., 33 $\times$ 33 grid), (b) 9 points per wavelength, (c) 12, (d) 19, (e) 25, and (f) 38.

robust. As yet, we do not have a good understanding of the behavior of the operator split scheme.

[29] Nevertheless, there are some obvious limitations of this benchmark problem with respect to testing the SLCN scheme for more complex flows. The principal issue is that in a shear flow, the velocity along any 1-D horizontal line is a constant that only depends on height. Thus the calculated characteristics are exact and interpolation is reduced to 1-D along lines (although the times reported are for full bicubic interpolation). A more demanding test, albeit without an analytic solution, is to calculate the advection-diffusion of tempera-

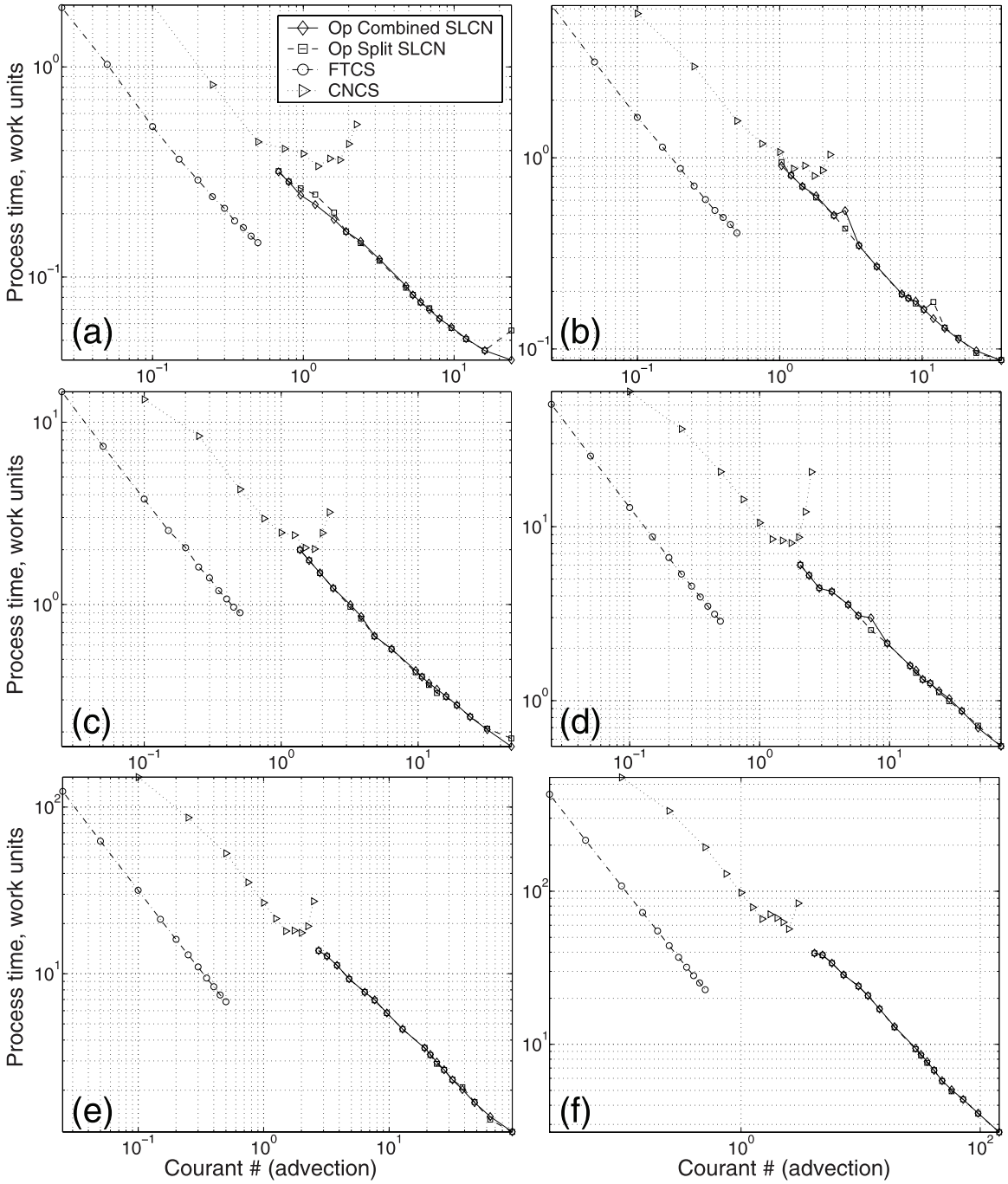

Figure 4. Computational time (in work units) as a function of Courant number, $\frac{\mathrm{Pe}v_{\max}\Delta t}{\Delta x}$ for the runs shown in Figure 3. One work unit is the time required for 1200 FTCS time steps on a 100 × 100 grid. Each graph represents a different grid spacing. These are, in terms of grid points per wavelength of the initial q field: (a) 6, (b) 9, (c) 12, (d) 19, (e) 25, and (f) 38.

ture for the kinematic "subduction zone" problem shown in Figure 5a. This problem is described in detail at the Web site associated with [*van Keken et al.*, 2005] (see http://www.geo.lsa.umich.edu/keken/subduction/benchmark/) and corresponds to benchmark 1a. This problem requires solving for the advection-diffusion of temperature in a flow field with strongly curved particle trajectories, a stagnation point at the wedge corner, a discontinuous velocity field between the upper plate and subducting plate, and extremely large temperature gradients in the thermal boundary layers. Nevertheless, the operator combined SLCN scheme performs well for this problem.

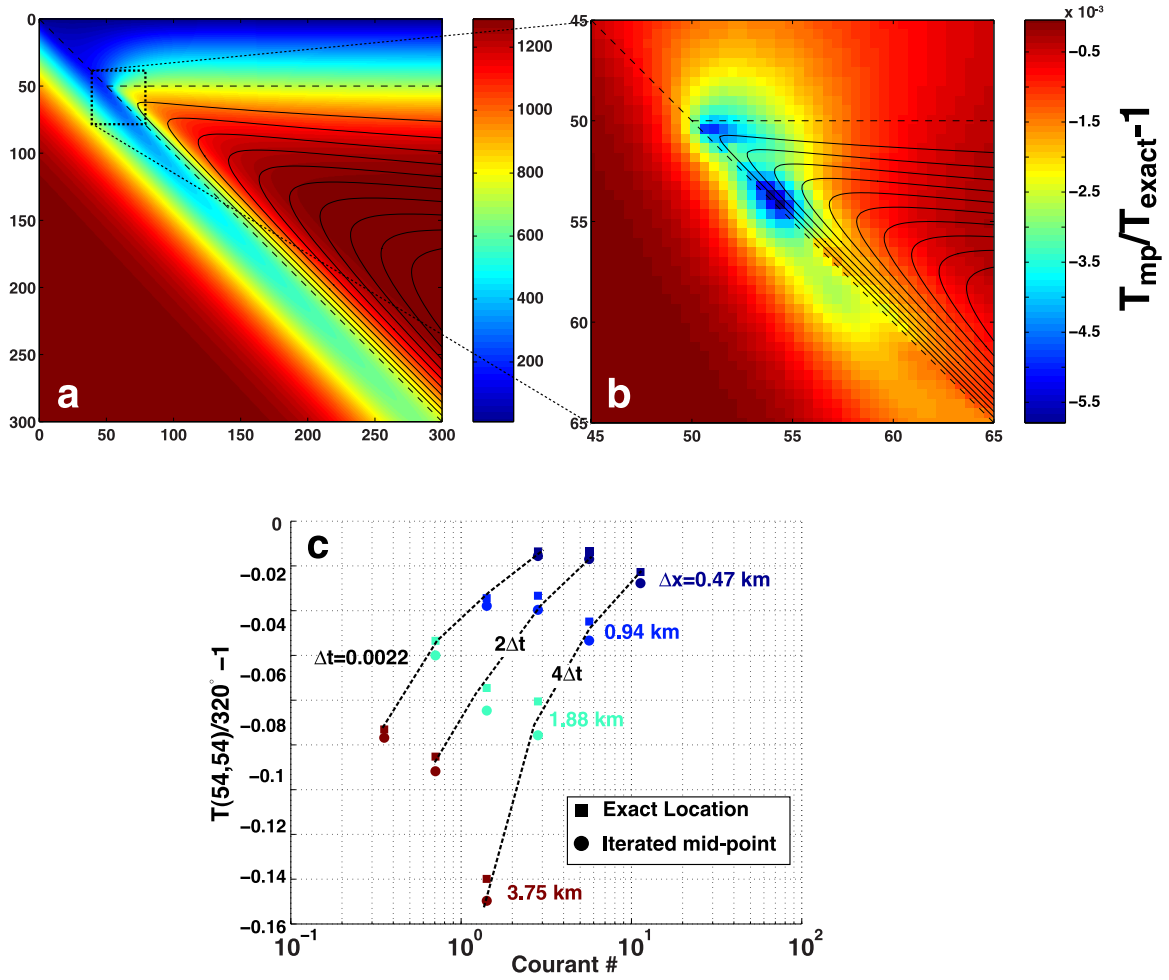

Figure 5. Advection-diffusion of temperature for the kinematic subduction zone benchmark 1a [*van Keken et al.*, 2005]. (a) Temperature field and stream function near the "wedge corner" for a problem with uniform grid spacing $\Delta = 0.475$ km and Courant number $\alpha = 4\sqrt{2} = 5.7$ which advects 4 grid squares along the slab per time step. (b) Relative temperature *difference* due to take-off point mislocation ($T_{\text{midpoint}}/T_{\text{exact}} - 1$). This calculation has the same grid spacings as Figure 5a but twice the Courant number. Even for $\alpha = 8\sqrt{2} = 11.3$, the maximum temperature difference due to point mislocation is less than 2°C. (c) Scaled temperature differences from 320°C at a point located at 54 km depth on the slab ($T(54,54)$) as a function of grid spacing, Courant number, and particle tracking scheme. Dashed lines connect runs with the same time step Δt. Note, smaller errors can actually be produced for comparable cost by using more highly refined grids and larger time steps.

[30] For example, the errors due to strongly curved streamlines are small even at Courant numbers ~10. Figure 5b shows the difference in temperatures calculated using the second-order iterated midpoint scheme to find the take-off points relative to a scheme with near exact take-off positions calculated using a adaptive time step Runge-Kutta scheme on the analytic velocity field. The maximum differences are less than 0.6% (<2°C for the temperature scaling used here). More significant errors arise from underresolving the very large temperature gradients near the wedge corner. Figure 5c shows the relative temperature difference from 320°C at a reference point located at 54 km depth on the slab (i.e., $T(54,54)$). This point has proved to be a useful benchmark for this problem [see *van Keken et al.*, 2005]) as it is extremely sensitive to resolution in the corner (and, as it turns out, to how the velocity discontinuity between the upper plate and the slab is implemented). Temperature differences are shown as a function of grid spacing and Courant number for constant values of the time step Δt. Temperature is most sensitive to the grid spacing and not the time step or particle tracking scheme and shows that accurate solution of this problem requires highly refined grids. For time-dependent problems, however, the SLCN scheme allows for accurate spatial resolution with-

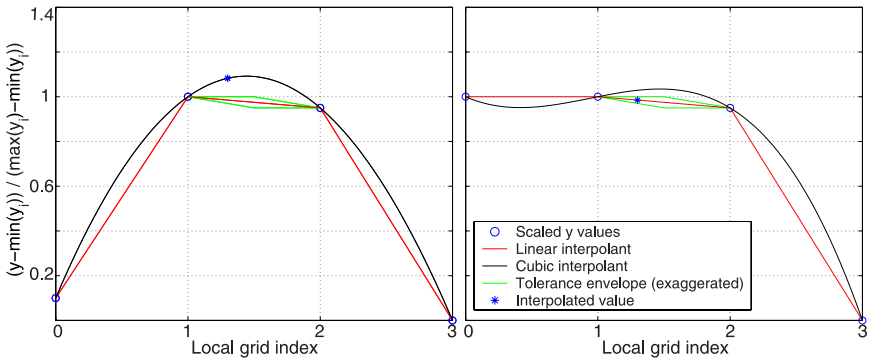

Figure 6. Two examples of adaptive interpolation with overshoot detection, both with $x = 1.3$. The green lines delineate the tolerance envelope, the thickness of which is exaggerated for clarity. (a) In this case, assuming that the field is smooth, the cubic interpolant is a better approximation than the linear one. The overshoot detection algorithm correctly returns false. (b) This figure shows the overshoots that characterize the cubic interpolant near an edge. The overshoot detection algorithm returns true, and thus the linear interpolant is chosen.

out a commensurate penalty in time step. In this example, it is even possible to achieve improved accuracy at the *same* computational cost by increasing grid resolution and taking longer time steps.

[31] Another limitation of the shearing-diffusing plane-wave is that the quantity being transported varies smoothly in space. It is common to encounter shocks or fronts in the transported quantity in high Peclet number advection-diffusion problems. A known problem with the semi-Lagrangian advection scheme arises under these circumstances. The bicubic interpolant, chosen to reduce numerical diffusion without incurring too great a computational cost, tends to overshoot near discontinuities in the field. The overshoot problem can be dealt with by considering error estimates between different interpolation schemes (e.g., see *Bermejo and Conde* [2002] for a conservative quasi-monotone scheme for pure advection problems). Here we present an adaptive interpolation method that retains the advantages of bicubic interpolation while avoiding its associated overshoot problems. To demonstrate this issue, and our solution, we performed a standard test of the semi-Lagrangian advection scheme: calculating solid-body rotation of a square bordered with a discontinuity. Issues of numerical diffusion should be negligible for problems with moderate physical diffusivity, as diffusion will smooth the advected quantities and avoid overshoots.

[32] The adaptive interpolation method makes use of the fact that a bicubic interpolation is executed by doing five cubic interpolations on four points each. Avoiding overshooting on each of these five one dimensional interpolation steps is thus required for avoiding overshooting in the overall bicubic interpolation.

[33] In constructing the one-dimensional cubic overshoot detection algorithm we make the assumption that the field is sufficiently well resolved on the grid that there are no oscillations at the grid scale. Thus we expect that over a set of four points, the first derivative of the field can change signs at most once. Given a set of field values y_i with local index $i \in \{0, 1, 2, 3\}$ and a real-valued interpolation target in local grid coordinates $x = i\Delta x$, $1 \leq x \leq 2$, we detect an overshoot using the following steps:

[34] 1. For each i, compute the slope, y'_i, of the cubic interpolant at $x = i\Delta x$.

[35] 2. There are oscillations in the interpolant over the grid segment if SIGN(y'_3) equals SIGN(y'_0) and SIGN(y'_2) does not equal SIGN(y'_3).

[36] Because this test is fairly expensive, it is performed only if the set of points y meets two conditions. First, the absolute range of y over the interpolating patch must be greater than a specified fraction of the range in values over the whole domain. If the set of interpolation points is roughly constant within this fraction it may still have features in the noise that cause detectable but small overshoots that do not damage the solution. Second, the cubic interpolant must fall outside of a specified envelope around the linear interpolant. This envelope is defined by a tolerance on the normalized difference between the

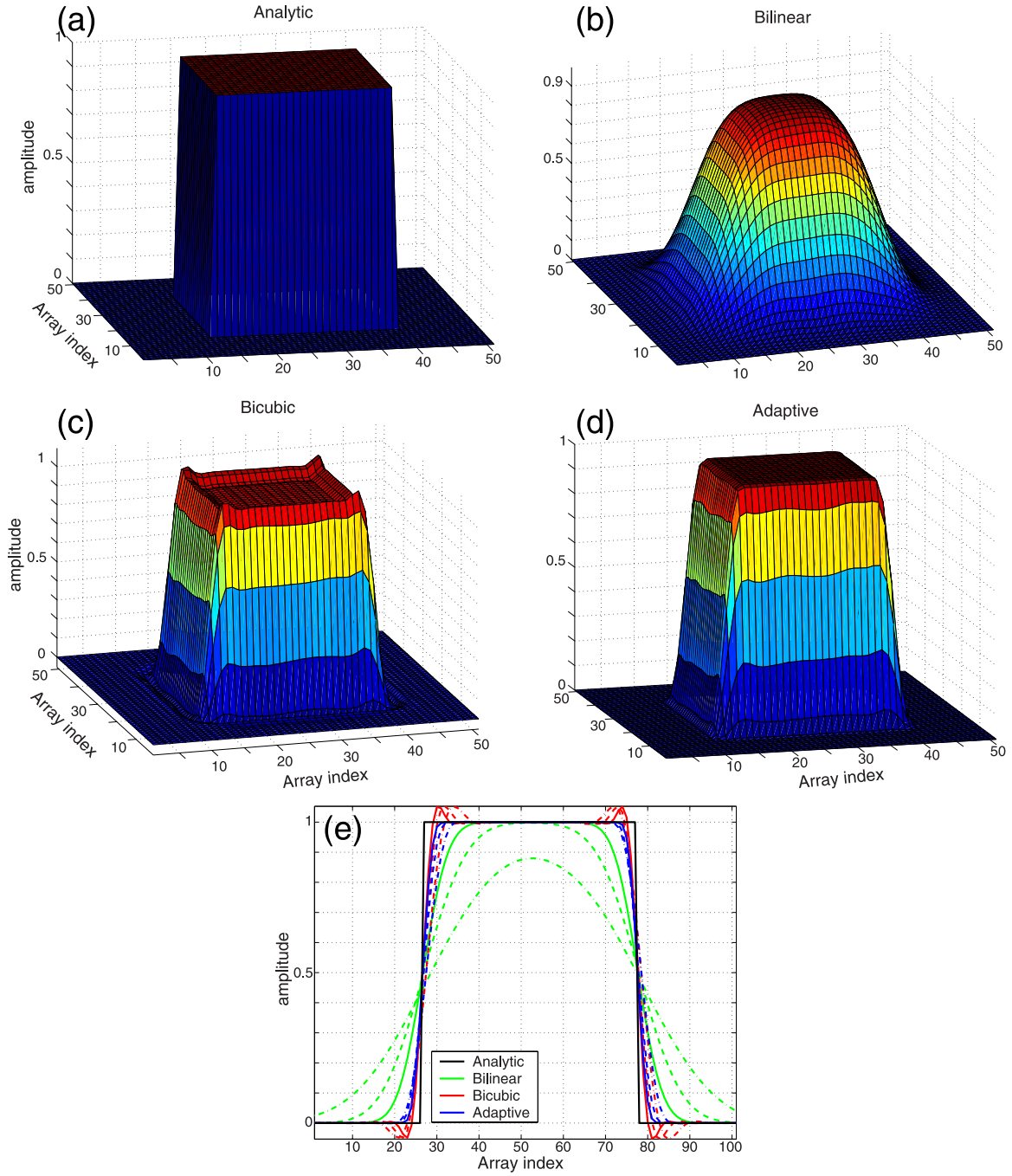

Figure 7. Results for three interpolation methods for the semi-Lagrangian advection scheme, compared with the analytic solution after three full rotations. (a) Analytic solution. (b) Bilinear interpolation. (c) Bicubic interpolation. (d) Adaptive interpolation. (e) Profiles across the box after 1 (solid line), 3 (dashed line), and 9 (dash-dotted line) full revolutions.

cubic and the linear interpolants, as shown in Figure 6.

[37] The results of the solid body rotation test are shown in Figure 7 for three full rotations around a corner of the box. The overshoots evident in the pure bicubic interpolation run are suppressed by the adaptive interpolation method. Bilinear interpolation is included for comparison (but tends to show large amounts of numerical diffusion).

6. Conclusions

[38] A general numerical solver for advection-diffusion problems must be capable of handling

pure advection, pure diffusion and any combination of the two. The accuracy and stability of such a solver becomes especially critical in the case of advectively dominated problems with sharp spatial gradients in the transported quantity. The SLCN algorithm draws on the strengths of the Crank-Nicolson scheme for diffusion and the semi-Lagrangian scheme for advection. Since neither of these have time steps limited by a stability criterion, the SLCN scheme is also unconditionally stable. We have shown that the SLCN scheme is accurate and efficient over a broad range of time step sizes relative to two other advection-diffusion schemes when the advection and diffusion operators act simultaneously on the transported field. For more complex flow fields, the SLCN schemes allows for enhanced spatial resolution to resolve steep boundary layers without a commensurate penalty in the time step. Finally, by employing our adaptive interpolation method, the SLCN scheme is able to handle discontinuities while reducing the problems of cubic interpolant overshooting, and maintaining its low numerical diffusivity.

Acknowledgments

[39] The authors would like to acknowledge Peter van Keken, Louis Moresi, David May, Ritske Huisman, and Phillipe Fullsack for timely and useful reviews and, in particular, for Moresi and May for reproducing the scheme and results. This work is supported in part by NSF grants OCE-0137108 and EAR-020785. R. Katz is supported by a DOE CSGF fellowship.

References

Bermejo, R., and J. Conde (2002), A conservative quasi-monotone semi-Lagrangian scheme, *Mon. Weather Rev.*, *130*(2), 423–430.

Briggs, W., V. Henson, and S. McCormick (2000), *A Multigrid Tutorial*, 2nd ed., Soc. for Ind. and Appl. Math., Philadelphia, Pa.

Giraldo, F., J. Perot, and P. Fischer (2003), A spectral element semi-Lagrangian (SESL) method for the spherical shallow water equations, *J. Comput. Phys.*, *190*(2), 623–650.

Kaazempur-mofrad, M., and C. Ethier (2002), An efficient characteristic Galerkin scheme for the advection equation in 3-D, *Comput. Methods Appl. Mech. Eng.*, *191*(46), 5345–5363.

Kaazempur-mofrad, M., P. Minev, and C. Ethier (2003), A characteristic/finite element algorithm for time-dependent 3-D advection-dominated transport using unstructured grids, *Comput. Methods Appl. Mech. Eng.*, *192*(11–12), 1281–1298.

Malevsky, A., and S. Thomas (1997), Parallel algorithms for semi-Lagrangian advection, *Int. J. Numer. Methods Fluids*, *25*(4), 455–473.

Martin, N., and S. Gorelick (2005), Semi-analytical method for departure point determination, *Int. J. Numer. Methods Fluids*, *47*(2), 121–137.

Spiegelman, M., and P. B. Kelemen (2003), Extreme chemical variability as a consequence of channelized melt transport, *Geochem. Geophys. Geosyst.*, *4*(7), 1055, doi:10.1029/2002GC000336.

Spiegelman, M., P. B. Kelemen, and E. Aharonov (2001), Causes and consequences of flow organization during melt transport: The reaction infiltration instability in compactible media, *J. Geophys. Res.*, *106*(B2), 2061–2077.

Staniforth, A., and J. Cote (1991), Semi-Lagrangian integration schemes for atmospheric models—A review, *Mon. Weather Rev.*, *119*(9), 2206–2223.

van Keken, P., C. Currie, S. King, and R. Katz (2005), A benchmark for the modeling of subduction zones, *Eos Trans. AGU*, *86*(52), Fall Meet. Suppl., Abstract T33C-0560.