in Proceedings of 7th SIAM Conf. on Parallel Processing, 1995, 118–124

Chapter 1
Parallel Algorithms for Semi-Lagrangian Transport in Global Atmospheric Circulation Models

J. B. Drake
*Oak Ridge National Laboratory**

I. T. Foster
Argonne National Laboratory†

J. Michalakes
Argonne National Laboratory†

P. H. Worley
*Oak Ridge National Laboratory**

Abstract

Global atmospheric circulation models (GCM) typically have three primary algorithmic components: columnar physics, spectral tranform, and semi-Lagrangian transport. In developing parallel implementations, these three components are equally important and can be examined somewhat independently. A two-dimensional horizontal data decomposition of the three-dimensional computational grid leaves all physics computations on processor, and the only efficiency issues arise in load balancing. A recently completed study by the authors of different approaches to parallelizing the spectral transform showed several viable algorithms. Preliminary results of an analogous study of algorithmic alternatives for parallel semi-Lagrangian transport are described here.

1 Introduction

The semi-Lagrangian transport (SLT) method is used in GCMs, for example, to advect moisture. The method updates the value of the moisture field at a grid point (the arrival point, A) by first establishing a trajectory through which the particle arriving at A has moved during the current timestep ($2\Delta t$). This trajectory is found iteratively using the interpolated velocity field at the mid-point, M, of the trajectory. From this mid-point the departure point, D, is calculated and the moisture field is interpolated at D using shape preserving interpolation.

Like the spectral tranform method [2], there are a variety of parallel algorithms for semi-Lagrangian transport, in two major classes: distributed and transpose. All the calculations involve the same computational grid as the columnar physics, and are initially decomposed in the same way, a block decomposition of the latitude and longitude dimensions with the vertical dimension left undecomposed. In a distributed algorithm, this decomposition is retained. If information is needed from a neighboring processor in order to calculate the trajectory or to evaluate the moisture field at the departure point, it must be acquired from the "owner". There are many ways of doing this, including (a) initializing a sufficiently large overlap region at the beginning (allowing subsequent computations to be independent), (b)

*This research was supported by the Atmospheric and Climate Research Division, Office of Energy Research, U.S. Department of Energy under contract DE-AC05-84OR21400 with Martin Marietta Energy Systems Inc.

†This research was supported by the Atmospheric and Climate Research Division of the Office of Energy Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

requesting only what is needed when it is needed, and (c) hybrid algorithms that attempt to estimate exactly how much overlap is needed, possibly from past history, and request any information that is needed that is not in the overlap region.

Transpose algorithms remap the computational grid onto the processors, decomposing the vertical dimension and undecomposing the longitudinal (east-west) dimension. Since the wind blows primarily in the longitudinal direction, this minimizes the amount of nonlocal information that is needed in the SLT. Either overlap or by-request approaches can be used to handle data requirements in the other directions. How the SLT algorithm choice can be integrated with the spectral transform algorithm is also of interest.

While some of these parallel SLT algorithms have been examined by different researchers, a comprehensive comparison has not previously been attempted. In this study, we use the same methodology developed for the evaluation of parallel spectral transform methods [2]:

1. Implement all parallel algorithms in a code that solves the shallow water equations on a sphere, using fictitious vertical levels to obtain the correct granularity for three-dimensional GCMs.
2. Implement runtime options for each algorithm for selecting communication protocols, data decomposition, and other algorithm variants.
3. Use runtime options to optimize each algorithm on each platform before comparing between algorithms.

Initial studies will be performed on the Intel iPSC/860 and Paragon multiprocessors, but the targets of interest also include the IBM SP-2, nCUBE/2S, and Cray T3D.

In the following, we describe the semi-Lagrangian transport algorithm in the context of solving the shallow water equations, and discuss how this changes the comparisons made in earlier studies of parallel shallow water equations solvers. Parallel implementation details and empirical results will be presented during the oral presentation.

2 Shallow Water Equations on the Sphere

The shallow water equations in advective form are

$$\frac{d\mathbf{v}}{dt} = -f\vec{k} \times \mathbf{v} - \nabla\phi \ , \tag{1}$$

$$\frac{d\phi}{dt} = -\phi\nabla\cdot\mathbf{v} \tag{2}$$

where the substantial derivative is given by

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v}\cdot\nabla \tag{3}$$

and the gradient operator is given by

$$\nabla = \frac{\vec{i}}{a\cos\theta}\frac{\partial}{\partial\lambda} + \frac{\vec{j}}{a}\frac{\partial}{\partial\theta}. \tag{4}$$

The vector $\mathbf{v} = u\vec{i} + v\vec{j}$ is the horizontal vector velocity. The free surface geopotential is denoted by $\phi = gh$, where g is the gravitational acceleration and h is the height of the free surface. The Coriolis parameter $f = 2\Omega\sin\theta$ incorporates the effect of rotatation of the sphere with angular velocity Ω at latitude θ. The equations require no boundary conditions but are posed with initial conditions on $\mathbf{v}$ and ϕ.

3 Semi-Implicit, Semi-Lagrangian Formulation

A three time-level semi-Lagrangian method for the shallow water equations has been described in Ritchie [4]. An extension of this using spatial averages in semi-implicit, semi-Lagrangian schemes is discussed in [5]. For meterological models the form of their method is based on a division of terms in the equations between those involved in the fast moving waves, e.g. gravity waves, and the slower moving waves such as the Rossby waves. Let G denote the fast moving wave terms and R the slower wave terms. Then an advected scalar field F is governed by the equation

$$\frac{dF}{dt} + G = R. \tag{5}$$

Let subscripts $()_A$, $()_M$ and $()_D$ represent quantities evaluated at the arrival point, mid-point and departure point, respectively, as described in the introduction. Let superscripts $n-1$, n and $n+1$ represent the time-levels at which quantites are evaluated. Then the spatially-averaged semi-implicit, semi-Lagrangian scheme for equation (5) is,

$$\frac{F_A^{n+1} - F_D^{n-1}}{2\Delta t} + \frac{G_A^{n+1} + G_D^{n-1}}{2} = \frac{R_A^n + R_D^n}{2}. \tag{6}$$

Applying this form to the momentum equation (1), where the gravity wave component is identified with the $\nabla\phi$ term and the Coriolis term is identified with the R, and rearranging terms gives

$$\mathbf{v}_A^{n+1} + \Delta t \nabla \phi'^{n+1}_A = \mathbf{v}_D^{n-1} - \Delta t[(f\vec{k} \times \mathbf{v})_A^n + (f\vec{k} \times \mathbf{v})_D^n + \nabla \phi'^{n-1}_D] \equiv N_{\mathbf{v}}, \tag{7}$$

and

$$\phi'^{n+1}_A + \Delta t \bar{\phi} \nabla \cdot \mathbf{v}_A^{n+1} = \phi'^{n-1}_D - \Delta t[(\phi' \nabla \cdot \mathbf{v})_A^n + (\phi' \nabla \cdot \mathbf{v})_D^n + \bar{\phi} \nabla \cdot \mathbf{v}_D^{n-1}] \equiv N_\phi, \tag{8}$$

where $\phi = \phi' + \bar{\phi}$ and $\bar{\phi}$ is a constant reference value.

To avoid pole problems with the vector representation of the velocity and to maintain compatibility with the scalar spectral transform method used to calculate the remaining differential terms, we introduce the vorticity $\zeta \equiv k \cdot \nabla \times \mathbf{v}$ and the divergence $\delta \equiv \nabla \cdot \mathbf{v}$. The coordinate system used is the $\lambda - \mu$ spherical coordinates where $\mu = \sin\theta$ and θ is the co-latitude. We define $U = u\cos\theta$ and $V = v\cos\theta$. Let $N_U = \vec{i} \cdot N_{\mathbf{v}} \cos\theta$ and $N_V = \vec{j} \cdot N_{\mathbf{v}} \cos\theta$.

Taking the curl $(k \cdot \nabla \times \mathbf{v})$ and divergence $(\nabla \cdot)$ of the discrete momentum equation (7) gives the following equations:

$$\zeta_A^{n+1} = \frac{1}{a(1-\mu^2)} \frac{\partial(N_V)}{\partial \lambda} - \frac{1}{a} \frac{\partial(N_U)}{\partial \mu}, \tag{9}$$

$$\delta_A^{n+1} + \Delta t \nabla^2 \phi'^{n+1}_A = \frac{1}{a(1-\mu^2)} \frac{\partial(N_U)}{\partial \lambda} + \frac{1}{a} \frac{\partial(N_V)}{\partial \mu}. \tag{10}$$

4 Spectral Semi-Implicit Formulation

The combination of the semi-Lagrangian method with the spectral transform method has proven to be a powerful technique, providing easy solution of the semi-implicit equations, high spatial accuracy, and effectively double the resolution of a pure spectral transform method with the same cost.

The spectral transform, or more correctly, the spherical harmonic transform, is based on the representation of scalar fields as a linear combination of spherical harmonics as

$$\xi(\lambda,\mu) = \sum_{m=-M}^{M} \sum_{n=|m|}^{N(m)} \xi_n^m P_n^m(\mu) e^{im\lambda}. \tag{11}$$

Once spectral coefficients are known, the transformation from spectral to physical space is accomplished using this formula, which is usually referred to as harmonic synthesis. The transform from physical space to spectral space (harmonic analysis) uses a discrete version of the continuous transform defined by

$$\xi_n^m = \int_{-1}^{1} \frac{1}{2\pi} \int_0^{2\pi} \xi(\lambda,\mu) e^{-im\lambda} d\lambda P_n^m(\mu) d\mu. \tag{12}$$

Returning to the derivation of the numerical method, a spectral analysis is used next. Dropping the primes from ϕ, the spectral forms of equations (9) and (10) are

$$(\zeta_n^m)_A^{n+1} = \sum_{j=1}^{J} [im(N_V)^m(\mu_j) P_n^m(\mu_j) + (N_U)^m(\mu_j) H_n^m] \frac{w_j}{a(1-\mu_j^2)} \tag{13}$$

$$\begin{aligned} (\delta_n^m)_A^{n+1} - \Delta t \frac{n(n+1)}{a^2} (\phi_n^m)_A^{n+1} &= \sum_{j=1}^{J} [im(N_U)^m(\mu_j) P_n^m(\mu_j) \\ &- (N_V)^m(\mu_j) H_n^m(\mu_j)] \frac{w_j}{a(1-\mu_j^2)}. \end{aligned} \tag{14}$$

(Note the use of n for both time level and spectral index. It will be clear from context which meaning to attribute to the variable in subsequent discussions.) The spectral form of the continuity equation (8) is

$$(\phi_n^m)_A^{n+1} + \Delta t \bar{\phi} (\delta_n^m)_A^{n+1} = \sum_{j=1}^{J} (N_\phi)^m(\mu_j) P_n^m(\mu_j) \frac{w_j}{a(1-\mu_j^2)}. \tag{15}$$

Equations (14) and (15) can be solved together for the advanced time-level values in spectral space of δ_n^m and ϕ_n^m. For each mode, a 2x2 system of equations is solved:

$$\begin{bmatrix} 1 & -\Delta t \frac{n(n+1)}{a^2} \\ \Delta t \bar{\phi} & 1 \end{bmatrix} \begin{bmatrix} (\delta_n^m)_A^{n+1} \\ (\phi_n^m)_A^{n+1} \end{bmatrix} = \begin{bmatrix} R \\ Q \end{bmatrix}, \tag{16}$$

where R and Q are the righthand sides of (14) and (15), respectively. This is the same system found in the semi-implicit spectral transform method of Hack [3].

With ζ_n^m, δ_n^m and ϕ_n^m at the arrival points at the new time-level, the synthesis (inverse spherical harmonic transform) is used to obtain the values of ϕ and $\mathbf{v}$ in physical space. The components of $\mathbf{v}$ are obtained through the diagnostic relationship in terms of ζ_n^m and δ_n^m,

$$U(\lambda_i,\mu_j) = -\sum_{m=-M}^{M} \sum_{n=|m|}^{N(m)} \frac{a}{n(n+1)} [im\delta_n^m P_n^m(\mu_j) - \zeta_n^m H_n^m(\mu_j)] e^{im\lambda_i} \tag{17}$$

and

$$V(\lambda_i,\mu_j) = -\sum_{m=-M}^{M} \sum_{n=|m|}^{N(m)} \frac{a}{n(n+1)} [im\zeta_n^m P_n^m(\mu_j) + \delta_n^m H_n^m(\mu_j)] e^{im\lambda_i}. \tag{18}$$

5 Semi-Lagrangian Interpolations

Following Williamson [6], $N_{\mathbf{V}}$ can be written as

$$N_{\mathbf{V}} = N_A^1 \vec{i}_A + N_A^2 \vec{j}_A + N_D^1 \vec{i}_D + N_D^2 \vec{j}_D \ . \tag{19}$$

In this expression $\vec{i}_A$ is the unit vector in the first component direction at the arrival point, and similarly for the other directional unit vectors. As shown by Bates *et. al.* [1], the relation between the direction vectors at the arrival point and the departure point can be expressed by equations of the form:

$$\begin{aligned} \vec{i}_D &= \alpha_1 \vec{i}_A + \beta_1 \vec{j}_A \\ \vec{j}_D &= \alpha_2 \vec{i}_A + \beta_2 \vec{j}_A, \end{aligned} \tag{20}$$

where the α's and β's are solely functions of the coordinates of the arrival and departure points. Using these expressions we can express equation (19) by

$$\begin{aligned} N_U &= N_A^1 + \alpha_1 N_D^1 + \alpha_2 N_D^2 \\ N_V &= N_A^2 + \beta_1 N_D^1 + \beta_2 N_D^2 \end{aligned} \tag{21}$$

Thus, in order to form $N_{\mathbf{V}}$ and N_ϕ, the following quantities in physical space must be interpolated at the departure points: $\mathbf{v}^{n-1}$, $\mathbf{v}^n$, ϕ^n, ϕ^{n-1}, δ^n, δ^{n-1}, and $\nabla\phi^{n-1}$.

The interpolations are calculated using a shape preserving tensor product formulation interprolation scheme developed by Williamson and Rasch [7]. The departure point calculation integrates the equation

$$\frac{d\mathbf{x}}{dt} = -\mathbf{v} \tag{22}$$

backwards in time along the trajectory from the arrival point. This trajectory calculation is done in the standard way using an iteration for the midpoint and evaluating the velocities at that midpoint using a Lagrangian interpolant (non-shape preserving).

6 Algorithm Outline

With values of $\mathbf{v}$, ϕ, $\nabla\phi'$, and δ at time-levels n and $n-1$ in physical space, the following steps advance to the new state at time-level $n+1$.

1. For each arrival point
   (a) Calculate the departure point (λ_D, θ_D), interpolating $\mathbf{v}_M$.
   (b) Interpolate the 9 field values, evaluate the α's and β's used in equation (21), and calculate all N's.
2. Fourier transform N_U, N_V and N_ϕ.
3. Evaluate R, Q and $(\zeta_n^m)_A^{n+1}$ using equations (13-15).
4. Solve the 2×2 semi-implicit system in spectral space for $(\delta_n^m)_A^{n+1}$ and $(\phi_n^m)_A^{n+1}$
5. Perform an inverse transform (synthesis) to calculate U, V and ϕ. Also inverse transform to get $\nabla\phi'$ and δ.

In order to parallelize this algorithm, we must parallelize each step in turn. Steps 2-5 must also be parallelized in the parallel spectral transform method analyzed in [2], and code already exists for these phases of the algorithm. Step 1 is independent for each arrival point, but the data needed for the calculation may not be resident in the local memory of the processor calculating the N's for this location, as discussed in the introduction. Thus, these experiments are primarily concerned with minimizing the communication requirements of the parallel SLT algorithm, and the interaction of decisions made in the parallel SLT with the efficiency of the parallel spectral transform components in the code.

Note that the code analogous to Step 1 in the pure spectral transform method for the shallow water equations is perfectly parallel given a two-dimensional horizontal decomposition, requiring no interprocessor communication. However, it also requires calculating forward Fourier transforms for 8 fields, instead of the 3 transformed here. (Both algorithms bring back 5 fields to physical space.) The SLT algorithm also allows longer timesteps to be taken, decreasing the number of timesteps that must be calculated.

7 Summary

The SLT algorithm is an important component of global atmospheric circulation models, and will be even more important in the next generation of models. Increased usage of the SLT technique promises both improved numerical accuracy and longer timesteps. However, effective utilization of the technique requires that efficient parallel implementations be developed. Since the SLT is only one component in the models, it is important that these implementations be evaluated in the context in which they will be used. During our talk, we will report on our evaluation of the various approaches.

References

[1] J. R. Bates, F. H. M. Semazzi, R. W. Higgins, and S. Barros, *Integration of the shallow water equations on the sphere using a vector semi-Lagrangian scheme with a multigrid solver*, Mon. Wea. Rev., 118 (1990), pp. 1615–1627.

[2] I. T. Foster and P. H. Worley, *Parallel algorithms for the spectral transform method*, Tech. Rep. ORNL/TM–12507, Oak Ridge National Laboratory, Oak Ridge, TN, May 1994.

[3] J. Hack and R. Jakob, *Description of a global shallow water model based on the spectral transform method*, NCAR Technical Note NCAR/TN-343+STR, National Center for Atmospheric Research, Boulder, CO, 1992.

[4] H. Ritchie, *Application of the semi-Lagrangian method to a spectral model for the shallow water equations*, Mon. Wea. Rev., (1988), pp. 1587–1598.

[5] M. Tanguay, E. Yakimiw, H. Ritchie, and A. Robert, *Advantages of spatial averaging in semi-implicit semi-Lagrangian schemes*, Mon. Wea. Rev., (1992), pp. 113–123.

[6] D. L. Williamson and J. Olson, *Climate simulations with a semi-Lagrangian version of the NCAR Community Climate Model*, Mon. Wea. Rev., (1994), pp. 1594–1610.

[7] D. L. Williamson and P. J. Rasch, *Two-dimensional semi-Lagrangian transport with shape-preserving interpolation*, Mon. Wea. Rev., 117 (1989), pp. 102–129.