



Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

A Particle-Mesh Method for the Shallow Water Equations Near
Geostrophic Balance

J. Frank, S. Reich

Modelling, Analysis and Simulation (MAS)

MAS-R0101 February 28, 2001

Report MAS-R0101
ISSN 1386-3703

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

CWI is the National Research Institute for Mathematics and Computer Science. CWI is part of the Stichting Mathematisch Centrum (SMC), the Dutch foundation for promotion of mathematics and computer science and their applications.

SMC is sponsored by the Netherlands Organization for Scientific Research (NWO). CWI is a member of ERCIM, the European Research Consortium for Informatics and Mathematics.

Copyright © Stichting Mathematisch Centrum
P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

A Particle-Mesh Method for the Shallow Water Equations Near Geostrophic Balance

Jason Frank

CWI

P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

e-mail: jason@cwi.nl

Sebastian Reich

Department of Mathematics, Imperial College

180 Queen's Gate, London, SW7 2BZ

e-mail: s.reich@ic.ac.uk

ABSTRACT

In this paper we outline several new particle-mesh methods that conserve potential vorticity (PV) in geophysical flows. The new methods are based on an operator splitting of the shallow water equations into an Eulerian gravity wave part and a Lagrangian advection part, each of which exactly preserves PV. Some of these methods are tested and compared with standard pseudospectral methods, based on qualitative features of the solutions obtained (i.e. dispersion or smoothness of PV contours), as well as quantitative measures of energy and generalized enstrophy preservation, and increase in mean divergence level. Based on these comparisons we suggest several directions for further research.

2000 Mathematics Subject Classification: 76M28 76B60

Keywords and Phrases: geophysical fluid dynamics, potential vorticity conserving methods, geometric methods, particle-mesh methods

Note: Work of the first author carried out under project MAS 1.1 - 'Atmospheric Flow and Transport Problems.' Partial support of this work by GMD is gratefully acknowledged.

1. INTRODUCTION

The dynamics of the atmosphere (and ocean) is characterized by the existence of motion on two scales: on the one hand is the relatively slow advection of turbulent structures, and on the other hand is the relatively fast motion of atmospheric gravity waves. The interaction of these types of motion is the subject of much current research in geophysical fluid dynamics. We expect that their proper numerical treatment is crucial both to an understanding of the motions in their own right and for obtaining meaningful results from long time simulations, for example, in climate studies.

The complete dynamics of the atmosphere are given by the three-dimensional primitive equation model. However, a simplified model which still retains much of the important dynamics of geophysical fluids is the rotating shallow water equations (SWEs):

$$\frac{d}{dt}\mathbf{u} = -f_0\mathbf{e}_z \times \mathbf{u} - c_0\nabla_{\mathbf{x}}\eta, \quad (1.1)$$

$$\frac{d}{dt}\eta = -(H_0 + \eta)\nabla_{\mathbf{x}} \cdot \mathbf{u}, \quad (1.2)$$

where $\mathbf{u} = (u, v)^T$ is the horizontal velocity field, η is the layer depth variation from a mean value of H_0 (i.e. the total layer depth is $h = H_0 + \eta$), $f_0/2 > 0$ is the angular velocity of the reference plane around the unit normal \mathbf{e}_z , $c_0 > 0$ is an appropriate constant [13], and $\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{x}}$ is the material time derivative. In this paper, we consider the SWEs over a periodic domain $(x, y) \in$

$[-\pi, +\pi] \times [-\pi, +\pi]$ with mean layer-depth $H_0 = 1$ and Rossby radius $L_R = \sqrt{c_0 H_0}/f_0 = 0.5$. This scaling essentially leaves f_0 a free parameter.

A dynamical quantity of significant import in geophysical fluid dynamics is the potential vorticity (PV)

$$q = \frac{\omega + f_0}{h}, \quad \omega = v_x - u_y = \nabla_{\mathbf{x}} \times \mathbf{u},$$

which is constant along particle trajectories; i.e. $dq/dt = 0$. The importance attached to PV in atmospheric dynamics can be seen in its central role in *quasigeostrophic theory*. In extra-tropical regions, the terms on the right hand side of (1.1) are nearly in balance. This motivates the definition of the *geostrophic wind*:

$$\mathbf{u}_g = \frac{c_0}{f_0} \mathbf{e}_z \times \nabla_{\mathbf{x}} \eta. \quad (1.3)$$

Note that if we assume the geostrophic wind, then all of the dynamic variables can be recovered from the PV distribution via

$$q(H_0 + \eta) = \frac{c_0}{f_0} \nabla_{\mathbf{x}}^2 \eta + f_0 \quad (1.4)$$

and (1.3). Furthermore, the PV itself is advected under the geostrophic flow field:

$$\frac{\partial q}{\partial t} + \mathbf{u}_g \cdot \nabla_{\mathbf{x}} q = 0. \quad (1.5)$$

The combined system (1.3), (1.4) and (1.5) is referred to as the quasigeostrophic approximation [13].¹

From a computational viewpoint it is important to notice that PV serves as a main organizing quantity of geostrophic flows. Accurate advection of the PV field is therefore of primary importance. This has been demonstrated using the contour-advective semi-Lagrangian (CASL) algorithm of Dritschel et al. [4]. Their approach is to advect the PV field along Lagrangian particles that form contour lines of constant PV. The time evolution of the divergence

$$\delta = \nabla_{\mathbf{x}} \cdot \mathbf{u}$$

and the layer-depth h are computed over an Eulerian grid using a hierarchy of nonlinear balance conditions [11]. The contour-advection schemes have been shown to result in a higher PV-field resolution compared to classical pseudospectral and semi-Lagrangian methods [11].

In this article we are mainly interested in problems near geostrophic balance, i.e., those for which the local Rossby number [11] satisfies

$$Ro := \frac{|\omega|}{f_0} \ll 1$$

and asymptotic theory applies. See §3 and, e.g., [10]. Hence, one also has

$$h \approx \frac{f_0}{q}. \quad (1.6)$$

The main computational challenge is the coexistence-existence of fast (small amplitude) non-balanced motion and slow motion in *geostrophic balance* [13, 2, 9]. The geostrophic wind (1.3) is divergence-free. In contrast, the generation of (fast) unbalanced gravity waves is characterized by the divergence δ . We assume throughout the paper that $|\delta|$ is small. Our main interest is in the long-time dynamics of such almost balanced fluid motion. We emphasize as well that our interest is in smooth solutions that do *not* develop into shocks.

¹Strictly speaking, the quasigeostrophic approximation makes use of a linearization of (1.4) to recover the geostrophic layer depth from PV.

In this paper, we suggest an operator splitting of the SWEs—into a linear wave equation and an advection step—that takes the importance of PV conservation into account and that can be implemented using an appropriate modification of a *particle-mesh* (PM) [7] or *particle-in-cell* (PIC) method [6]. Contrary to contour-advection, we advect both the velocity field *and* PV using Lagrangian particle dynamics. The linear wave equation is solved over a fixed Eulerian grid. The overall method is time-reversible and exactly conserves PV along Lagrangian particle motion. It requires the solution of a linear system of equations to transform the data from the Lagrangian particles to the Eulerian grid representation. The main feature of the new method, as demonstrated by a series of numerical experiments, is the excellent conservation of the total energy

$$\mathcal{E} = \frac{1}{2} \int [hu^2 + hv^2 + c_0 h^2] d\mathbf{x} \quad (1.7)$$

and the PV-momenta (*generalized enstrophies*)

$$\bar{q}_s = \int h q^s d\mathbf{x}, \quad s = 1, 2, 3, \quad (1.8)$$

over the Eulerian (x, y) -domain.

2. A PV-CONSERVING GEOSTROPHIC SPLITTING

We assume (1.6) and introduce the *geostrophic layer-depth*

$$H_g = \frac{f_0}{q}. \quad (2.1)$$

Note that $H_g \approx h$ under the assumption (1.6).

The SWEs can then be split into an Eulerian wave equation

$$\mathbf{u}_t = -f_0 \mathbf{e}_z \times \mathbf{u} - c_0 \nabla_{\mathbf{x}} \eta, \quad (2.2)$$

$$\eta_t = -H_g \nabla_{\mathbf{x}} \cdot \mathbf{u}, \quad (2.3)$$

and a Lagrangian advective system

$$\frac{d}{dt} \mathbf{u} = \mathbf{0}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{u}. \quad (2.4)$$

The missing equation for the time evolution of the layer-variation η will be derived below.

Note that the system (2.2)–(2.3) is linear in (\mathbf{u}, η) and satisfies

$$q_t = \frac{v_{xt} - u_{yt}}{H_0 + \eta} - \frac{q}{H_0 + \eta} \eta_t = -\frac{f_0}{H_0 + \eta} (u_x + v_y) + \frac{q}{H_0 + \eta} H_g (u_x + v_y) = 0.$$

This implies that PV is conserved and that equation (2.2) can be rewritten as

$$\mathbf{u}_t = -f_0 \mathbf{e}_z \times \mathbf{u} - c_0 \nabla_{\mathbf{x}} \frac{\omega}{q} - c_0 \nabla_{\mathbf{x}} H_g. \quad (2.5)$$

Since $q_t = 0$ under the flow (2.2)–(2.3), we must have, for the Lagrangian part (2.4),

$$\frac{d}{dt} q = 0, \quad h = \frac{\omega + f_0}{q}, \quad (2.6)$$

along particle paths. In summary, equation (2.4), supplemented with (2.6), describes the Lagrangian advection part of the splitting; whereas (2.5), along with (2.1) and $q_t = 0$, describes the Eulerian wave part.

For numerical purposes one can split the equations (2.5) even further into

$$\mathbf{u}_t = -c_0 \nabla_{\mathbf{x}} h, \quad h = \frac{\omega + f_0}{q}, \quad q_t = 0, \quad (2.7)$$

and

$$\mathbf{u}_t = -f_0 \mathbf{e}_z \times \mathbf{u}, \quad q_t = 0. \quad (2.8)$$

Up to spatial representation, the system (2.4) can be solved exactly in Lagrangian coordinates, while (2.7) and (2.8) can be solved exactly in Eulerian representation. Let us denote the associated solution operators by Ψ_t^L , Ψ_t^{E1} , and Ψ_t^{E2} , respectively. This suggests a simple time-reversible fractional time-step approach to the integration of the SWEs; e.g.,

$$\Psi := \Psi_{\Delta t/2}^L \circ \left[\Psi_{\delta t/2}^{E1} \circ \Psi_{\delta t}^{E2} \circ \Psi_{\delta t/2}^{E1} \right]^K \Psi_{\Delta t/2}^L, \quad (2.9)$$

where $\Delta t = \delta t \cdot K$, $K \geq 1$. To actually implement this method we need a way to transform from Eulerian to Lagrangian coordinates and back. This will be discussed in §4.

3. THE OPERATOR SPLITTING UNDER THE QUASI-GEOSTROPHIC LIMIT

Let us make a simple scaling analysis of the linear wave equation (2.2)–(2.3). We assume that

$$(i) \ f_0 = \varepsilon^{-1} F_0 \quad (\text{small Rossby number limit}).$$

We know that the geostrophic layer-depth H_g is constant along solutions of (2.2)–(2.3) and that $q = \mathcal{O}(\varepsilon^{-1})$. Hence

$$\eta_t = \varepsilon \xi_t, \quad \text{where} \quad \xi := \varepsilon^{-1} \frac{\omega}{q} = \frac{\eta + H_0 - H_g}{\varepsilon},$$

and we can rewrite (2.2)–(2.3) as

$$\varepsilon \mathbf{u}_t = -F_0 \mathbf{e}_z \times \mathbf{u} - \varepsilon c_0 \nabla_{\mathbf{x}} \eta = -F_0 \mathbf{e}_z \times \mathbf{u} - \varepsilon^2 c_0 \nabla_{\mathbf{x}} \xi - \varepsilon c_0 \nabla_{\mathbf{x}} H_g, \quad (3.1)$$

$$\varepsilon \xi_t = -H_g \nabla_{\mathbf{x}} \cdot \mathbf{u}. \quad (3.2)$$

The equations (3.1)–(3.2) are equivalent to

$$\varepsilon \mathbf{u}_t = -F_0 \mathbf{e}_z \times \mathbf{u} - \varepsilon^2 c_0 \nabla_{\mathbf{x}} \frac{\omega}{Q} - \varepsilon c_0 \nabla_{\mathbf{x}} H_g,$$

$$\varepsilon Q_t = 0,$$

where $Q = \varepsilon q$.

A constant Rossby radius $L_R = \sqrt{c_0 H_0}/f_0$ together with (i) implies a low Froude number scaling and leads to the additional assumptions

$$(ii) \ c_0 = \varepsilon^{-2} C_0 \quad \text{and} \quad (iii) \ \nabla_{\mathbf{x}} H_g = \mathcal{O}(\varepsilon).$$

These conditions guarantee that the right hand side of the momentum equation (3.1) remains bounded as $\varepsilon \rightarrow 0$. They also imply that $H_g = H_0$ as $\varepsilon \rightarrow 0$ and $\eta = \mathcal{O}(\varepsilon)$. But more can be said. Smooth solutions of (3.1)–(3.2); i.e., solutions with $(\mathbf{u}_t, \xi_t) = \mathcal{O}(1)$; require balanced initial conditions

$$f_0 \mathbf{e}_z \times \mathbf{u}(0) + c_0 \nabla_{\mathbf{x}} \eta(0) = \mathcal{O}(\varepsilon^0) \quad \text{and} \quad \nabla_{\mathbf{x}} \cdot \mathbf{u}(0) = \mathcal{O}(\varepsilon), \quad (3.3)$$

which follows, e.g., from Kreiss' bounded time derivative principle [1]. These balance conditions are satisfied by the geostrophic velocity \mathbf{u}_g .

The main motivation for this paper is the efficient computation of solutions with initial data in almost geostrophic balance (3.3). To see what this assumption implies for our operator splitting method, let us, formally, take the limit $\varepsilon \rightarrow 0$ in the above analysis of the linear wave equation. This leads to $\mathbf{u} = \mathbf{u}_g$ and the Lagrangian part of the splitting gives rise to the “slow” PV advection equation

$$q_t + \mathbf{u}_g \cdot \nabla_{\mathbf{x}} q = 0. \quad (3.4)$$

Compare with the quasigeostrophic approximation (1.3), (1.4) and (1.5).

4. NUMERICAL IMPLEMENTATION

A popular approach for solving advection dominated fluid problems is to apply a *particle-mesh* (PM) or *particle-in-cell* (PIC) spatial discretization. Both methods work with a fixed spatial grid $\{\mathbf{X}_{ij}\}$, $i, j = 1, \dots, N$, $\mathbf{X}_{ij} \in \mathbb{R}^2$, and a set of moving particles $\{\mathbf{x}_k\}$, $k = 1, \dots, M$, $\mathbf{x}_k \in \mathbb{R}^2$. One typically assumes that $M \gg N^2$. Whenever a dependent variable $z(\mathbf{x}) \in \mathbb{R}$ is approximated over the fixed grid $\{\mathbf{X}_{ij}\}$, then the approximations $\{Z_{ij}\} \in \mathbb{R}^{N \times N}$, $Z_{ij} \approx z(\mathbf{x} = \mathbf{X}_{ij})$, are called Eulerian. Similarly, if $z(\mathbf{x})$ is approximated over particle positions $\{\mathbf{x}_k\}$, then the values $\{z_k\} \in \mathbb{R}^M$, $z_k \approx z(\mathbf{x} = \mathbf{x}_k)$, provide a Lagrangian approximation. One clearly also needs interpolation operators $\mathcal{A} : E \rightarrow L$ and $\mathcal{B} : L \rightarrow E$ between the Eulerian space $E = \mathbb{R}^{2 \times N \times N}$ and the Lagrangian space $L = \mathbb{R}^{2 \times M}$.

We now outline a PM method for the PV-conserving splitting described in §2. We introduce M Lagrangian (moving) particles with locations $\{\mathbf{x}_k\}$, velocities $\{\mathbf{u}_k\}$, and potential vorticities $\{q_k\}$. Once initialized the values $\{q_k\}$ are fixed for the entire integration. Solution of (2.4) is exact:

$$\mathbf{x}_k(t + \Delta t) = \mathbf{x}_k(t) + \Delta t \mathbf{u}_k, \quad k = 1, \dots, M.$$

The velocities $\{\mathbf{u}_k\}$ satisfy the “conservation law”

$$\frac{d}{dt} \mathbf{u}_k = \mathbf{0}.$$

during the Lagrangian step. This completes the numerical treatment of the Lagrangian advection step (2.4), (2.6). In the next section we consider solution of the wave equation (2.5) on an Eulerian grid.

4.1 Spatial discretization of the linear wave equation

Let us outline the spatial truncation of the linear wave equation (2.5). We introduce a rectangular (fixed) Eulerian grid $\mathbf{X}_{ij} = (X_{ij}, Y_{ij})$,

$$X_{ij} = X_0 + i \cdot \Delta x, \quad Y_{ij} = Y_0 + j \cdot \Delta y, \quad i, j = 0, \dots, N-1,$$

and denote approximations to the velocity field $\mathbf{u} = (u, v)$, the layer-depth h , the PV field q by $\mathbf{U}_{ij} = (U_{ij}, V_{ij})$, H_{ij} , and Q_{ij} , respectively. In this paper, we always use $\Delta x = \Delta y$.

The Eulerian grid-based functions $\mathbf{U} = \{U_{ij}\}$, $\mathbf{V} = \{V_{ij}\}$, $\mathbf{Q} = \{Q_{ij}\} \in E$ are obtained at time level $t = t_n$ from their Lagrangian counterparts $\mathbf{u}, \mathbf{v}, \mathbf{q} \in L$ using the linear interpolation operator \mathcal{B} to be defined in the next section, i.e.

$$\mathbf{U} = \mathcal{B}\mathbf{u}, \quad \mathbf{V} = \mathcal{B}\mathbf{v}, \quad \mathbf{Q} = \mathcal{B}\mathbf{q}. \quad (4.1)$$

Equation (2.8) represents a decoupled set of harmonic oscillators that can be solved exactly at grid points (or on Lagrangian particles, for that matter). To solve (2.7), we define the Eulerian layer depth via

$$H_{ij} = \frac{\Omega_{ij} + f_0}{Q_{ij}}, \quad \text{where} \quad \{\Omega_{ij}\} = \boldsymbol{\Omega} = D_x \mathbf{V} - D_y \mathbf{U},$$

as an approximation to h . Here and elsewhere D_x and D_y represent spectral difference operators (but the same discussion holds for central finite differences as well). Note that under the flow of (2.7), both Ω_{ij} and Q_{ij} and therefore H_{ij} are constant. The spectral discretization of (2.7) is also trivially integrated in time. In fact, we can directly update the velocity components on the Lagrangian particles by using the interpolation operator \mathcal{A} and solving:

$$\mathbf{u}_t = -c_0 \mathcal{A} D_x \mathbf{H}, \quad \mathbf{v}_t = -c_0 \mathcal{A} D_y \mathbf{H}, \quad (4.2)$$

At this point our method is completely defined except for the interpolation operators \mathcal{A} and \mathcal{B} .

4.2 Interpolation between Eulerian and Lagrangian variables

We need linear interpolation operators $\mathcal{A} : E \rightarrow L$ and $\mathcal{B} : L \rightarrow E$ to transport the quantities between Lagrangian and Eulerian representations. Using (4.1), we can pre-multiply equations (4.2) by \mathcal{B} , obtaining:

$$\mathbf{U}_t = -c_0 \mathcal{B} \mathcal{A} D_x \mathbf{H}, \quad \mathbf{V}_t = -c_0 \mathcal{B} \mathcal{A} D_y \mathbf{H},$$

respectively. Now, Eulerian PV conservation is guaranteed if $\boldsymbol{\Omega}_t = D_x \mathbf{V}_t - D_y \mathbf{U}_t = 0$, which is the case if D_x and D_y commute and, additionally,

$$\mathcal{B} \mathcal{A} = \text{id}_E$$

over the Eulerian space E . In our numerical experiments, we first define the interpolation operator \mathcal{A} and derive \mathcal{B} as the solution operator to the least-square problem for Z :

$$\mathcal{A}Z = z \quad \Rightarrow \quad Z = \mathcal{B}z;$$

i.e., $\mathcal{B} = (\mathcal{A}^T \mathcal{A})^{-1} \mathcal{A}^T$. We implemented two pairs of interpolation operators, to be described next.

4.2.1 A Fourier series interpolation A very simple and accurate approach is to use Fourier transformation to interpolate from the Eulerian grid to the Lagrangian particles. We define the interpolation operator $\mathcal{A} : E \rightarrow L$ by (assuming a periodic domain $[-\pi, +\pi] \times [-\pi, +\pi]$)

$$z_k = \sum_{i,j} Z_{ij} e^{i(X_i - x_k)} e^{i(Y_j - y_k)}.$$

This method is quite expensive, however.

4.2.2 Bilinear interpolation For a more efficient approach, we first introduce a second, finer Eulerian grid $\{\bar{\mathbf{X}}_{mn}\}$, $m, n = 1, \dots, 2N$, with half the mesh width $\Delta x/2$. Standard FFT is used to interpolate the data from $\{\mathbf{X}_{ij}\}$ to $\{\bar{\mathbf{X}}_{mn}\}$. Once the values are given over the finer grid $\{\bar{\mathbf{X}}_{mn}\}$, bilinear interpolation is applied over each grid-cell to obtain approximate values on the Lagrangian particle locations $\{\mathbf{x}_k\}$, $k = 1, \dots, M$.

5. A PSEUDOSPECTRAL METHOD

For numerical comparison, we implemented a pseudospectral (PS) discretization of the SWEs in vorticity-divergence form:

$$\omega_t = -\nabla_{\mathbf{x}} \cdot [(\omega + f_0)\mathbf{u}], \quad (5.1)$$

$$\delta_t + a^2 \nabla_{\mathbf{x}}^2 h - f_0^2 h = f_0(\omega - f_0 h) + 2J(u, v) - \nabla_{\mathbf{x}} \cdot (\mathbf{u}\delta), \quad (5.2)$$

$$h_t + H_0 \delta = -\nabla_{\mathbf{x}} \cdot (\mathbf{u}h), \quad (5.3)$$

where $a^2 = c_0 H_0$ and $J(u, v) = u_x v_y - u_y v_x$, and the velocity field is uniquely determined from (ω, δ) . In (5.1)–(5.3) the terms on the left are responsible for (linear) gravity waves. Denote $\mathbf{w} = (\omega, \delta, \eta)^T \in \mathbb{R}^3$ and write (5.1)–(5.3) in the abstract form

$$\mathbf{w}_t = \mathbf{A}\mathbf{w} + \mathbf{f}(\mathbf{w}), \quad \mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -a^2 \nabla_{\mathbf{x}}^2 + f_0^2 \\ 0 & -1 & 0 \end{bmatrix} \quad (5.4)$$

Spatial derivatives are computed in Fourier space using an FFT, and the product of any two functions is computed in physical space. The time-discretization is done using the trapezoidal rule for gravity waves and leapfrog for advection (LF/TR):

$$\frac{\mathbf{w}^{n+1} - \mathbf{w}^{n-1}}{2\Delta t} = \mathbf{A} \frac{\mathbf{w}^{n+1} + \mathbf{w}^{n-1}}{2} + \mathbf{f}(\mathbf{w}^n). \quad (5.5)$$

This two-step method is started with one time step of an analogous implicit/explicit Euler step of size $\Delta t/2^K$, followed by K stationary applications of (5.5) each time restarting from the initial condition and doubling the stepsize.

It was also necessary to stabilize the time stepping procedure using a Robert-Asselin filter, see for example [5], and we further stabilized the spatial discretization by removing all $2\Delta x$ -frequency modes after each step.

To obtain a smooth PV field it is usually necessary to include a small hyperviscosity term in the vorticity equation, replacing (5.1) with

$$\omega_t - \nu (\nabla_{\mathbf{x}}^2)^3 \omega = -\nabla_{\mathbf{x}} \cdot [(\omega + f_0)\mathbf{u}], \quad (5.6)$$

where the viscosity coefficient was taken to be:

$$\nu = \frac{H_0 Q}{(N/2)^6}, \quad \max_{\mathbf{x}} |q(\mathbf{x}) - f_0| \leq Q.$$

When hyperviscosity is included in the vorticity equation, the viscous term is discretized in time using implicit Euler differencing.

An entirely analogous implicit/explicit approach can be used with the SWEs in the form (1.1)–(1.2). In this case, the linearly implicit gravity wave part in (5.4) is

$$\mathbf{w}_t = \mathbf{A}\mathbf{w} + \mathbf{f}(\mathbf{w}), \quad \mathbf{A} = \begin{bmatrix} 0 & f_0 & -c_0 D_x \\ -f_0 & 0 & -c_0 D_y \\ -H_0 D_x & -H_0 D_y & 0 \end{bmatrix}, \quad (5.7)$$

where D_x and D_y represent pseudospectral difference operators. Hyperviscosity is applied to both u - and v -momentum equations (1.1).

Finally, one can replace equation (5.1) by the PV equation $\frac{d}{dt}q = 0$ and use $\omega := hq - f_0$. Similar to the contour-advection scheme of [4], we advect the constant PV field \mathbf{q} along Lagrangian particles. Using interpolation operators \mathcal{A} and \mathcal{B} , as introduced in §4, we obtain Eulerian PV-fields $\mathbf{Q}^{n+1} = \mathcal{B}^{n+1}\mathbf{q}$ and Lagrangian particle velocities $\mathbf{u}^n = \mathcal{A}^n\mathbf{U}^n$, $\mathbf{v}^n = \mathcal{A}^n\mathbf{V}^n$. Particles are advected by the second-order Adams-Bashforth method:

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k^n + \frac{\Delta t}{2} (3\mathbf{u}_k^n - \mathbf{u}_k^{n-1}).$$

6. NUMERICAL EXPERIMENTS

We consider a domain $\mathbf{x} \in [-\pi, \pi] \times [-\pi, \pi]$ with periodic boundary conditions. We use $f_0 = 4\pi$ and $c_0 = 4\pi^2$. The mean layer-depth is $H_0 = 1$. These parameter values correspond to a Rossby radius of $R_L = 0.5$. One rotation of the plane (one “day”) in physical time corresponds to one time unit in the computational model.

The initial conditions are defined as follows. We first introduce a PV field $\bar{q}(\mathbf{x})$. See below for specific choices. This field is then used to provide an initial layer-depth perturbation via

$$\Delta H_{ij} = \frac{f_0}{\bar{q}(\mathbf{X}_{ij})} - 1 + k_0,$$

where the constant k_0 is chosen such that $\sum_{ij} \Delta H_{ij} = 0$. Then the initial layer depth is given by

$$H_{ij} = 1 + \Delta H_{ij}$$

and the initial (purely geostrophic) Eulerian velocity field by

$$\mathbf{U}_{ij} = \frac{c_0}{f_0} \mathbf{e}_z \times \nabla_{\mathbf{x}} H_{ij}.$$

These initial values imply an Eulerian PV field $\{Q_{ij}\}$ which, together with the Eulerian velocities $\{(U_{ij}, V_{ij})\}$, can be mapped onto the Lagrangian particles $\{\mathbf{x}_k\}$ using the operator \mathcal{A} . The particles are initially placed on a uniform grid. The resulting motion is nearly in geostrophic balance.

A large part of the potential energy in (1.7) will be conserved by any numerical method conserving total mass. Specifically, if we write the potential energy as

$$\frac{1}{2} \int c_0 h^2 d\mathbf{x} = \frac{1}{2} \int c_0 h(h - H_0) d\mathbf{x} + \frac{1}{2} \int c_0 h H_0 d\mathbf{x},$$

then the last term on the right hand side will be exactly conserved in discrete form. To obtain a relative error in energy conservation which reflects the true available energy, therefore, we omit this term in defining the discrete total energy:

$$\mathcal{E}(t_n) = \frac{L^2}{2N^2} \sum_{i,j} H_{ij}(t_n) U_{ij}(t_n)^2 + H_{ij}(t_n) V_{ij}(t_n)^2 + c_0 H_{ij}(t_n) (H_{ij}(t_n) - H_0),$$

where $L = 2\pi$ is the domain length. We monitor the relative error in the total energy

$$\delta E(t_n) = \frac{\mathcal{E}(t_n) - \mathcal{E}(0)}{\mathcal{E}(0)},$$

as well as the relative error in the first few PV-momenta (1.8), which we discretize by

$$\bar{Q}_s(t_n) = \frac{L^2}{N^2} \sum_{i,j} H_{ij}(t_n) Q_{ij}(t_n)^s - f_0^s, \quad s = 1, 2, 3.$$

We also compute the L_2 -norm of the Eulerian divergence field

$$\Delta = D_x \mathbf{U} + D_y \mathbf{V}$$

as a measure of the ageostrophic component in the solution.

We used the second-order time-reversible fractional time-stepping method (2.9) with $\Delta t = 4 \cdot \delta t$. The overall scheme was implemented using MATLAB and a mex-subroutine for computing the bilinear interpolation.

6.1 Experiment A. Two-vortex interaction

As a simple test case, we define a PV field as a sum of Gaussian pulses

$$\bar{q}(x, y) = f_0 \left[1 + \sum_{\ell=1}^p \alpha_{\ell} \exp(-\beta_{\ell} \{(x - x_{\ell})^2 + (y - y_{\ell})^2\}) \right].$$

For this experiment we choose $p = 2$ and

$$\begin{aligned} \alpha_1 &= (4\pi)^{-1}, & \beta_1 &= 12/L, & x_1 &= 0.5, & y_1 &= 0.5, \\ \alpha_2 &= (4\pi)^{-1}, & \beta_2 &= 12/L, & x_2 &= -0.5, & y_2 &= -0.5, \end{aligned} \tag{6.1}$$

This field, representing two positively oriented vortices that are initially separated, is used to initialize the other variables as described in the previous section.

The simulations are run over a time interval $t \in [0, 30]$ using an Eulerian grid with $N = 16$ or $N = 32$.

6.1.1 Hyperdiffusion or reversibility? To begin with, we want to point out some problems with using a straightforward pseudospectral method as described in §5. In Fig. 1 we compare contour plots of the PV field at time $T = 30$ computed on a 64×64 grid without and with hyperdiffusion (5.6). In all contour plots the levels just above and below the value f_0 have been omitted to reduce some of the noise. Without hyperdiffusion the solution becomes quite noisy due to numerical dispersion (Fig. 1a). If integration is carried out on very long time intervals, the PV field becomes hopelessly lost in the noise. A small amount of hyperdiffusion serves to correct this (Fig. 1b); however, this comes at a cost of irreversibility, and energy decays as shown in Fig. 2. (There is also a small energy decay due to the Robert-Asselin Filter which can be observed in Fig. 2.) Such systematic decay of energy is also unacceptable for long time simulations.

The pseudospectral method described in §5 does nothing to prevent the aliasing of high frequency components in the computation of nonlinear terms. Antialiasing techniques can be used in place of the Robert-Asselin filter to maintain stability [5]. However, the energy loss is still present.

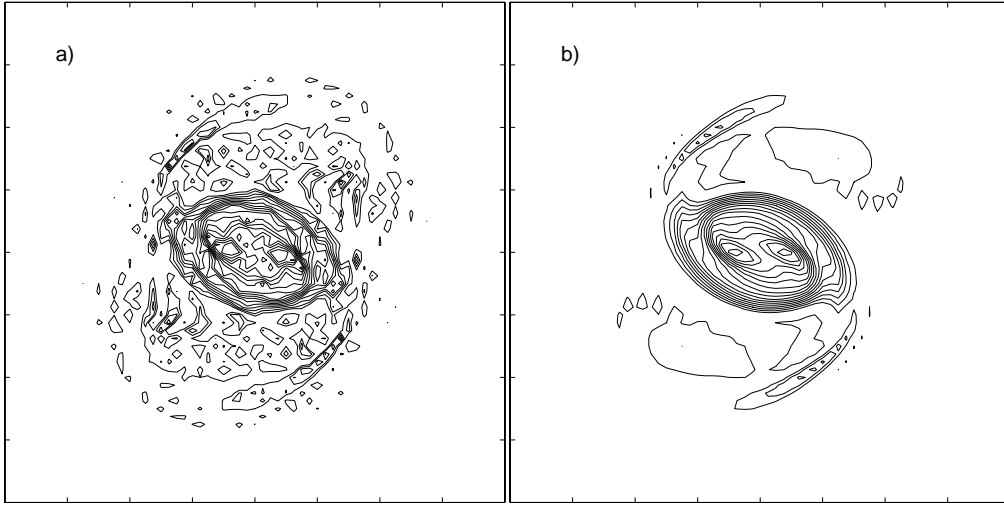


Figure 1: Pseudospectral solution of two-vortex interaction at time $T = 30$, computed on a 64×64 grid, a) without hyperdiffusion and b) with hyperdiffusion.

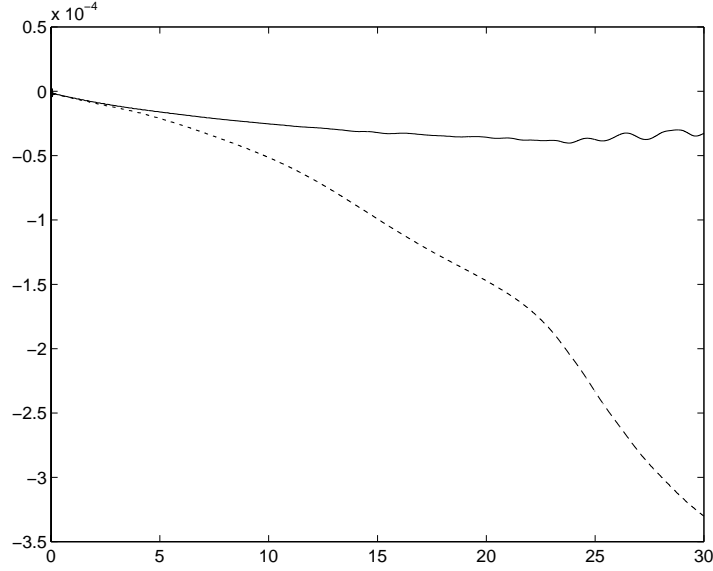


Figure 2: Relative error in total energy for PS method without (solid) and with (dashed) hyperdiffusion.

6.1.2 PVSPM with bilinear interpolation We solved the two-vortex interaction using the PVSPM method and bilinear interpolation between particles and grid. The simulation was run on a grid size of $N = 32$ with $M = 256^2$ particles. The Eulerian PV field \mathbf{Q} obtained at time $t = 30$ is shown in Fig. 3a. Compare the solution obtained here with the pseudospectral approximation on a grid $N = 64$ in Fig. 1. Note that the PV field does not suffer from the dispersion of the PS method Fig. 1a. However, some wiggles do appear in the contours which are not present in Fig. 1b. We believe the appearance of these wiggles is due to the crudeness of bilinear interpolation used during the simulation. Nonetheless, the resolution is good considering the relative coarseness of the mesh.

In Fig. 3b a 32^2 subset of particles is displayed along with the corresponding velocity field at the end of integration. It is remarkable that the method remains stable despite the drastic

displacements of the particles in the vicinity of the vortex interaction.

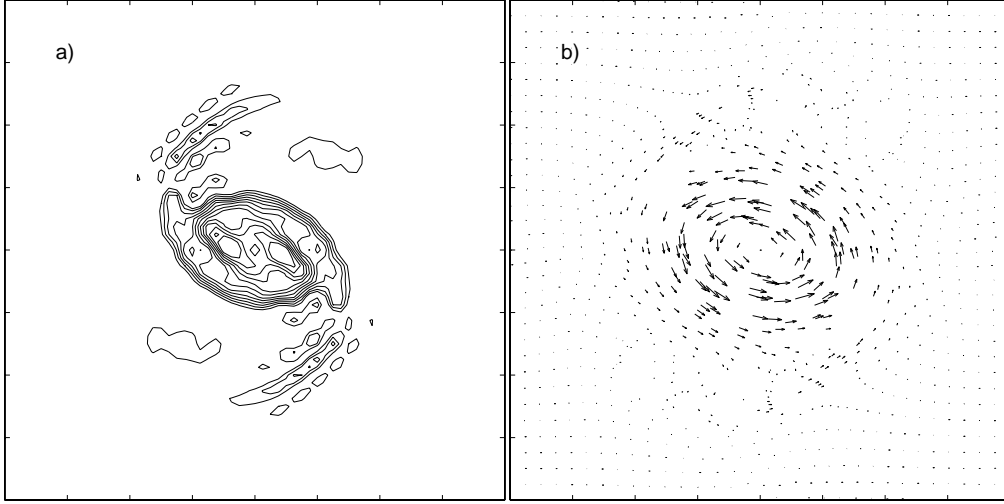


Figure 3: Solution at time $t = 30$ obtained with PVSPM method/bilinear interpolation, $N = 32$, $M = 256^2$ a) PV contours b) a 32^2 subset of particles with associated velocity vectors.

Next we compare the conservation properties of the PVSPM method with those of a PS method on a grid $N = 32$. The relative errors of total energy and the generalized enstrophies are given in Fig. 4. Note that \bar{Q}_1 is exactly conserved by the PS method. This is due to the fact that PV is a diagnostic variable for the PS method, and therefore $Q_{ij}H_{ij} = \Omega_{ij} + f_0$ holds by definition, and $\sum \Omega_{ij} = 0$ for any central difference operator and periodic boundary conditions. For the PVSPM method, on the other hand, q is a dynamic variable, so some error is not unexpected. In general we see that for all other conserved quantities in Fig. 4, PVSPM exhibits approximate conservation, whereas the PS method shows a systematic decay.

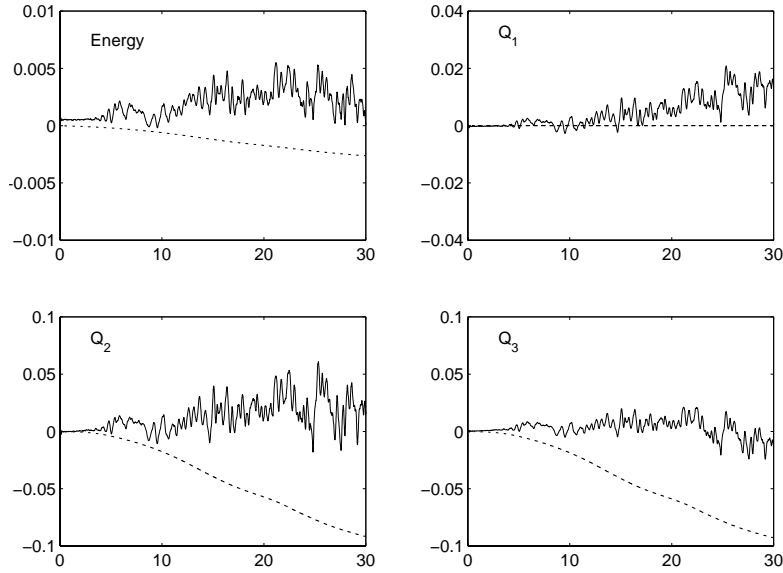


Figure 4: Relative errors in total energy and enstrophies for PVSPM (solid) and PS (dashed) methods.

6.1.3 Divergence If we now compare the norm of the divergence fields for the PVSPM and PS methods, Fig. 5, we see a discrepancy. Whereas the divergence as computed by the PS method remains nearly constant during the iteration, indicating that the amount of gravity wave activity is steady, the divergence computed by the PVSPM method grows linearly in time. Also shown in Fig. 5 is the solution obtained with PVSPM using Fourier interpolation on a $N = 16$ grid and 64^2 particles. In this case we observe no growth in the divergence field.² Again, we are led to conclude that much of the generation of gravity waves in this case is due to the nondifferentiability of the grid functions generated with bilinear interpolation. On the other hand, Fourier interpolation is too expensive to use on finer grids. We therefore suggest using a local, higher order interpolation, see §7.1.

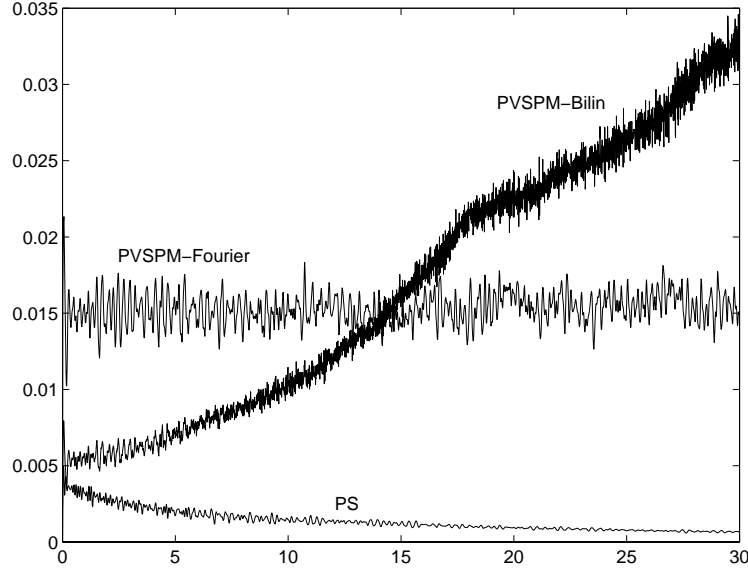


Figure 5: L_2 norm of the divergence field produced by PVSPM with bilinear and Fourier interpolation, as well as by PS.

6.1.4 Geostrophic scaling In this section we address the dependence of the PVSPM method on the assumption (1.6). To do so, we institute a scaling of the test parameters as suggested in §3, i.e. $f_0 \mapsto \varepsilon^{-1} f_0$, $c_0 \mapsto \varepsilon^{-2} c_0$ and the initial later depth disturbance $\Delta H_{ij} \mapsto \varepsilon^{-1} \Delta H_{ij}$. Under this scaling the large scale PV structures evolve at a rate independent of ε . In Fig. 6 we have plotted the norm of the divergence for PVSPM-Bilinear on $N = 16$ with 128^2 particles for $\varepsilon = 1, 2$ and 4. Observe that as the test conditions approach the quasigeostrophic limit, the rate of growth of the divergence field decreases.

6.2 Experiment B. “Peaks”

We consider a PV anomaly³

$$\begin{aligned} \Delta \bar{q}(x, y) = & 3(1-x)^2 \exp(-x^2 - (y+1)^2) - 10 \left(\frac{x}{5} - x^3 - y^5 \right) \exp(-x^2 - y^2) \\ & - \frac{1}{3} \exp(-(x+1)^2 - y^2), \end{aligned} \quad (6.2)$$

and define the PV field by

$$\bar{q}(x, y) = f_0 \left[1 + \frac{\Delta \bar{q}(1.2 \cdot x, 1.2 \cdot y)}{20 \cdot \pi} \right].$$

²The mean divergence level depends on the resolution. For a spectral method on $N = 16$ without hyperdiffusion or Asselin damping, the mean divergence level is about 0.008, somewhat lower than for PVSPM-Fourier.

³The function $\Delta \bar{q}$ is the one evaluated by the standard MATLAB m-file `peaks.m`.

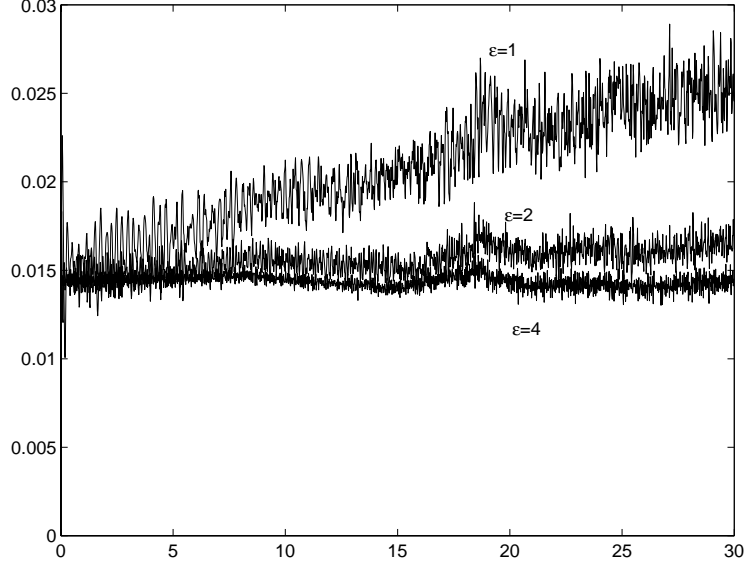


Figure 6: L_2 norm of the divergence field as a function of scaling parameter ε for PVSPM-Bilinear.

This field is used to initialize H_{ij} and all the other variables as described above. The simulations are run over a time interval $t \in [0, 30]$. The PV evolution generated by the PVSPM method is shown in Fig. 7

6.2.1 PV-Particle Advection Method We computed the solution to Experiment B using the PV-particle advection (PPA) method described in §5 with $N = 32$ and 256^2 particles. The final PV field is shown in Fig. 8.

The relative errors in conserved quantities for the PPA method are compared with those of PS and PVSPM in Fig. 9. The relative errors of the PPA method are rather large in scale, but the conservation appears to be good. There is a small observable energy decay probably attributable to the Robert-Asselin filter.

Since PPA requires no differentiation of interpolated quantities, it does a good job of preserving balance, as illustrated by the evolution of the divergence field norm in Fig. 10. The mean divergence level does not show the systematic growth observed with the PVSPM method.

7. DIRECTIONS FOR FURTHER RESEARCH

7.1 Implementation of improved interpolation operators

The numerical experiments indicate that some smoothness in the interpolation operators \mathcal{A} and \mathcal{B} is necessary to avoid an undesirable drift in the divergence of the Eulerian velocity field and the artificial generation of short wave-length gravity waves. This can be achieved by cubic spline interpolation [3].

One could also think of better ways of defining the operator \mathcal{B} such that $\mathcal{B}\mathcal{A} = \text{id}_E$. For example, assuming that \mathcal{A} is defined by bilinear interpolation, one can define \mathcal{B} locally over each Eulerian grid cell by minimizing

$$S(a, b, c, d) = \sum_i (f_i - f(x_i, y_i))^2,$$

where $f(x, y) = a + bx + cy + dxy$ and (x_i, y_i) are the Lagrangian coordinates of particles within the cell. Once the parameters a, b, c, d are known, an interpolated grid value of f can be found. Since four cells are attached to each grid point, the average of the four interpolation values is taken.

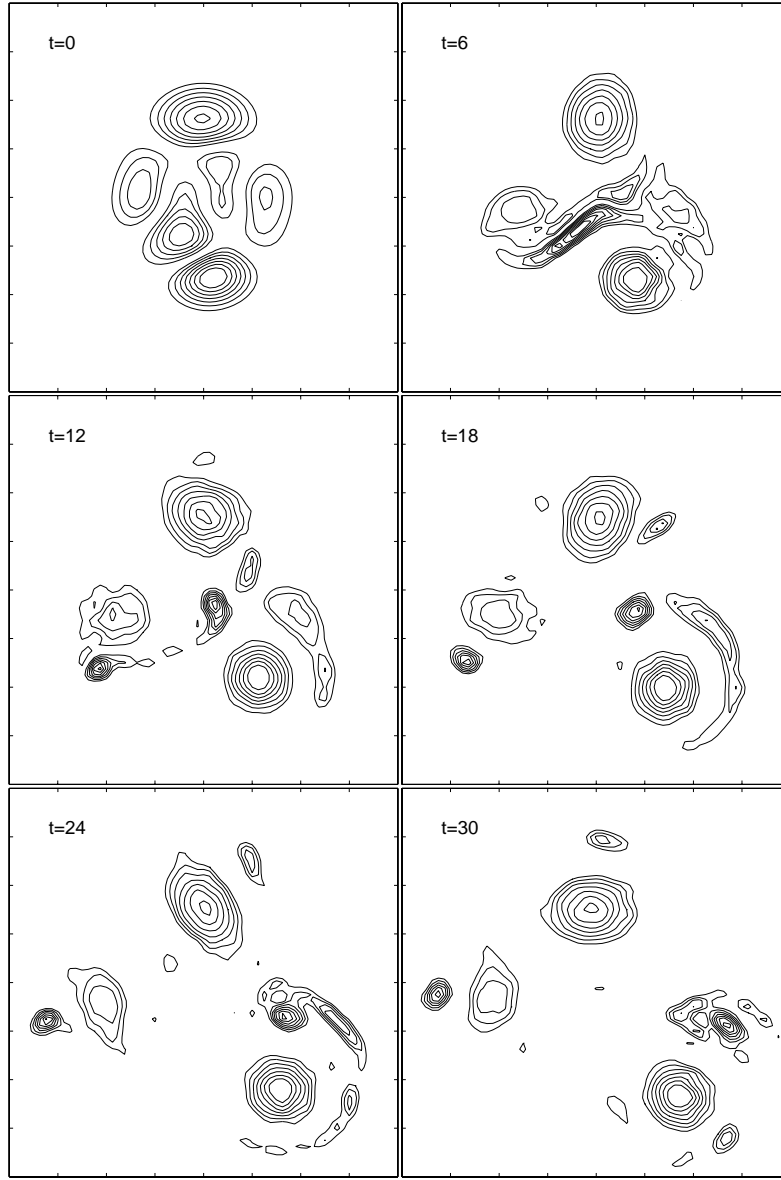


Figure 7: PV contours evolved by the PVSPM method, $N = 32$, $M = 256^2$.

7.2 Other implementations of the PV-conserving splitting

In this paper, we have focused on particle-mesh implementations of the PV-conserving splitting of the SWEs. But there are several other implementations that seem worthwhile to pursue. Let us briefly outline a few of these.

7.2.1 A modified LF/TR implementation We write the SWEs in the form

$$\begin{aligned} \mathbf{u}_t &= \mathbf{A}\mathbf{u} + \mathbf{c}(h) + \mathbf{f}(\mathbf{u}), \\ q_t &= \mathbf{u} \cdot \nabla_{\mathbf{x}} q, \end{aligned}$$

where

$$\mathbf{c}(h) = -c_0 \nabla_{\mathbf{x}} h, \quad \mathbf{A}\mathbf{u} = -f_0 \mathbf{e}_z \times \mathbf{u}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} -uu_x - vv_y \\ -uv_x - vv_y \end{pmatrix},$$

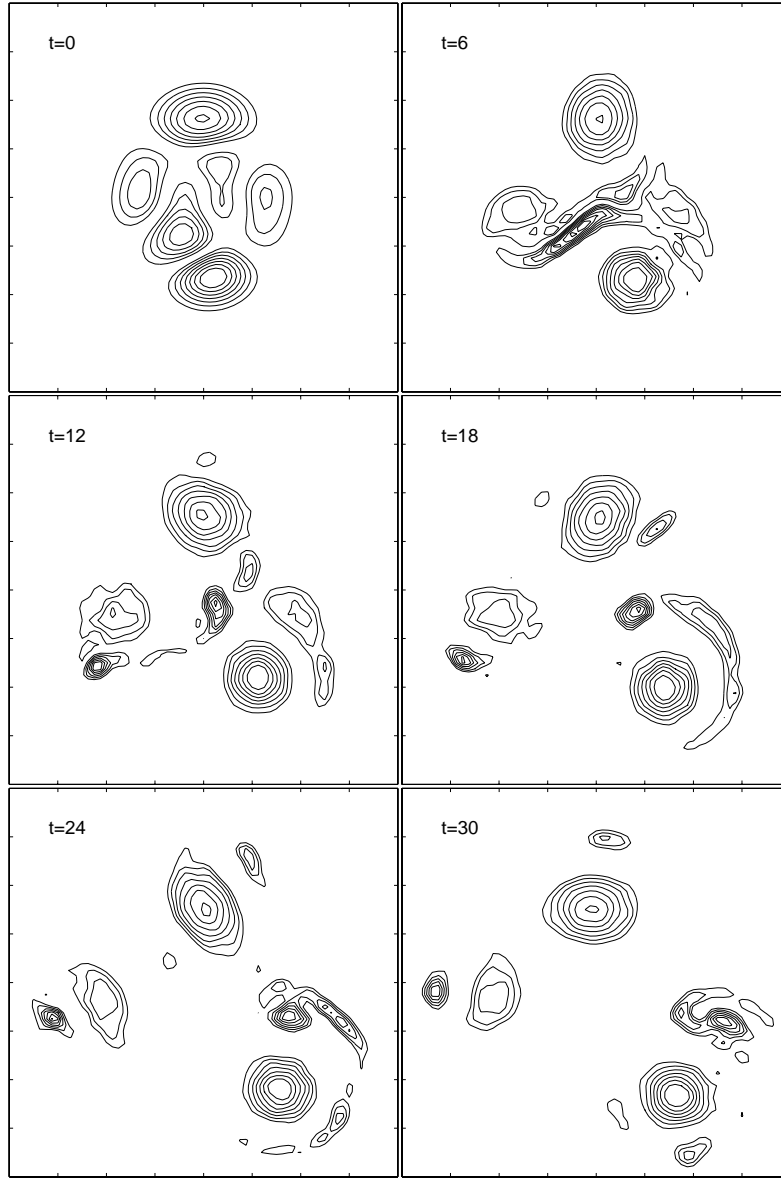


Figure 8: PV contours evolved by the PPA method, $N = 32$, $M = 256^2$.

and the layer-depth $h = h(\mathbf{u}, q)$ is defined as

$$h = \frac{\omega - f_0}{q}.$$

We first propose the following generalization of the time-reversible two-step LF/TR scheme (5.5) to solve the above formulation:

$$\mathbf{u}^{n-1/2} = \mathbf{u}^{n-1} + \Delta t \mathbf{c}(h^{n-1}), \quad (7.1)$$

$$(\mathbf{I} - \Delta t \mathbf{A}) \mathbf{u}^{n+1/2} = 2\Delta t \mathbf{f}(\mathbf{u}^n) + (\mathbf{I} + \Delta t \mathbf{A}) \mathbf{u}^{n-1/2}, \quad (7.2)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^{n+1/2} + \Delta t \mathbf{c}(h^{n+1}), \quad (7.3)$$

$$q^{n+1} = q^{n-1} - 2\Delta t \mathbf{u}^n \cdot \nabla_{\mathbf{x}} q^n. \quad (7.4)$$

If $\mathbf{c}(h) = \mathbf{0}$, then the standard LF/TR-rule is obtained.

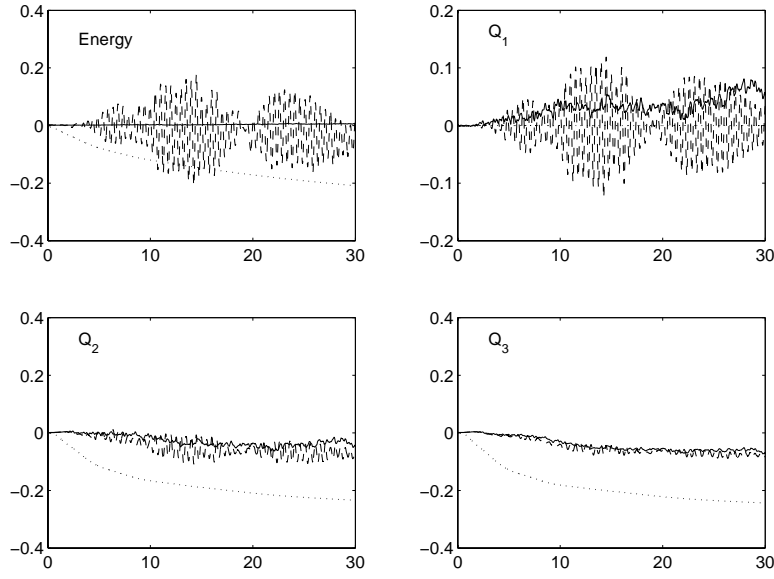


Figure 9: Relative errors in total energy and enstrophies for PVSPM (solid), PPA (dashed) and PS (dotted) methods.

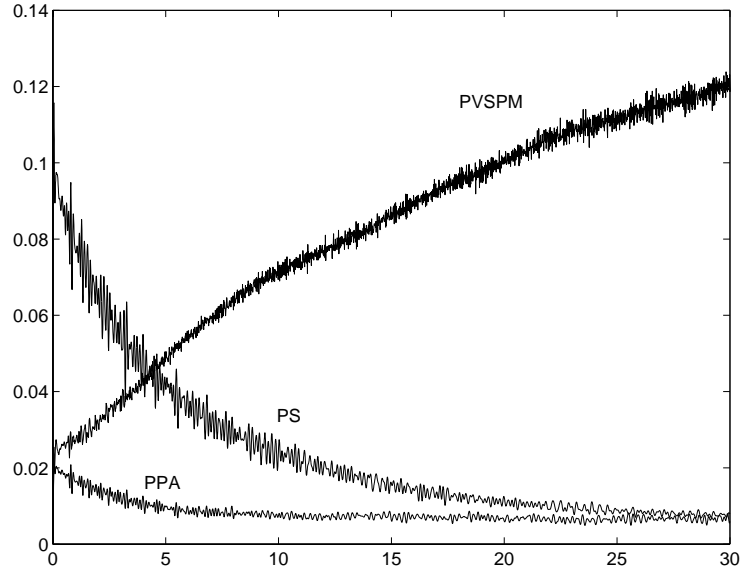


Figure 10: Time evolution of the L_2 norm of the divergence field produced by PVSPM, PS and PPA methods.

However, the above scheme is by no means optimal. First we note that the matrix operations in (7.2) can be replaced by exact exponentiation; i.e.

$$(\mathbf{I} - \Delta t \mathbf{A}) \approx e^{-\Delta t \mathbf{A}} \quad \text{and} \quad (\mathbf{I} + \Delta t \mathbf{A}) \approx e^{\Delta t \mathbf{A}}.$$

Furthermore, eqs. (7.1)–(7.3) can then be condensed into the single equation

$$\mathbf{u}^{n+1} = 2\Delta t e^{\Delta t \mathbf{A}} \mathbf{f}(\mathbf{u}^n) + e^{2\Delta t \mathbf{A}} (\mathbf{u}^{n-1} + \Delta t \mathbf{c}(h^{n-1})) + \Delta t \mathbf{c}(h^{n+1}). \quad (7.5)$$

Hence we have obtained a fully explicit method. But we can push things even further. While the velocities \mathbf{u} are defined on integer time-steps, we now suggest to keep PV q on half steps only.

Hence we modify (7.4) to the linearly implicit one-step formulation

$$q^{n+1/2} = q^{n-1/2} - 2\Delta t \mathbf{u}^n \cdot \nabla_{\mathbf{x}} \frac{q^{n+1/2} + q^{n-1/2}}{2}. \quad (7.6)$$

To make the overall method (7.5)–(7.6) time-reversible⁴, we have to modify (7.5) to

$$\mathbf{u}^{n+1} = 2\Delta t e^{\Delta t \mathbf{A}} \mathbf{f}(\mathbf{u}^n) + e^{2\Delta t \mathbf{A}} (\mathbf{u}^{n-1} + \Delta t \mathbf{c}(h_+^{n-1})) + \Delta t \mathbf{c}(h_-^{n+1}).$$

where

$$h_+^{n-1} := \frac{\omega^{n-1} + f_0}{q^{n-1/2}} \quad \text{and} \quad h_-^{n+1} := \frac{\omega^{n+1} + f_0}{q^{n+1/2}}.$$

Note that $h_+^n \neq h_-^n$, in general.

The PV-advection step (7.6) is linearly implicit. Upon introducing Lagrangian particles $\mathbf{x}_i = (x_i, y_i)$, $i = 1, \dots, M$, we can make that step explicit and define

$$\mathbf{x}_i^{n+1/2} = \mathbf{x}_i^{n-1/2} + \Delta t \mathbf{u}_i^n$$

since

$$\frac{d}{dt} q_i = \text{const.}$$

along particle paths. To close the scheme, we need interpolation operators from the Eulerian grid to the Lagrangian particles and vice-versa. Note that these two operators can be chosen independently since we *do not* have to map the velocities back and forth! Hence the mixed Eulerian-Lagrangian scheme is implicit only in ω . Note also that the staggered arrangement of the velocities and particle locations is similar to the Störmer-Verlet method in classical mechanics. The main difference is that the velocities are here propagated by a two-step method. The overall scheme is, however, still time-reversible. It can also be implemented as a fractional time-stepping method.

Let us also briefly mention the following larger time-step variant of the LF/TR rule applied to a system

$$\mathbf{z}_t = \mathbf{A}\mathbf{z} + \mathbf{f}(\mathbf{z}).$$

Upon assuming that $\mathbf{f}(\mathbf{z})$ is almost constant over one time-step, we can apply the variation of constant formula to obtain

$$\mathbf{z}(\Delta t) = e^{\Delta t \mathbf{A}} \mathbf{z}_0 + e^{\Delta t \mathbf{A}} \int_0^{\Delta t} e^{-\tau \mathbf{A}} \mathbf{f}(\mathbf{z}_0) d\tau,$$

and

$$\mathbf{z}(-\Delta t) = e^{-\Delta t \mathbf{A}} \mathbf{z}_0 + e^{-\Delta t \mathbf{A}} \int_0^{-\Delta t} e^{-\tau \mathbf{A}} \mathbf{f}(\mathbf{z}_0) d\tau.$$

Hence

$$e^{-\Delta t \mathbf{A}} \mathbf{z}(\Delta t) = e^{\Delta t \mathbf{A}} \mathbf{z}(-\Delta t) + \int_{-\Delta t}^{\Delta t} e^{-\tau \mathbf{A}} \mathbf{f}(\mathbf{z}_0) d\tau$$

which suggest the modified LF/TR scheme

$$\mathbf{z}^{n+1} = e^{2\Delta t \mathbf{A}} \mathbf{z}^{n-1} + e^{\Delta t \mathbf{A}} \int_{-\Delta t}^{\Delta t} e^{-\tau \mathbf{A}} \mathbf{f}(\mathbf{z}^n) d\tau.$$

⁴As formulated here, the method is time-reversible (symmetric) if it is symmetric about the time-step index n .

7.2.2 A modified splitting This is a variant of the PV-Split-Particle-Mesh (PVSPM) method. The novel part is that we decompose the layer-depth variation η into its geostrophic part

$$\eta_g = -\left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} q$$

and its unbalanced contribution $h_{\text{ag}} = h - \eta_g$, $h = H_0 + \eta$ the total layer-depth, and $L_R = \sqrt{c_0 H_0}/f_0$ the Rossby radius. Let us derive the governing differential equation for h_{ag} . We start with

$$\begin{aligned} h_t &= -\left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} q_t + \partial_t h_{\text{ag}} \\ &= -\left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \frac{d}{dt} q + \partial_t h_{\text{ag}} + \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) \\ &= \partial_t h_{\text{ag}} + \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q). \end{aligned}$$

On the other hand,

$$\begin{aligned} \nabla_{\mathbf{x}} \cdot (h\mathbf{u}) &= \nabla_{\mathbf{x}} \cdot (h_{\text{ag}}\mathbf{u}) + \eta_g \nabla_{\mathbf{x}} \cdot \mathbf{u} + \mathbf{u} \cdot \nabla_{\mathbf{x}} \eta_g \\ &= \nabla_{\mathbf{x}} \cdot (h_{\text{ag}}\mathbf{u}) + \eta_g \nabla_{\mathbf{x}} \cdot \mathbf{u} - \mathbf{u} \cdot \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \nabla_{\mathbf{x}} q. \end{aligned}$$

Putting everything together, we obtain

$$\partial_t h_{\text{ag}} = -\nabla_{\mathbf{x}} \cdot (h_{\text{ag}}\mathbf{u}) - \eta_g \nabla_{\mathbf{x}} \cdot \mathbf{u} - \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) + \mathbf{u} \cdot \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \nabla_{\mathbf{x}} q$$

or

$$\frac{d}{dt} h_{\text{ag}} = -h \nabla_{\mathbf{x}} \cdot \mathbf{u} - \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) + \mathbf{u} \cdot \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \nabla_{\mathbf{x}} q.$$

The shallow-water equations are now rewritten as

$$\begin{aligned} \frac{d}{dt} \mathbf{u} &= -f_0 \mathbf{e}_z \times \mathbf{u} - c_0 \nabla_{\mathbf{x}} \eta_g - c_0 \nabla_{\mathbf{x}} h_{\text{ag}}, & (7.7) \\ \frac{d}{dt} h_{\text{ag}} &= -(\eta_g + h_{\text{ag}}) \nabla_{\mathbf{x}} \cdot \mathbf{u} - \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) + \mathbf{u} \cdot \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \nabla_{\mathbf{x}} q & (7.8) \\ \frac{d}{dt} q &= 0. & (7.9) \end{aligned}$$

We split the equations into the following three parts:

STEP 1.

$$\mathbf{u}_t = -c_0 \nabla_{\mathbf{x}} (\eta_g + h_{\text{ag}}), \quad q_t = 0, \quad \partial_t h_{\text{ag}} = 0.$$

STEP 2.

$$\mathbf{u}_t = -f_0 \mathbf{e}_z \times \mathbf{u}, \quad q_t = 0, \quad \partial_t h_{\text{ag}} = -\frac{f_0}{q} \nabla_{\mathbf{x}} \cdot \mathbf{u}.$$

STEP 3.

$$\frac{d}{dt} \mathbf{u} = \mathbf{0}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{u}, \quad \frac{d}{dt} q = 0,$$

and

$$\frac{d}{dt} h_{\text{ag}} = -\left(h - \frac{f_0}{q}\right) \nabla_{\mathbf{x}} \cdot \mathbf{u} - \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) + \mathbf{u} \cdot \left(1 - L_R^2 \nabla_{\mathbf{x}}^2\right)^{-1} \nabla_{\mathbf{x}} q.$$

The last of these equations can be simplified to

$$\frac{d}{dt}h_{\text{ag}} = - (1 - L_R^2 \nabla_{\mathbf{x}}^2)^{-1} (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) + \mathbf{u} \cdot (1 - L_R^2 \nabla_{\mathbf{x}}^2)^{-1} \nabla_{\mathbf{x}} q,$$

since

$$\left(h - \frac{f_0}{q}\right) \nabla_{\mathbf{x}} \cdot \mathbf{u} \approx 0$$

for the flows considered here. Note that

$$\mathcal{S} = (1 - L_R^2 \nabla_{\mathbf{x}}^2)^{-1}$$

is a nice smoothing operator with smoothing length proportional to the Rossby radius L_R .

STEP 1 & 2 are purely Eulerian steps while STEP 3 involves advection of Lagrangian particles. But all steps can be implemented as explicit integration steps.

STEP 2 can be rewritten as

$$\delta_t = -f_0 \omega, \quad \omega_t = f_0 \delta, \quad q_t = 0, \quad \partial_t h_{\text{ag}} = -\frac{f_0}{q} \delta.$$

If non-smoothness in the divergence δ is causing troubles in the numerical simulation, one could “regularize” STEP 2 by replacing δ by $\mathcal{S}_\alpha \delta$, i.e.,

$$\delta_t = -f_0 \omega, \quad \omega_t = f_0 \mathcal{S}_\alpha \delta, \quad q_t = 0, \quad \partial_t h_{\text{ag}} = -\frac{f_0}{q} \mathcal{S}_\alpha \delta,$$

where

$$\mathcal{S}_\alpha = (1 - (\alpha L_R)^2 \nabla_{\mathbf{x}}^2)^{-1}, \quad 0 < \alpha \ll 1.$$

Note that this does *not* affect the geostrophic velocity which is obtained by setting $\delta_t = 0$ in the combined STEPS 1 & 2 nor conservation of PV!

One could also apply the modified LF/TR implementation of §7.2.1. We write the shallow-water equations (7.7)-(7.9) as

$$\begin{aligned} \mathbf{w}_t &= \mathbf{A} \mathbf{w} + \mathbf{c}(\eta_g) + \mathbf{f}(\mathbf{w}), \\ q_t &= \mathbf{u} \cdot \nabla_{\mathbf{x}} q, \end{aligned}$$

$\mathbf{w} = (u, v, h_{\text{ag}})^T$. The linear operator \mathbf{A} is the same as in (5.7) and $\mathbf{c}(\eta_g) = -(c_0 \partial_x \eta_g, c_0 \partial_y \eta_g, 0)^T$. Hence we obtain the scheme

$$\mathbf{w}^{n+1} = 2\Delta t e^{\Delta t \mathbf{A}} \mathbf{f}(\mathbf{w}^n) + e^{2\Delta t \mathbf{A}} \left(\mathbf{w}^{n-1} + \Delta t \mathbf{c}(\eta_g^{n+1/2}) \right) + \Delta t \mathbf{c}(\eta_g^{n-1/2}),$$

$\eta_g^{n\pm 1/2} = (1 - L_R^2 \nabla_{\mathbf{x}}^2)^{-1} q^{n\pm 1/2}$, and particle advection for the PV field q .

Following the discussion in the previous subsection, one could also implement the following “large time-step” variant of the above scheme:

$$\mathbf{w}^{n+1} = e^{2\Delta t \mathbf{A}} \mathbf{w}^{n-1} + e^{\Delta t \mathbf{A}} \int_{-\Delta t}^{\Delta t} e^{-\tau \mathbf{A}} [\mathbf{f}(\mathbf{w}^n) + \mathbf{c}(\eta_g^n)] d\tau,$$

where

$$\eta_g^n = (1 - L_R^2 \nabla_{\mathbf{x}}^2)^{-1} \frac{q^{n+1/2} + q^{n-1/2}}{2}.$$

This might require that the particles are also advected along a mean-flow velocity.

7.3 Radial basis function approach to the continuity equation

To avoid the limitations placed by the assumption (1.6) on our method, we propose a method based on radial basis functions.

Assume that a set of Lagrangian particles $\{\mathbf{x}_k(t)\}$ is given as a function of time and that

$$\frac{d}{dt}\mathbf{x}_k = \mathbf{u}_k.$$

Then the time evolution of a quantity f , satisfying a continuity equation

$$\frac{d}{dt}f = -f\nabla_{\mathbf{x}} \cdot \mathbf{u},$$

can be approximated by

$$f(\mathbf{x}, t) = \sum_{k=1}^M \gamma_k \psi(\|\mathbf{x} - \mathbf{x}_k(t)\|^2).$$

Here $\{\gamma_k\}$ are constants determined by the initial $f(\mathbf{x})$ field and $\psi(r)$ is an appropriate radial basis function such as, for example,

$$\psi(r^2) = \left(\left(\frac{r}{R} \right)^2 + c^2 \right)^{-1/2},$$

$c, R > 0$ two parameters. This idea is exploited in smoothed particle hydrodynamics (SPH) and related methods.

In particular, introduce the Hamiltonian (energy)

$$\mathcal{H} = \sum_{k=1}^M \frac{1}{2} \|\mathbf{u}_k\|^2 + \sum_{l,k=1}^M \alpha_k \psi(\|\mathbf{x}_l - \mathbf{x}_k\|^2),$$

where the layer-depth h at $\mathbf{x} = \mathbf{x}_l$ is approximated by

$$h_l(t) = \sum_{k=1}^M \alpha_k \psi(\|\mathbf{x}_l(t) - \mathbf{x}_k(t)\|^2),$$

$\{\alpha_k\}$ a fixed set of parameters.

Then the discretized SWE equations are

$$\frac{d}{dt}\mathbf{x}_k = \nabla_{\mathbf{u}_k} \mathcal{H}, \tag{7.10}$$

$$\frac{d}{dt}\mathbf{u}_k = -f_0 \mathbf{e}_z \times \mathbf{u}_k - \nabla_{\mathbf{x}_k} \mathcal{H}. \tag{7.11}$$

These equations are canonical but do not conserve PV.

Let us introduce fixed Eulerian grid points $\{\mathbf{X}_{ij}\}$ and use radial basis functions to approximate the PV field over this Eulerian grid given the PV values at the Lagrangian points $\{\mathbf{x}_k\}$. Hence, we introduce a second set of fixed parameters $\{\beta_k\}$ and define

$$Q_{ij}(t) = \frac{\sum_{k=1}^M \beta_k \psi(\|\mathbf{X}_{ij} - \mathbf{x}_k(t)\|^2)}{\sum_{k=1}^M \psi(\|\mathbf{X}_{ij} - \mathbf{x}_k(t)\|^2)}. \tag{7.12}$$

The values of $\{\beta_k\}$ are initially determined by the value of the PV field at $\{\mathbf{x}_k\}$. Since these values are constant, we can assume that the parameters $\{\beta_k\}$ are constant as well.

Next note that

$$\partial_t Q_{ij} = -(\mathbf{u} \cdot \nabla_{\mathbf{x}} q)_{ij} = \sum_{k=1}^M \nabla_{\mathbf{x}_k} \Psi_{ij}(\{\mathbf{x}_l\}) \cdot \mathbf{u}_k,$$

where

$$\Psi_{ij}(\{\mathbf{x}_k\}) = \frac{\sum_{k=1}^M \beta_k \psi(\|\mathbf{X}_{ij} - \mathbf{x}_k\|^2)}{\sum_{k=1}^M \psi(\|\mathbf{X}_{ij} - \mathbf{x}_k\|^2)}.$$

Let us introduce the vector \mathbf{Q} of all $\{Q_{ij}\}$ and the vector \mathbf{u} of all $\{\mathbf{u}_k\}$. Hence it holds that

$$\langle \mathbf{u}, \mathbf{A}\mathbf{Q} \rangle \approx - \int (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) dxdy,$$

where \mathbf{A} is a matrix with entries $Q_{ij}^{-1} \nabla_{\mathbf{x}_k} \Psi_{ij}(\{\mathbf{x}_l\})$. Assuming periodic boundary conditions, we obtain

$$\int (q \nabla_{\mathbf{x}} \cdot \mathbf{u}) dxdy = - \int (\mathbf{u} \cdot \nabla_{\mathbf{x}} q) dxdy \approx \langle \mathbf{Q}, \mathbf{A}^T \mathbf{u} \rangle$$

which implies the approximations

$$(\nabla_{\mathbf{x}} \cdot \mathbf{u})_{ij} \approx Q_{ij}^{-1} \sum_{k=1}^M \nabla_{\mathbf{x}_k} \Psi_{ij}(\{\mathbf{x}_l\}) \cdot \mathbf{u}_k$$

and

$$(\nabla_{\mathbf{x}} \times \mathbf{u})_{ij} \approx Q_{ij}^{-1} \sum_{k=1}^M \nabla_{\mathbf{x}_k} \Psi_{ij}(\{\mathbf{x}_l\}) \times \mathbf{u}_k$$

We can now formulate conservation of PV over the Eulerian grid as

$$Q_{ij}(Q_{ij}H_{ij} - f_0) = \sum_{k=1}^M \nabla_{\mathbf{x}_k} \Psi_{ij}(\{\mathbf{x}_l\}) \times \mathbf{u}_k, \quad (7.13)$$

where

$$H_{ij} = \sum_{k=1}^M \alpha_k \psi(\|\mathbf{X}_{ij} - \mathbf{x}_k\|^2)$$

is the layer-depth over the Eulerian grid points.

The non-holonomic constraint (7.13) can be imposed on (7.10)–(7.11) via

$$\frac{d}{dt} \mathbf{x}_k = \nabla_{\mathbf{u}_k} \mathcal{H}, \quad (7.14)$$

$$\frac{d}{dt} \mathbf{u}_k = -f_0 \mathbf{e}_z \times \mathbf{u}_k - \nabla_{\mathbf{x}_k} \mathcal{H} - \sum_{i,j} \Lambda_{ij} \mathbf{e}_z \times \nabla_{\mathbf{x}_k} \Psi_{ij}, \quad (7.15)$$

where the $\{\Lambda_{ij}\}$'s are appropriate Lagrangian multipliers.

These equations are no longer canonical but approximate conservation of PV is now built in. We discretize the equations using a variant of SHAKE:

$$\begin{aligned} \mathbf{x}_k^{n+1/2} &= \mathbf{x}_k^n + \frac{\Delta t}{2} \nabla_{\mathbf{u}_k} \mathcal{H}^n, \\ \mathbf{u}_k^{n+1} &= \mathbf{u}_k^n - \Delta t f_0 \mathbf{e}_z \times \frac{\mathbf{u}_k^{n+1} + \mathbf{u}_k^n}{2} - \Delta t \nabla_{\mathbf{x}_k} \mathcal{H}^{n+1/2} - \Delta t \sum_{i,j} \Lambda_{ij}^{n+1/2} \mathbf{e}_z \times \nabla_{\mathbf{x}_k} \Psi_{ij}^{n+1/2}, \\ \mathbf{x}_k^{n+1} &= \mathbf{x}_k^{n+1/2} + \frac{\Delta t}{2} \nabla_{\mathbf{u}_k} \mathcal{H}^{n+1}, \\ 0 &= Q_{ij}^{n+1} (Q_{ij}^{n+1} H_{ij}^{n+1} - f_0) - \sum_{k=1}^M \nabla_{\mathbf{x}_k} \Psi_{ij}^{n+1} \times \mathbf{u}_k^{n+1}, \end{aligned}$$

$k = 1, \dots, 0$.

7.4 Application to more general GFD models

7.4.1 β -plane approximation of SWEs Often the assumption of a constant angular velocity is not appropriate and one replaces the constant f_0 in the SWEs by the β -plane approximation

$$f = f_0 + \beta y.$$

The associated PV-conserving geostrophic splitting of §2 is given by

$$\mathbf{u}_t = -f_0 \mathbf{e}_z \times \mathbf{u} - c_0 \nabla_{\mathbf{x}} \eta, \quad \eta_t = -H_g \nabla_{\mathbf{x}} \cdot \mathbf{u},$$

where $H_g = f_0/q$ as before, and

$$\frac{d}{dt} \mathbf{u} = -\beta y \mathbf{e}_z \times \mathbf{u}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{u}.$$

The Lagrangian step is no longer fully explicit but can be integrated numerically by a simple splitting into

$$\frac{d}{dt} \mathbf{u} = -\beta y \mathbf{e}_z \times \mathbf{u}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{0},$$

and

$$\frac{d}{dt} \mathbf{u} = \mathbf{0}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{u}.$$

7.4.2 The primitive equations in isentropic coordinates The SWEs are often too crude of an approximation and one has to resort to the primitive equations [13]. Here we outline how to apply our splitting to those equations. We take coordinates $\mathbf{x} = (x, y)^T \in \mathbb{R}^2$ and potential temperature θ as independent variables. The dependent variables are velocity $\mathbf{u} = (u, v)^T \in \mathbb{R}^2$, pressure B and layer-depth h . The equations of motion are

$$\begin{aligned} \frac{d}{dt} \mathbf{u} &= -f_0 \mathbf{e}_z \times \mathbf{u} - \nabla_{\mathbf{x}} B, \\ \frac{d}{dt} h &= -h \nabla_{\mathbf{x}} \cdot \mathbf{u}, \\ 0 &= h + B_{\theta\theta} \end{aligned}$$

where $dg/dt = g_t + g_x u + g_y v$ is the material derivative as before. Potential vorticity is defined by

$$q = \frac{v_x - u_y + f_0}{h}$$

and satisfies $dq/dt = 0$. Following §2, we introduce the geostrophic layer-depth $H_g = f_0/q$ and consider the system

$$\begin{aligned} \mathbf{u}_t &= -f_0 \mathbf{e}_z \times \mathbf{u} - \nabla_{\mathbf{x}} B, \\ h_t &= -H_g \nabla_{\mathbf{x}} \cdot \mathbf{u}, \\ 0 &= h + B_{\theta\theta} \end{aligned}$$

as the equivalent to (2.2)–(2.3). It is easily checked that $q_t = 0$ along solutions of these linear equations. The second part of the splitting is given, as before, by the Lagrangian advection equations

$$\frac{d}{dt} \mathbf{u} = \mathbf{0}, \quad \frac{d}{dt} \mathbf{x} = \mathbf{u}.$$

SPH can be implemented by allowing the coefficients $\{\rho_k\}$ and the velocities $\{\mathbf{u}_k\}$ to depend on the potential temperature θ and by introducing an additional set of variables (functions) $\{B_k(\theta)\}$. Note that we now also have to solve a 1D Poisson equation in B .

7.4.3 Spherical geometry The method described in §7.3, which does not assume the near balance condition (1.6), is suitable for adaptation to spherical geometry. The Eulerian grid functions should be expanded in spherical harmonics to avoid difficulties at the pole, and the Lagrangian advection can be handled by standard methods for constrained dynamics.

8. CONCLUSIONS

Standard pseudospectral methods are unsuitable for long time simulations of geophysical flows, due to the artificial measures required to keep them stable. In this paper we have presented preliminary results for a the PVSPM method, based on an operator splitting of the shallow water equations into a linear, Eulerian gravity wave problem for which the PV is constant and a Lagrangian advection problem for which PV is exactly conserved along particle trajectories. These two subproblems are coupled via interpolation, and we have tested the idea using both bilinear and Fourier interpolation methods. The results are promising with respect to conservation of energy and PV-momenta, as well as to the quality of the solution obtained. The numerical experiments indicate that a better interpolation operator than simple bilinear interpolation is needed. Furthermore, the global nature of the Fourier interpolator makes it far too expensive for practical use.

Additional experiments with the PPA method, similar to the contour advective semi-Lagrangian method of Dritschel et al. [4], but without the extra computational baggage of contour advection, are also quite promising.

The results of the numerical experiments suggest a number of directions for further research:

- Possibilities for getting a smoother interpolation between particles and mesh are higher order spline interpolations (§7.1) and radial basis functions (§7.3), which yield a method related to smoothed particle hydrodynamics. This approach has the additional advantage that the assumption of nearly geostrophic conditions can be dropped.
- A modified splitting (§7.2.2) allows smoothing of the interpolated divergence field and direct interpolation of the layer depth, rather than computation as the ratio of interpolated (and differentiated) quantities.
- A modified LF/TR-type method for PPS-type methods is also suggested, based on formulation of the SWEs in velocity-PV variables.

Extensions to more sophisticated geophysical models have also been discussed.

REFERENCES

1. G. BROWNING, A. KASAHARA & H.O. KREISS, *Initialization of the primitive equations by the bounded time derivative method*, J. Atmospheric Sci. **37**, 1980, 1424–1436.
2. O. BÜHLER & M.E. MCINTYRE, *On non-dissipative wave-mean interactions in the atmosphere or ocean*, J. Fluid Mech. **354**, 1998, 301–343.
3. C. DE BOOR, *A practical guide to splines*, Springer-Verlag, 1978.
4. D.G. DRITSCHER, L.M. POLVANI, & A.R. MOHEBALHOJEH, *The contour-advective semi-Lagrangian algorithm for the shallow water equations*, Mon. Weather Rev. **127**, 1999, 1551–1565.
5. D.R. DURRAN, *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*, Springer-Verlag, New York, 1999.
6. F.H. HARLOW, *The particle-in-cell computing methods for fluid dynamics*, Methods Comput. Phys. **3**, 1964, 319–343.
7. R. W. HOCKNEY & J. W. EASTWOOD, *Computer Simulation Using Particles*, Adam Hilger, Bristol, New York, 1988.
8. A. ISERLES, *A First Course in the Numerical Analysis of Differential Equations*, Cambridge University Press, 1996.

9. D.A. JONES, A. MAHALOV & B. NICOLAENKO, *A numerical study of an operator splitting method for rotating flows with large ageostrophic initial data*. Theoret. Comput. Fluid Dynamics **13**, 1999, 143–159.
10. A.J. MAJDA & P. EMBID, *Averaging over fast gravity waves for geophysical flows with unbalanced initial data*, Theoret. Comput. Fluid Dynamics **11**, 1997, 155–169.
11. A.R. MOHEBALHOJEH & D.G. DRITSCHER, *On the representation of gravity waves in numerical models of the shallow-water equations*, Q. J. R. Meteorol. Soc. **126**, 2000, 669–688.
12. J. J. MONAGHAN, *Smoothed particle hydrodynamics*, Ann. Rev. Astron. Astrophys. **30** (1992), 543–574.
13. R. SALMON, *Lectures on Geophysical Fluid Dynamics*, Oxford University Press, Oxford, 1999.