A Lagrangian Particle Method with Remeshing for Tracer Transport on the Sphere

Peter A. Bosler[a,*], James Kent[b], Robert Krasny[c], Christiane Jablonowski[d]

[a] *Center for Computing Research, Sandia National Laboratories, Albuquerque NM 87123-1321, USA*
[b] *Mathematics and Statistics, University of South Wales, CF37 1DL, Pontypridd, UK*
[c] *Department of Mathematics, University of Michigan, Ann Arbor, MI 48109-1043, USA*
[d] *Department of Climate and Space Sciences and Engineering, University of Michigan, Ann Arbor, MI 48109-2143, USA*

Abstract

A Lagrangian particle method (called LPM) based on the flow map is presented for tracer transport on the sphere. The particles carry tracer values and are located at the centers and vertices of triangular Lagrangian panels. Remeshing is applied to control particle disorder and two schemes are compared, one using direct tracer interpolation and another using inverse flow map interpolation with sampling of the initial tracer density. Test cases include a moving-vortices flow and reversing-deformational flow with both zero and nonzero divergence, as well as smooth and discontinuous tracers. We examine the accuracy of the computed tracer density and tracer integral, and preservation of nonlinear correlation in a pair of tracers. We compare results obtained using LPM and the Lin-Rood finite-volume scheme. An adaptive particle/panel refinement scheme is demonstrated.

Keywords: tracer transport, Lagrangian particle method, flow map, remeshing, adaptive refinement

1. Introduction

Tracer transport plays an important role in many multiphysics applications. In atmospheric dynamics, for example, the tracer may represent a chemical species or distribution of solid particles and liquid droplets [36]. A key goal in future climate simulations is to track the large number of tracer species that appear in advanced atmospheric models [33]. In order for these efforts to succeed it is clear that the numerical method used in computing tracer transport must be sufficiently accurate and efficient [52].

With these considerations in mind we consider the *tracer transport problem* on the sphere. An initial tracer mixing ratio, or concentration, $q_0(\boldsymbol{x})$ and fluid velocity $\boldsymbol{u}(\boldsymbol{x}, t)$ are given, where $\boldsymbol{x} \in S$ denotes points on the sphere S and t is time. The tracer is advected in the given velocity field and the goal is to compute the tracer mixing ratio $q(\boldsymbol{x}, t)$ for $t > 0$. In practice the velocity field is computed either by a general circulation model or acquired by data assimilation, but here it is known analytically. This idealized problem is often used to test

*Corresponding author
Email address: pabosle@sandia.gov (Peter A. Bosler)

the capability of numerical methods for tracer transport [37, 38] and we present a Lagrangian particle method (called LPM) for this purpose. Our approach is motivated by previous work on vortex methods for incompressible flow [13, 16]. First we review the standard Eulerian formulation of the tracer transport problem and then the Lagrangian formulation on which LPM is based.

1.1. Eulerian formulation

In the Eulerian formulation the tracer density satisfies the *transport equation*,

$$\frac{\partial(\rho q)}{\partial t} + \nabla \cdot \big((\rho q)\boldsymbol{u}\big) = 0, \tag{1}$$

where $\rho(\boldsymbol{x}, t)$ is the fluid mass density, which satisfies the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0. \tag{2}$$

Both the mass density $\rho(\boldsymbol{x}, t)$ and the tracer mixing ratio $q(\boldsymbol{x}, t)$ are subject to initial conditions

$$\rho(\boldsymbol{x}, 0) = \rho_0(\boldsymbol{x}), \quad q(\boldsymbol{x}, 0) = q_0(\boldsymbol{x}). \tag{3}$$

In this work $\rho_0(\boldsymbol{x}) = 1$ and $q_0(\boldsymbol{x})$ is defined by a set of common test cases for transport on the sphere [37, 44].

Equations (1) and (2) are written in *flux form*; we may combine them to derive their *advective forms*,

$$\frac{Dq}{Dt} = \frac{\partial q}{\partial t} + \boldsymbol{u} \cdot \nabla q = 0, \tag{4}$$

and

$$\frac{D\rho}{Dt} = \frac{\partial \rho}{\partial t} + \boldsymbol{u} \cdot \nabla \rho = -(\nabla \cdot \boldsymbol{u})\rho, \tag{5}$$

where $D/Dt = \partial/\partial t + \boldsymbol{u} \cdot \nabla$ is the material derivative. We note that the advective form of the continuity equation (5) is the same as the advective form of the tracer transport equation (4) in non-divergent flow satisfying $\nabla \cdot \boldsymbol{u} = 0$. In the test cases presented below we show results for both this special case and for the more general case of flow with non-zero velocity divergence.

For later reference we note that the tracer satisfies the following two properties. The first property is that the tracer integral over the sphere is conserved in time,

$$Q(t) = Q(0), \quad t \geq 0, \tag{6}$$

where the tracer integral is defined by

$$Q(t) = \int_S \rho(\boldsymbol{x}, t) q(\boldsymbol{x}, t)\, dS(\boldsymbol{x}). \tag{7}$$

The second property is the preservation of correlations in the sense that if $q_1(\boldsymbol{x}, t)$ and $q_2(\boldsymbol{x}, t)$ are two tracers whose initial values are correlated,

$$q_{20}(\boldsymbol{x}) = F(q_{10}(\boldsymbol{x})), \tag{8}$$

where F is a given function, then the tracers remain correlated at later times,

$$q_2(\boldsymbol{x}, t) = F(q_1(\boldsymbol{x}, t)), \quad t > 0. \tag{9}$$

These two properties (conservation of tracer integral, preservation of correlations) are often used as benchmarks in evaluating tracer transport schemes and we also use them for this purpose.

A common approach to computing tracer transport is to discretize the advection equation in form (1) on an Eulerian grid (see for example [6, 35]). Many techniques have been employed including finite-difference [55], finite-volume [41, 51], discontinuous Galerkin [45], and spectral element [57] methods. Another approach discretizes the advection equation using radial basis functions on a set of quasi-uniformly distributed nodes [21, 22, 23]. In some cases filters and limiters are applied to control numerical errors arising from the discretization of the spatial gradient operator. However this can lead to excessive smoothing of sharp fronts [27, 28, 37], and it is still challenging to design Eulerian schemes that satisfy the competing requirements of accuracy and efficiency [32]. This has motivated the investigation of alternative Lagrangian schemes.

1.2. Lagrangian formulation

The Lagrangian formulation is based on the *flow map*, $\boldsymbol{\alpha} \to \boldsymbol{x}(\boldsymbol{\alpha}, t)$, defined by the equation

$$\frac{\partial}{\partial t}\boldsymbol{x}(\boldsymbol{\alpha}, t) = \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{\alpha}, t), t), \tag{10}$$

with initial condition

$$\boldsymbol{x}(\boldsymbol{\alpha}, 0) = \boldsymbol{\alpha}, \tag{11}$$

where $\boldsymbol{\alpha} \in S$ is a Lagrangian parameter on the sphere [13, 16]. The flow map $\boldsymbol{x}(\boldsymbol{\alpha}, t)$ gives the location $\boldsymbol{x}$ at time t of a fluid particle labeled by its initial location $\boldsymbol{\alpha}$. In the formulation (10), the time derivative $\partial/\partial t$ acts along particle trajectories and is equivalent to the material derivative operator. As a consequence, the partial differential equations (4) and (5) become ordinary differential equations (ODEs) along particle trajectories. It follows that the tracer is conserved along these trajectories,

$$q(\boldsymbol{x}(\boldsymbol{\alpha}, t), t) = q_0(\boldsymbol{\alpha}), \tag{12}$$

and this can be viewed as an integrated form of the advection equation. Similarly, the fluid density ρ satisfies the ODE,

$$\frac{\partial \rho}{\partial t}(\boldsymbol{x}(\boldsymbol{\alpha}, t), t) = -(\nabla \cdot \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{\alpha}, t), t))\rho(\boldsymbol{x}(\boldsymbol{\alpha}, t), t), \tag{13}$$

along the particle paths defined by (10).

After changing variables by the flow map, the tracer integral (7) can be expressed as

$$Q(t) = \int_S \rho(\boldsymbol{x}(\boldsymbol{\alpha}, t), t) q(\boldsymbol{x}(\boldsymbol{\alpha}, t)) J(\boldsymbol{\alpha}, t) dS(\boldsymbol{\alpha}), \tag{14}$$

where $J(\boldsymbol{\alpha}, t) = \det(\partial \boldsymbol{x}/\partial \boldsymbol{\alpha})$ is the Jacobian determinant of the flow map. In this approach one solves for the flow map using (10), the tracer mixing ratio is determined by (12), and the fluid density is determined by (13). Hence in this formulation there is no need to discretize the spatial gradient operator $\boldsymbol{u} \cdot \nabla$ which is an advantage of Lagrangian schemes. However, Lagrangian schemes face other challenges as we shall see.

1.3. Lagrangian schemes

A variety of Lagrangian schemes have been developed for geophysical flow simulations [40]. One class of semi-Lagrangian methods computes particle trajectories that arrive at grid points at each time step [7, 34, 56, 63]. Related approaches include particle-mesh methods [15] and the Arbitrary Lagrangian-Eulerian (ALE) method [25], as well as more recent hybrid schemes [26, 62]. There are also contour dynamics methods that track the boundary of vortex patches [18, 59] and schemes that track more general Lagrangian parcels [17, 24]. Here we propose a method for tracer transport on the sphere based on the particle-panel method [12, 20] in which the flow map is represented by moving particles and triangular Lagrangian panels.

Lagrangian schemes in general remain stable for longer time steps and are less prone to numerical diffusion than Eulerian grid-based schemes. However one difficulty for Lagrangian schemes is that even if the particles or parcels start from a regular grid, they typically become disordered as the flow evolves and this leads to a loss of accuracy [49]. In the context of ALE simulations, rezoning and remapping techniques are employed to overcome the problem of Lagrangian mesh distortion [10, 19]. Here we follow an alternative approach common in vortex methods research by remeshing the particles at regular time intervals (see for example [2, 4, 7, 30, 43, 48, 60]). In a remeshing step, a new set of well-ordered particles is introduced and the tracer values are interpolated from the old particles to the new ones; we refer to this as *direct remeshing*. A recent alternative scheme interpolates the inverse flow map and then samples the initial tracer density [11, 12]; we refer to this as *indirect remeshing*. In this work we will compare these two techniques for particle remeshing.

Another challenge for tracer transport schemes is the need to resolve small-scale features in the tracer density. For Eulerian schemes this is often addressed by adaptive mesh refinement (AMR), so that resolution is maintained without incurring the cost of a uniform fine mesh [9, 31]. In the context of radial basis function methods, locally refined node distributions have been investigated [22]. Adaptive refinement techniques have also been developed for Lagrangian particle methods [5, 8], and here we demonstrate an adaptive refinement scheme for LPM along the lines of previous work in vortex methods [12, 20], which is used to maintain accuracy and reduce the cost of long time integrations.

We will apply LPM to test cases for which the exact solution is known including a moving-vortices flow and a reversing-deformational flow with smooth and discontinuous tracers [29, 37, 44, 61]. We examine the accuracy of the computed tracer density and tracer integral, and preservation of nonlinear correlation in a pair of tracers. We also compare results using LPM and the Lin-Rood finite-volume scheme [41].

The article is organized as follows. Section 2 presents the LPM scheme for tracer transport on the sphere. Numerical results are presented in section 3 for the moving-vortices flow and in section 4 for the reversing-deformational flow. Adaptive particle/panel refinement is discussed in section 5. Section 6 provides a summary and conclusions.

2. LPM scheme for tracer transport on the sphere

In this section we explain the discrete representation of the flow map, followed by particle advection, remeshing, the error norms, and finally additional information about the scheme.

2.1. Discrete representation of the flow map

In LPM the flow map is represented by moving particles $\{\boldsymbol{x}_j(t)\}$ and their Lagrangian parameters $\{\boldsymbol{\alpha}_j\}$, satisfying

$$\boldsymbol{x}_j(0) = \boldsymbol{\alpha}_j, \quad \boldsymbol{x}_j(t) = \boldsymbol{x}(\boldsymbol{\alpha}_j, t), \quad j = 1, \ldots, M. \tag{15}$$

In the current implementation of LPM, the particles are located at the centers and vertices of disjoint triangular Lagrangian panels $\{P_k(t)\}$ that cover the sphere and are advected by the flow map,

$$S = \cup_{k=1}^{N} P_k(t), \quad P_k(t) = \boldsymbol{x}(P_k(0), t), \quad k = 1, \ldots, N. \tag{16}$$

Figure 1(a) depicts a triangular panel and its associated particles. The initial panels $\{P_k(0)\}$ are defined by an icosahedral triangulation of the sphere which is constructed as described by [3]. A set of twenty triangles is given by an icosahedron and the vertices are connected to form the first level of panels. A refinement step divides each panel into four subpanels, as shown in figure 1(b), with the new particles projected onto the sphere. The process continues for a specified number of levels, yielding a set of almost uniform spherical triangles, as shown in figure 2(a) after three levels of refinement with $N = 1280$ triangles. These triangles give the initial panels $\{P_k(0)\}$, and the panel centers and vertices define the Lagrangian parameters $\{\boldsymbol{\alpha}_j\}$ for the initial particle positions $\{\boldsymbol{x}_j(0)\}$ in (15).

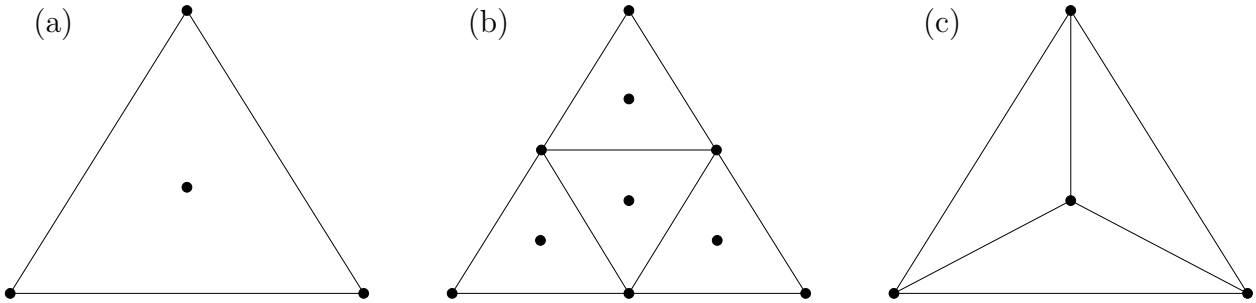

Figure 1: Schematic of panels, (a) each triangular panel has particles (•) at the center and vertices, (b) panel refinement for icosahedral triangulation of the sphere, (c) panel refinement for linear interpolation of tracer density.

While LPM could be implemented using particles only, panels provide several capabilities including a simple midpoint rule quadrature scheme. New particle locations are defined efficiently and locally within each parent panel using the recursive refinement shown in figure 1(b). The refinement procedure imposes a quad-tree organization of the panels data structure which enables fast searching of both uniform and adaptively refined meshes. Panels are also used for plotting purposes.

Table 1 gives the relation between the number of panels N, particles M, and the average grid spacing (angular extent of a panel edge) which satisfies $\Delta\lambda = O(N^{-1/2})$. Note that in LPM the particles are advected and the panels are defined implicitly by their corresponding center and vertex particles.

2.2. Particle advection

The particles are advected by the fluid velocity, leading to a set of ODEs,

$$\frac{d}{dt}\boldsymbol{x}_j(t) = \boldsymbol{u}\big(\boldsymbol{x}_j, t\big), \quad j = 1, \ldots, M, \tag{17}$$

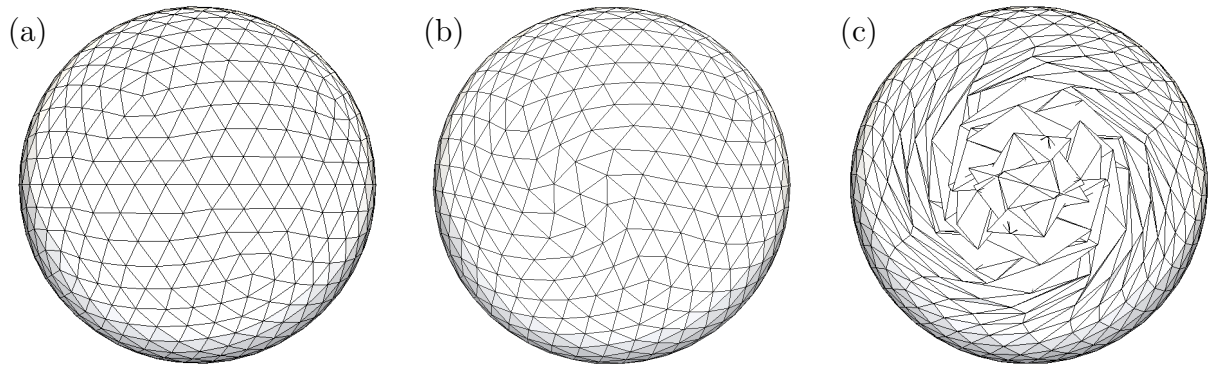

Figure 2: Icosahedral triangulation of the sphere with $N = 1280$ panels, the panels are visualized by connecting their vertex particles, the particles are located at the panel centers and vertices (not shown), (a) initial time, (b) early time, (c) late time. The particle/panel distribution becomes disordered as time proceeds.

Table 1: Icosahedral triangulation of the sphere, N: number of triangles (panels), M: number of particles (panel centers and vertices), $\Delta\lambda$: average grid spacing (angular extent of a panel edge).

N	20	80	320	1280	5120	20480	81920	327680
M	32	122	482	1922	7682	30722	122882	491522
$\Delta\lambda$	63.43°	33.87°	17.22°	8.64°	4.33°	2.16°	1.08°	0.54°

with initial condition given in (15). The particle locations and velocities are expressed in Cartesian coordinates to avoid the singularities occuring in spherical coordinates. Following (12), each particle $\boldsymbol{x}_j(t)$ carries a tracer value obtained by sampling the initial tracer at the Lagrangian parameter,

$$q_j = q_0(\boldsymbol{\alpha}_j), \quad j = 1, \ldots, M, \tag{18}$$

and a density value, $\rho_j(t) = \rho(\boldsymbol{x}(\boldsymbol{\alpha}_j, t), t)$, that obeys the ODE,

$$\frac{d}{dt}\rho_j(t) = -\big(\nabla \cdot \boldsymbol{u}(\boldsymbol{x}_j, t)\big)\rho_j(t), \tag{19}$$

with $\rho_j(0) = 1$.

The Jacobian determinant of the flow map $J(\boldsymbol{\alpha}, t)$ can be used to derive an ODE for panel areas, $A_k(t) = \text{area}[P_k(t)]$, $k = 1, \ldots, N$, $t \geq 0$,

$$\frac{d}{dt}A_k(t) = \big(\nabla \cdot \boldsymbol{u}(\boldsymbol{x}_k, t)\big)A_k(t), \tag{20}$$

where the initial value $A_k(0)$ is computed by a spherical triangle area formula. The ODEs (17), (19), and (20) are integrated by the fourth-order Runge-Kutta method (RK4). We note that the computational work of integrating these three ODEs is not affected by the number of tracers in the simulation; each tracer is implicitly carried by the particles.

Following (14), the tracer integral is approximated using a spherical midpoint rule,

$$Q_N(t) = \sum_{k=1}^{N} \rho_k(t) q_k A_k(t), \tag{21}$$

where q_k is the tracer value carried by the center particle $\boldsymbol{x}_k(t)$ of panel $P_k(t)$. Note that the approximate tracer integral Q_N in (21) is constant since $D(\rho_k A_k)/Dt = 0$, though later on we will see that the value of Q_N jumps in time when remeshing is employed.

2.3. Remeshing

As mentioned earlier, Lagrangian schemes encounter a difficulty when the particles become disordered [49]. Figure 2 shows an example in which the particles are well-ordered at $t = 0$, but they become slightly disordered at early times and then highly disordered at later times. In general when this happens the particles can no longer adequately resolve the tracer density. This can occur even when the particle positions are computed to high accuracy. Using more particles will diminish the problem, but this is expensive and instead the issue is typically addressed by remeshing [2, 4, 7, 30, 43, 48, 60]. In this approach a time is chosen when the particles are still only slightly disordered and they are replaced by a new set of well-ordered particles and panels with re-computed areas. An interpolation scheme transfers the density and tracer from the old particles to the new ones, and the computation resumes using the new particles and tracer values. In LPM the new well-ordered particles are the same as the initial particles (centers and vertices of an icosahedral triangulation) and remeshing is performed after a specified number of time steps. In the following discussion we describe two remeshing schemes that differ in how the new tracer values are obtained.

1. Direct remeshing. The standard approach interpolates the tracer directly from the old particles to the new ones. In this case the output of the interpolation step can be written as

$$\{q_j^{new}\} = I(\{\boldsymbol{x}_j^{new}\}; \{\boldsymbol{x}_j^{old}\}, \{q_j^{old}\}), \tag{22}$$

where I is the interpolation operator. We refer to this as remeshing with direct tracer interpolation, or *direct remeshing*. A variety of interpolation techniques have been employed using moment conserving kernels [30], redistribution schemes [60], non-oscillatory methods [43], and radial basis functions [2, 7]. In the present work the interpolation is carried out using the STRIPACK and SSRFPACK libraries for scattered data interpolation on the sphere [53, 54]. First a Delaunay triangulation of the old particles is constructed and approximate gradient values are computed by local quadratic least squares fitting. Then the new particles are located in the Delaunay triangulation and cubic Hermite interpolation is applied to compute the required output values. The cubic Hermite interpolant is formally fourth-order accurate in the grid spacing $\Delta\lambda$, but since approximate gradients are used, somewhat slower convergence is seen in practice. Nonetheless this improves upon the LPM results reported in the 2011 NCAR workshop on transport schemes where linear interpolation on subtriangles was used as depicted in figure 1(c) [38].

The cost of the Delaunay triangulation algorithm scales as $O(M \log M)$ [53]; however, we find that the cost of remeshing is dominated by the search required to locate an output point $\boldsymbol{x}_j^{new}$ within the Delaunay triangulation of existing particles $\{\boldsymbol{x}_j^{old}\}$. The cost of this search is also $O(M \log M)$ in general and depends upon each search's initial starting point. We observe $O(M)$ search time in practice by using our mesh data structure to inform the starting point of each search.

In the following sections, LPM simulations that use direct remeshing will be annotated as LPM-d. Our description has used the tracer q; the mass density ρ is treated similarly.

2. Indirect remeshing. An alternative approach uses the *inverse flow map*, $\boldsymbol{x} \to \boldsymbol{\alpha}(\boldsymbol{x}, t)$, from physical space to Lagrangian parameter space [11, 12] to take advantage of the fact that the tracer is a material invariant, $Dq/Dt = 0$. There are two steps. The first step interpolates the inverse flow map to find the Lagrangian parameters of the new particles,

$$\{\boldsymbol{\alpha}_j^{new}\} = I(\{\boldsymbol{x}_j^{new}\}; \{\boldsymbol{x}_j^{old}\}, \{\boldsymbol{\alpha}_j^{old}\}). \tag{23}$$

The second step obtains the new tracer values by sampling the initial tracer mixing ratio at the new Lagrangian parameters,

$$q_j^{new} = q_0(\boldsymbol{\alpha}_j^{new}), \quad j = 1, \ldots, M, \tag{24}$$

which is consistent with (18). We refer to this as remeshing by inverse flow map interpolation with sampling of the initial tracer density, or *indirect remeshing*. The interpolation in (23) is also carried out using STRIPACK/SSRFPACK.

LPM simulations that use indirect remeshing are annotated below as LPM-i. With indirect remeshing the quantity being interpolated is the inverse flow map. The flow map and its inverse begin as the identity and vary smoothly in time as a function of the flow velocity $\boldsymbol{u}(\boldsymbol{x}, t)$. As a result, high-order accuracy is possible even for the case of discontinuous tracers. Furthermore, the smoothness of this interpolation problem may be controlled by the choice of the Lagrangian parameter, which may be redefined at some $t > 0$ for long time simulations.

Up to the limits of interpolation error, indirect remeshing preserves the Lagrangian information carried by the flow map. This is a new capability introduced by LPM-i. Additionally, indirect remeshing cannot introduce any new extrema, since its tracer values are assigned by sampling (rather than interpolating) a preexisting tracer distribution.

Some additional comments about remeshing are in order here.

1. The tracer values q_j are constant in time in between remeshing intervals, but they change due to interpolation when remeshing is performed. Hence from now on we write $q_j(t)$ to indicate this time dependence.
2. The initial particles and newly remeshed particles lie on the sphere, but since the ODEs (17) are integrated in Cartesian coordinates, the particles are not constrained to lie on the sphere in between remeshing intervals. However if the ODEs are solved with sufficient accuracy, as for the results presented below, then the particles remain close to the sphere.
3. Direct remeshing requires a separate interpolation for each tracer. Indirect remeshing interpolates the three components of the Lagrangian parameter and reuses this data for each tracer. Hence, whenever the number of tracers is ≥ 3, which is common in modern climate models, indirect remeshing is likely more efficient. In this work the test cases use no more than 2 tracers and we do not notice a significant difference in the time required by each strategy.
4. We observe an overall approximate $O(M)$ scaling in wall-clock time as the number of particles M increases for the time-stepping subroutine as well as both direct and indirect remeshing procedures, yielding an overall $O(M)$ scaling for each LPM scheme. Section 4.1 includes a timing experiment that documents this result.

2.4. Error norms

To assess the accuracy of the computed tracer we use the following error norms [61],

$$l_\infty = \frac{\max\{|q_j(t) - q(\boldsymbol{x}_j(t), t)|\,; j = 1, \ldots, M\}}{\max\{|q(\boldsymbol{x}_j(t)|\,; j = 1, \ldots, M\}}, \tag{25a}$$

$$l_2 = \left(\frac{\displaystyle\sum_{k=1}^{N} (q_k(t) - q(\boldsymbol{x}_k(t), t))^2 A_k}{\displaystyle\sum_{k=1}^{N} q(\boldsymbol{x}_k(t), t)^2 A_k} \right)^{1/2}. \tag{25b}$$

Recall that index j runs over the particles and index k runs over the panels. Also note in (25a) that $q_j(t)$ is the tracer value carried by particle j, while $q(\boldsymbol{x}_j(t), t)$ is obtained by sampling the exact tracer $q(\boldsymbol{x}, t)$ at the particle location $\boldsymbol{x}_j(t)$. There are two sources of error to consider. First, the tracer values $q_j(t)$ have interpolation error due to the remeshing scheme and this is controlled by the grid spacing $\Delta\lambda$. Second, the particle locations $\boldsymbol{x}_j(t)$ have time-stepping error due to integrating the ODEs (17) and this is controlled by the time step Δt. Here we choose Δt sufficiently small so that the time-stepping error in $\boldsymbol{x}_j(t)$ is negligible compared to the interpolation error in $q_j(t)$. Hence we will focus on how the error depends on the grid spacing $\Delta\lambda$. These comments also apply to $q_k(t)$ and $\boldsymbol{x}_k(t)$ in (25b).

To assess tracer conservation we report the tracer integral error defined by

$$l_Q = \frac{|Q_N(t) - Q_N(0)|}{|Q_N(0)|}, \tag{26}$$

where $Q_N(t)$ is the approximate tracer integral in (21). As with the tracer values $q_j(t)$, the tracer integral $Q_N(t)$ is constant in time in between remeshing intervals, but it changes due to interpolation when remeshing is performed.

2.5. Additional information

The computations were done in modern Fortran on a standard workstation, except for the timing experiment presented in section 4.1, which ran on one node of a high performance computing cluster. The output from LPM is a set of particle positions and corresponding density and tracer values which are presented in two ways. (1) To plot the tracer on the sphere, each panel is split into three subpanels by connecting the center to the vertices, as in figure 1(c), and a ParaView plotting routine is employed on each subpanel [1]. (2) To plot the tracer on a planar latitude-longitude grid, we use STRIPACK/SSRFPACK to interpolate the tracer from the particles to the grid [53, 54], and then use an NCL filled-contour plotting routine on the grid [47].

We compare LPM results to those obtained using the Lin-Rood finite-volume scheme [41, 42] which is representative of the Eulerian schemes reported in [38]. The Lin-Rood scheme uses two one-dimensional steps which are carefully combined to reduce the dimension-splitting error. Here it was implemented in spherical coordinates on a uniform latitude-longitude grid with grid spacing $\Delta\lambda$, using the flux-form semi-Lagrangian extension [41]. Fluxes were computed by the piecewise parabolic method (PPM) which uses a parabolic

subgrid distribution and fourth-order edge reconstruction with a limiter to enforce monotonicity [14].

We consider two test cases, a moving-vortices flow [44] and reversing-deformational flows [37] with two variants, one divergence-free and one with non-zero velocity divergence. The velocity fields are defined using (λ, θ) as longitude and latitude, and (u, v) are the corresponding velocities. The sphere radius is $a = 6371$ km, the computations extend to time $T = 12$ days, and $\Omega = 2\pi/T$ is the angular velocity. The LPM results used grid spacing $0.54° \leq \Delta\lambda \leq 33.87°$ and time step $\Delta t = T/400 = 43.2$ minutes with remeshing every 20 time steps. For this value of Δt the LPM computations are sufficiently converged in time. Decreasing Δt further does not significantly change our results; due to the high order time discretization relative to the spatial discretization, spatial discretization error is dominant and time stepping error is negligible. The Lin-Rood results used grid spacing $\Delta\lambda = 0.5°, 1°, 2°$ with time step $\Delta t = T/4000, T/2000, T/1000$, respectively.

3. Test case 1: Moving-vortices flow

In this example two antipodal counter-rotating vortices move along the equator with angular velocity Ω [44]. The velocity field is

$$u(\lambda, \theta, t) = \omega(r(\lambda, \theta, t)) \sin(\lambda - \Omega t) \sin\theta + a\Omega \cos\theta, \tag{27a}$$

$$v(\lambda, \theta, t) = \omega(r(\lambda, \theta, t)) \cos(\lambda - \Omega t), \tag{27b}$$

where

$$r(\lambda, \theta, t) = 3(1 - \cos^2\theta \sin^2(\lambda - \Omega t))^{1/2} \tag{27c}$$

is the radial distance from the center of a vortex, and the profile function is

$$\omega(r) = \begin{cases} \dfrac{3\sqrt{3}\, a\Omega}{2} \dfrac{\operatorname{sech}^2(r) \tanh(r)}{r} & \text{if } r \neq 0, \\ 0 & \text{if } r = 0. \end{cases} \tag{27d}$$

The tracer density is

$$q(\lambda, \theta, t) = 1 - \tanh\left[0.2 r(\lambda, \theta, t) \sin\left(\lambda'(\lambda, \theta, t) - \frac{\omega(r)}{a} t\right)\right], \tag{28a}$$

where

$$\lambda'(\lambda, \theta, t) = \arctan\left[\frac{-\cos(\lambda - \Omega t)}{\tan\theta}\right]. \tag{28b}$$

Figures 3(a,b,c) show the exact solution. Figure 3(a) shows the vorticity which has a concentrated core that travels around the equator and returns to its initial location at time $t = T$. Figure 3(b) shows the initial tracer density which decreases smoothly from $q \approx 1.5$ in the west to $q \approx 0.5$ in the east. Following previous work, the tracer color bar depicts the contour $q = 1$ as a sharp interface [44]. For $t > 0$ the interface rolls up into a spiral as shown by in figure 3(c). Note that the example of particle disorder in figure 2 was obtained by advecting particles in this flow.

Figures 3(d,e,f) show LPM results with direct remeshing and grid spacing $\Delta\lambda = 8.64°$, $4.33°$, $2.16°$. Figures 3(g,h,i) show the pointwise error distribution at $t = T$ for the same $\Delta\lambda$

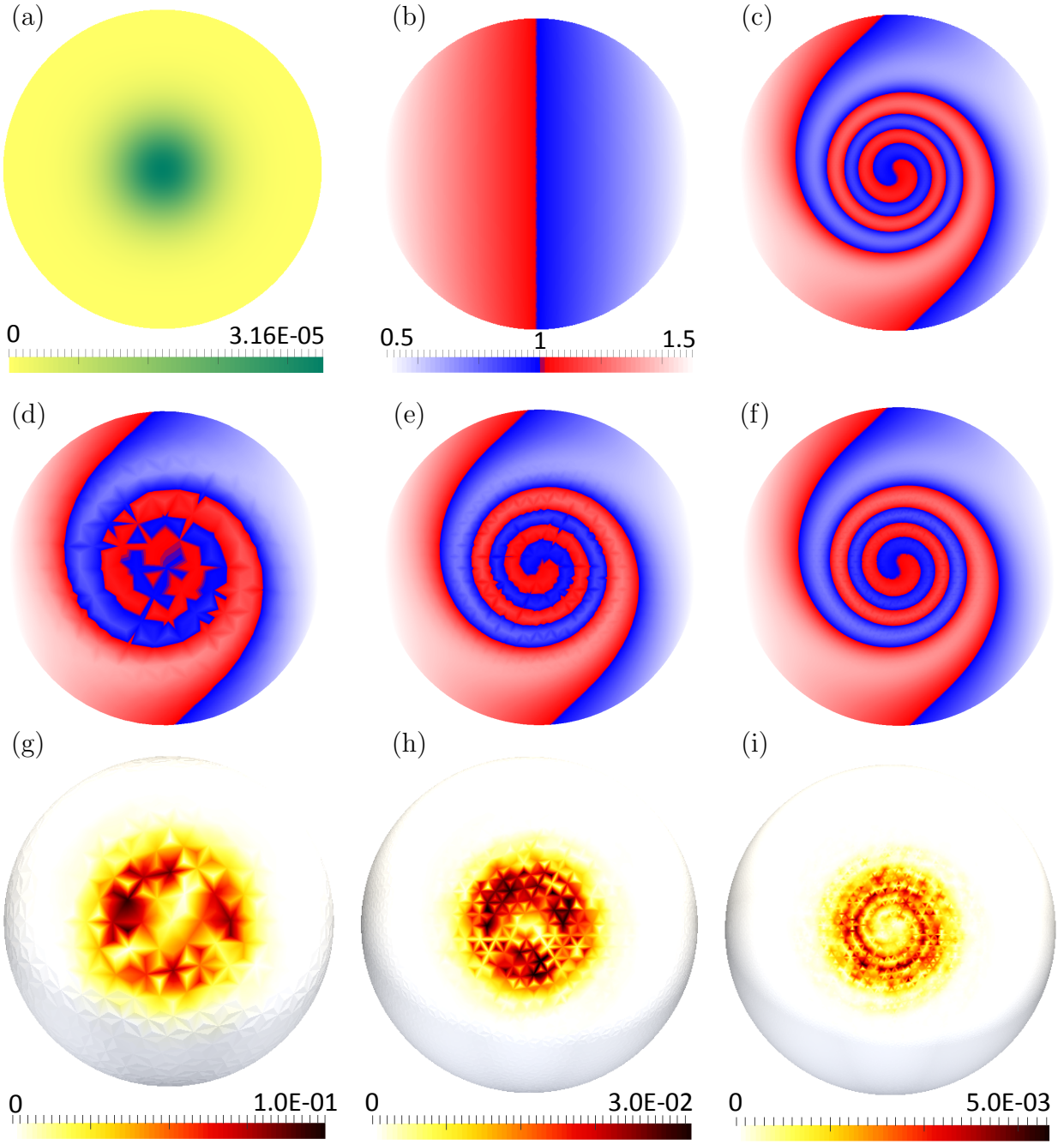

Figure 3: Moving-vortices flow, (a,b,c) exact solution, (a) vorticity, $t = 0$, (b) tracer, $t = 0$, (c) tracer, $t = T$; (d,e,f) LPM tracer solution and (g,h,i) error using direct remeshing at $t = T$ with grid spacings (d,g) $\Delta\lambda = 8.64°$; (e,h) $\Delta\lambda = 4.33°$; (f,i) $\Delta\lambda = 2.16°$.

values. The numerical results converge to the exact solution in figure 3(c) as $\Delta\lambda$ is reduced, and in this case there was no significant difference between direct and indirect remeshing. Next we examine the error quantitatively.

Figure 4(a) presents the tracer error (l_∞, l_2) at time $t = T$ for LPM with direct and indirect remeshing and for the Lin-Rood scheme. In all cases the l_2 error (dashed lines) is smaller than the l_∞ error (solid lines). For comparable values of the grid spacing $\Delta\lambda$, LPM is more accurate than the Lin-Rood scheme. For LPM the two remeshing schemes give almost identical errors and the rate of convergence is close to $O(\Delta\lambda^4)$.

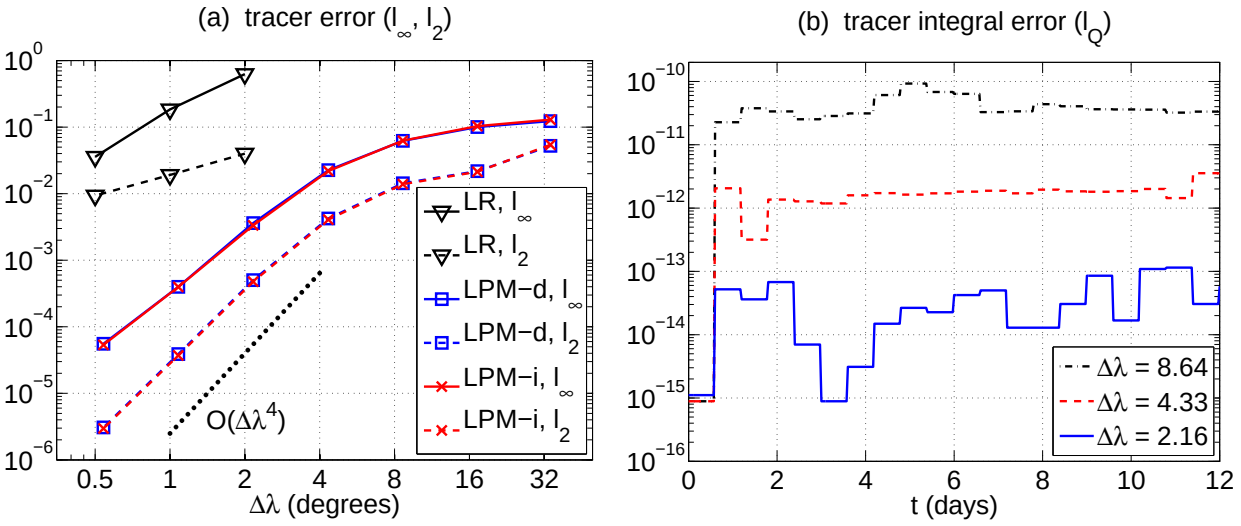

Figure 4: Moving-vortices flow; (a) tracer error at time $t = T$, l_∞ (solid lines), l_2 (dashed lines), Lin-Rood ($\triangledown$), LPM with direct remeshing ($\square$), LPM with indirect remeshing ($\times$), (b) tracer integral error l_Q for $0 \le t \le T$, LPM with direct remeshing, grid spacing $\Delta\lambda$ decreases from top to bottom.

Next we discuss the conservation of the tracer integral (7), (21). As a finite-volume method, the Lin-Rood scheme conserves the tracer integral to machine precision. On the other hand LPM with remeshing does not enforce tracer conservation, so it is important to understand how the scheme behaves in practice. Figure 4(b) shows the tracer integral error l_Q for $0 \le t \le T$ using LPM with direct remeshing (indirect remeshing gave comparable results). The tracer integral error is almost flat in time; the jumps occur at the remeshing intervals and the greater variation for $\Delta\lambda = 2.16°$ is attributed to rounding error in evaluating l_Q. For grid spacing $\Delta\lambda = 8.64°$ the tracer integral error is less than 10^{-10}, and it decreases at least one order of magnitude when $\Delta\lambda$ is cut in half.

4. Test case 2: Reversing-deformational flow

The non-divergent velocity field for the reversing-deformational flow is defined by

$$u(\lambda, \theta, t) = \frac{10a}{T} \sin^2(\lambda - \Omega t) \sin(2\theta) \cos(\Omega t/2) + a\Omega \cos(\theta), \tag{29a}$$

$$v(\lambda, \theta, t) = \frac{10a}{T} \sin(2(\lambda - \Omega t)) \cos(\theta) \cos(\Omega t/2). \tag{29b}$$

which deforms for $0 \le t \le T/2$ and then reverses for $T/2 \le t \le T$ [46]. The tracer returns to its initial state at the final time, $q(\boldsymbol{x}, T) = q(\boldsymbol{x}, 0)$, and this property is used to compute the

error in the numerical method. We tested three tracer densities, (1) a pair of Gaussian hills, (2) two cosine bell pairs with nonlinear correlation, and (3) a pair of slotted cylinders. Each tracer is defined relative to two center points on the equator, chosen to be $(\lambda_1, \theta_1) = (5\pi/6, 0)$ and $(\lambda_2, \theta_2) = (7\pi/6, 0)$. The great-circle distance between a point on the sphere (λ, θ) and the center points (λ_i, θ_i) for $i = 1, 2$ is given by

$$r_i(\lambda, \theta) = a \arccos(\sin\theta_i \sin\theta + \cos\theta_i \cos\theta \cos(\lambda - \lambda_i)). \tag{30}$$

4.1. Gaussian-hills tracer

The first example is test case 1 from [37, 38]. The initial tracer density is the sum of two Gaussians,

$$\begin{aligned} q_0(x, y, z) = 0.95 \exp\left[-5\left((x - x_1)^2 + (y - y_1)^2 + z^2\right)\right] \\ + 0.95 \exp\left[-5\left((x - x_2)^2 + (y - y_2)^2 + z^2\right)\right], \end{aligned} \tag{31}$$

where $x_{1,2} = a\cos\lambda_{1,2}$, $y_{1,2} = a\sin\lambda_{1,2}$, and (x, y, z) are Cartesian coordinates on the sphere. In this case the tracer is smooth, but small-scale features develop for $t > 0$.

Figure 5 presents LPM results at time $t = 0, T/2, T$ with grid spacing $\Delta\lambda = 8.64°$ using (a) direct remeshing and (b) indirect remeshing. For plotting purposes the computed tracer was interpolated from the particles to a latitude-longitude grid. At $t = T/2$ the tracer is deformed into a pair of thin filaments, and at $t = T$ the initial condition is restored. With direct remeshing the numerical solution at $t = T/2, T$ has undershoots and the peak values are affected by diffusive smoothing. With indirect remeshing these artifacts are absent and the solution is more accurate. Next we document this quantitatively.

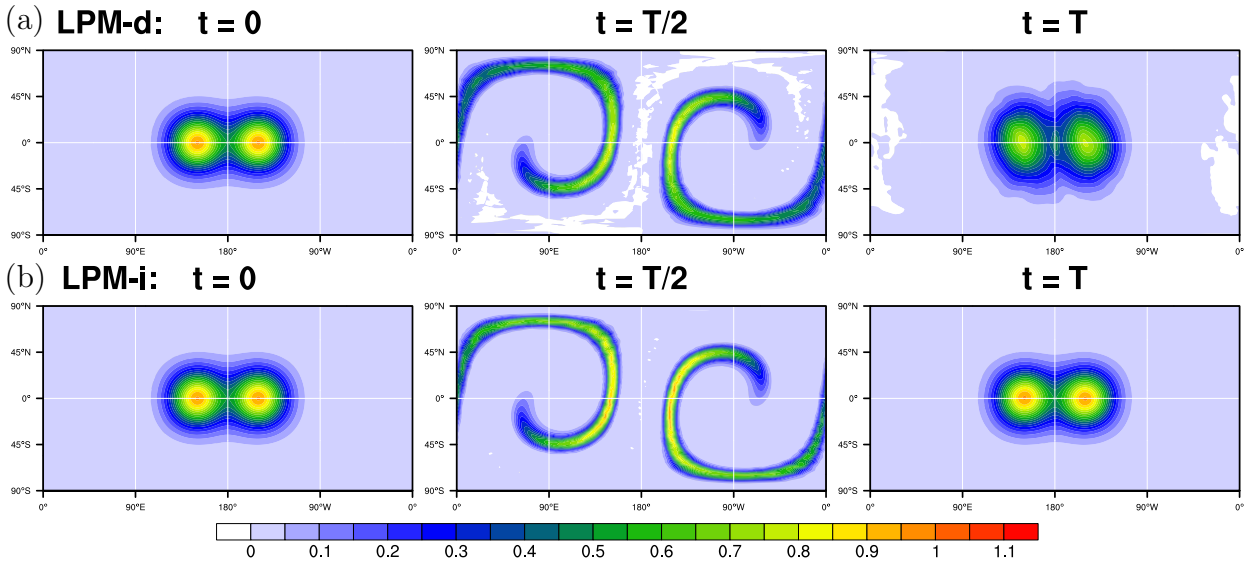

Figure 5: Non-divergent reversing-deformational flow (29), Gaussian-hills tracer (31), LPM results at $t = 0, T/2, T$ with grid spacing $\Delta\lambda = 8.64°$, (a) direct remeshing, (b) indirect remeshing.

Figure 6(a) presents the tracer error (l_∞, l_2) at time $t = T$ for LPM with direct and indirect remeshing and for the Lin-Rood scheme. The l_2 error (dashed lines) is slightly smaller than the l_∞ error (solid lines). For comparable values of the grid spacing $\Delta\lambda$, LPM is more accurate than the Lin-Rood scheme and indirect remeshing is more accurate than

direct remeshing. The LPM results converge at a rate close to $O(\Delta\lambda^4)$. Figure 6(b) shows the tracer integral error l_Q for $0 \le t \le T$ using LPM with indirect remeshing. The tracer integral error is larger than in the previous test case, but it is almost flat in time and it decreases as the grid spacing $\Delta\lambda$ is reduced.

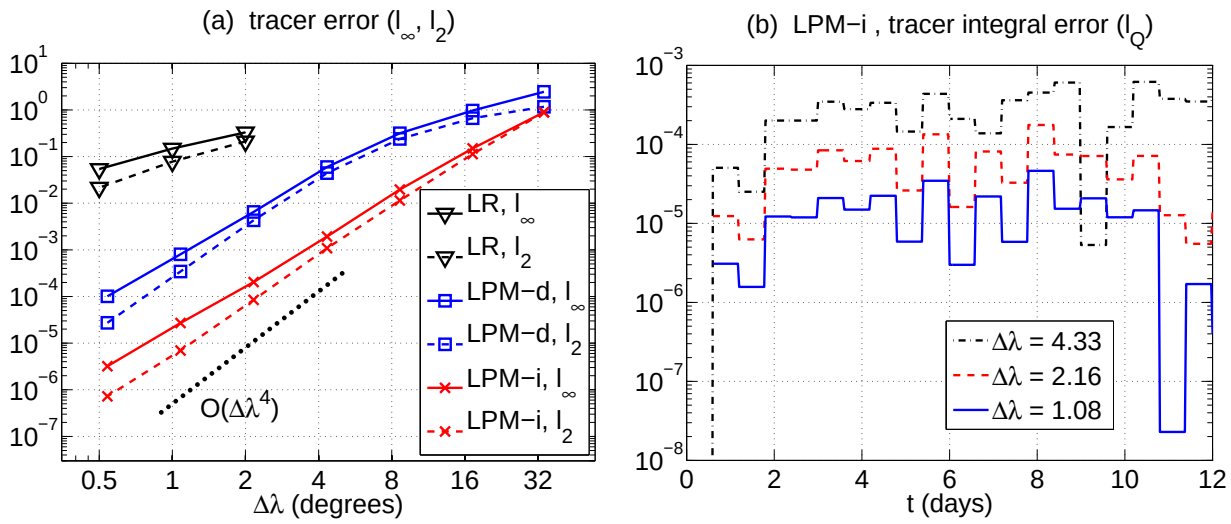

Figure 6: Non-divergent reversing-deformational flow (29), Gaussian-hills tracer (31); (a) tracer error at $t = T$, l_∞ (solid lines), l_2 (dashed lines), Lin-Rood ($\triangledown$), LPM with direct remeshing ($\square$), LPM with indirect remeshing ($\times$), (b) tracer integral error l_Q for $0 \le t \le T$, LPM with indirect remeshing, grid spacing $\Delta\lambda$ decreases from top to bottom.

Next we report LPM timing results for the Gaussian hills test case running on one node on the Sandia National Laboratories Sky Bridge high performance computing cluster (2.6 GHz Intel Sandy Bridge, 64GB RAM per node) using the Intel Fortran compiler version 15.0 with standard compiler optimizations. We use OpenMPI version 1.8 to parallelize the time-stepping subroutine and we report results for a computation that used four cores. The runtime reported here includes time-stepping, remeshing, and file input/output. Figure 7(a) shows the runtime vs. the number of particles M for direct remeshing (LPM-d) and indirect remeshing (LPM-i). The runtime scales linearly, like $O(M)$ with the number of particles M, and the two remeshing schemes require essentially the same runtime. It can be noted that the runtime is modest; even for the largest number of particles treated here, $M = 327680$, the runtimes for both LPM-d and LPM-i were approximately 10 minutes. Computations on these benchmark test cases are relatively inexpensive because the velocity field is given analytically, as opposed to being computed by a fluid solver. Figure 7(b) shows the runtime vs. tracer error, where each data point corresponds to a value of M. It can be seen that for a given amount of runtime, indirect remeshing yields smaller tracer error than direct remeshing.

4.2. Correlated cosine-bell tracers

It is desireable for transport schemes to preserve tracer correlations [39, 58] and the Lin-Rood scheme is designed to preserve linear correlations [41]. Here we consider test case 5 from [37, 38] involving two nonlinearly correlated tracers, q_1 and q_2, whose initial

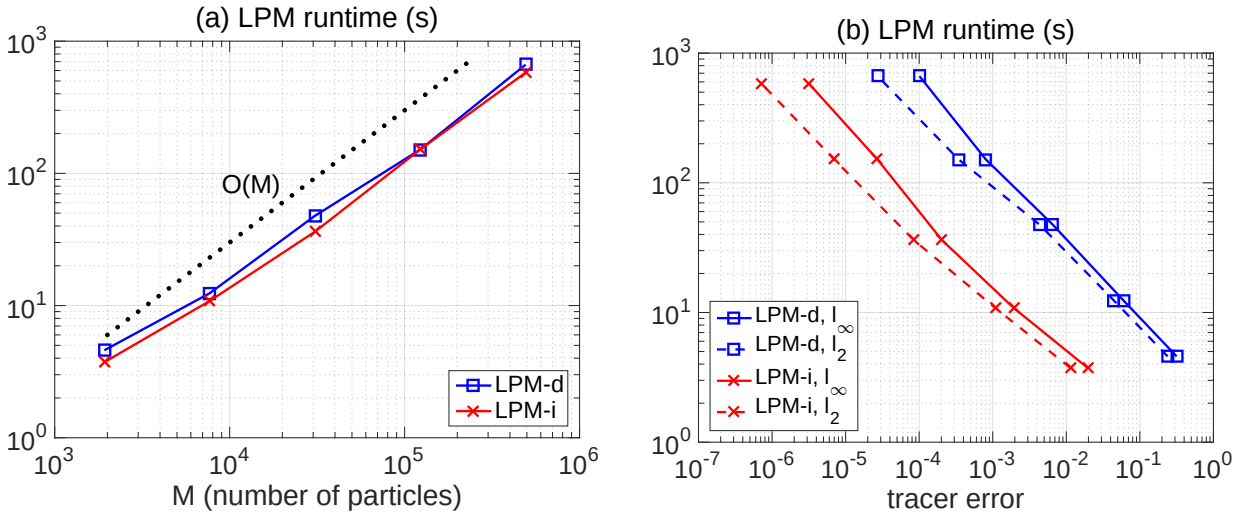

Figure 7: Non-divergent reversing-deformational flow (29), Gaussian-hills tracer (31), LPM runtime (s) for $0 \le t \le T$, direct remeshing (LPM-d, □), indirect remeshing (LPM-i, ×), (a) runtime vs. M (number of particles), (b) runtime vs. error l_∞ (solid lines), l_2 (dashed lines).

distributions are given by q_{10} and q_{20},

$$q_{10}(\lambda,\theta) = \begin{cases} 0.1 + 0.45\,(1+\cos(2\pi\, r_1(\lambda,\theta)/a)) & \text{if } r_1(\lambda,\theta) < \frac{a}{2}, \\ 0.1 + 0.45\,(1+\cos(2\pi\, r_2(\lambda,\theta)/a)) & \text{if } r_2(\lambda,\theta) < \frac{a}{2}, \\ 0.1 & \text{otherwise,} \end{cases} \tag{32a}$$

$$q_{20}(\lambda,\theta) = -0.8 q_{10}^2(\lambda,\theta) + 0.9, \tag{32b}$$

so that q_{10} has two cosine bells above a constant backgound and q_{20} is a quadratic function of q_{10}. Figure 8 plots tracer q_2 against tracer q_1 at time $t = T/2$ for the Lin-Rood scheme and LPM with direct remeshing and indirect remeshing. The computed results are plotted as red circles and the exact quadratic relation is a blue line. The Lin-Rood and LPM-d results have errors in which the maximum values of q_1 and the minimum values of q_2 are truncated, a sign of numerical diffusion [38]. Both Lin-Rood and LPM-d become more accurate as the grid spacing $\Delta\lambda$ is reduced, but LPM-d is more accurate. The LPM-i results are accurate to machine precision for all $\Delta\lambda$; this is because indirect remeshing samples the initial tracer densities for which the correlation (32b) holds by definition.

4.3. Slotted-cylinders tracer

The next example is test case 4 in [37, 38]. The initial tracer density is a pair of slotted-cylinders,

$$q_0(\lambda,\theta) = \begin{cases} 1 & \text{if } r_i(\lambda,\theta) < \frac{a}{2} \text{ and } |\lambda-\lambda_i| \ge \frac{1}{12} \text{ for } i = 1,2, \\ 1 & \text{if } r_1(\lambda,\theta) < \frac{a}{2} \text{ and } |\lambda-\lambda_1| < \frac{1}{12} \text{ and } \theta-\theta_1 < -\frac{5}{24}, \\ 1 & \text{if } r_2(\lambda,\theta) < \frac{a}{2} \text{ and } |\lambda-\lambda_2| < \frac{1}{12} \text{ and } \theta-\theta_2 > \frac{5}{24}, \\ 0.1 & \text{otherwise.} \end{cases} \tag{33}$$

In this case the tracer is discontinuous. Results are presented in figure 9. The top row (a) shows Lin-Rood results with grid spacing $\Delta\lambda = 1.5°$ and time step $\Delta t = T/800$, in which

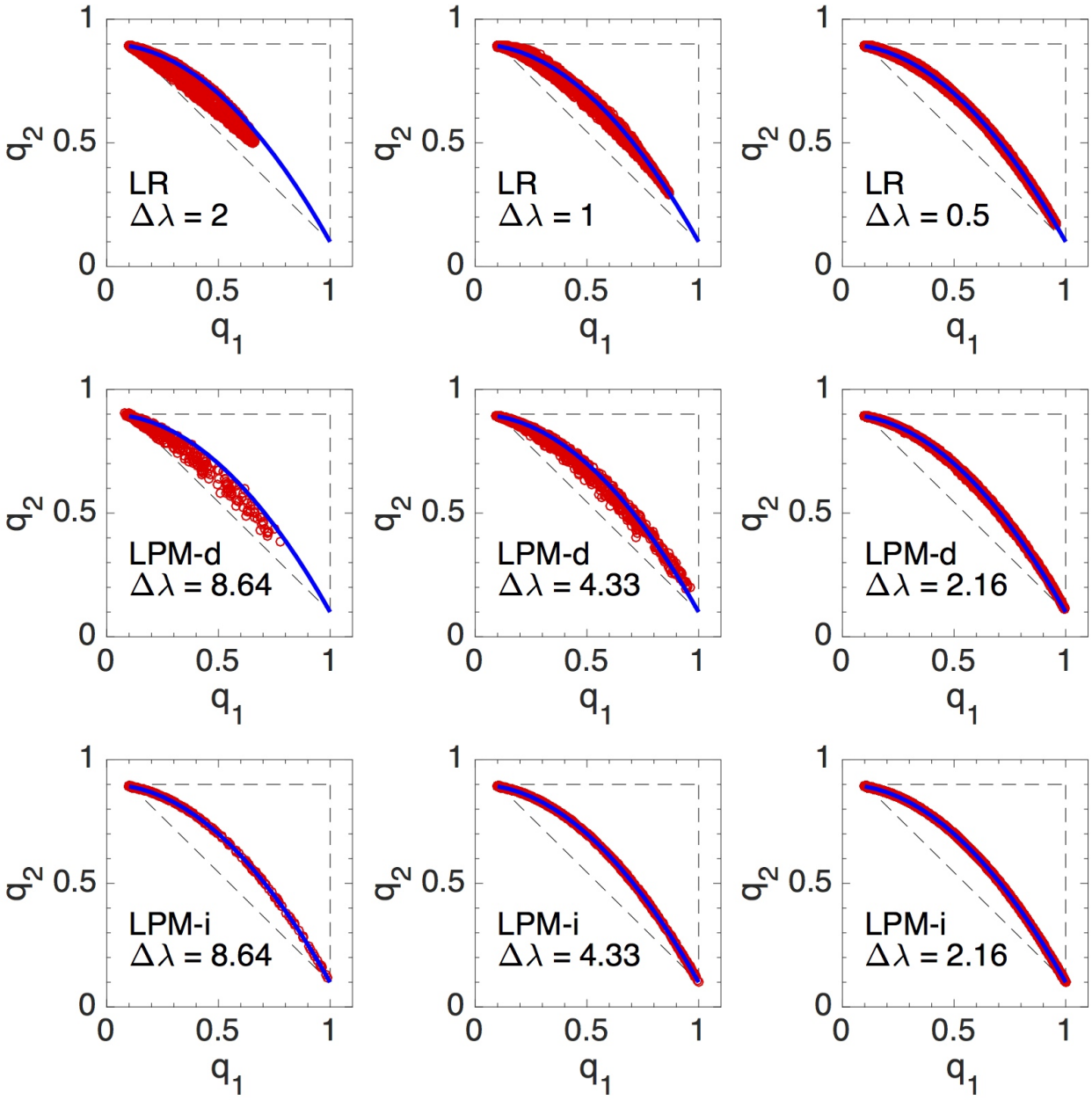

Figure 8: Non-divergent reversing-deformational flow (29), correlated cosine-bell tracers (32), tracer q_2 is plotted against tracer q_1 at $t = T/2$, exact quadratic relation (blue, −), computed results (red, o), Lin-Rood (LR), LPM with direct remeshing (LPM-d) and indirect remeshing (LPM-i), grid spacing $\Delta\lambda$ as indicated. Dashed lines provide a visual aid for tracer bounds.

the computed solution at $t = T/2, T$ has undershoots and diffusive errors. The middle and bottom rows show LPM results with $\Delta\lambda = 2.16°$ and $\Delta t = T/400$. The middle row (b) used direct remeshing and the results are an improvement over (a), but errors are still evident. The bottom row (c) used indirect remeshing and the results are further improved; while close examination reveals that the cylinder edges are jagged due to the finite resolution provided by $\Delta\lambda = 2.16°$, there are no visible undershoots or diffusive errors.

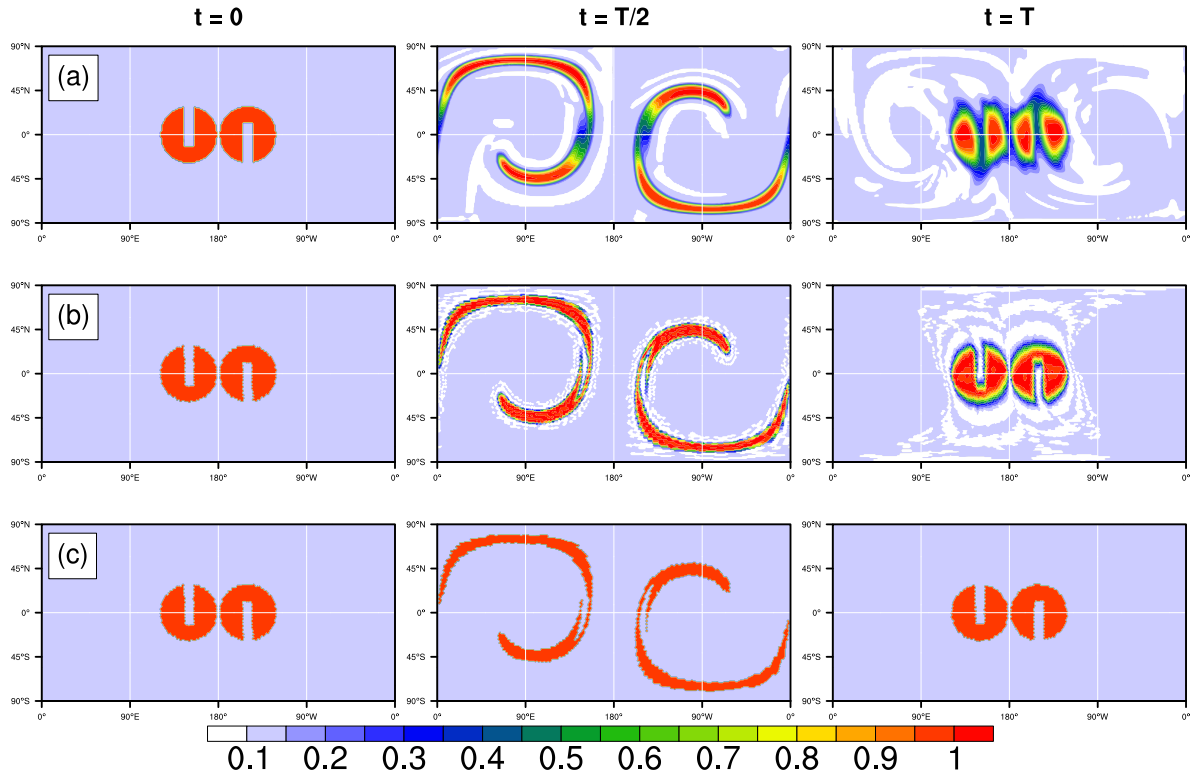

Figure 9: Non-divergent reversing-deformational flow (29), slotted-cylinders tracer (33), results at $t = 0, T/2, T$, (a) Lin-Rood with $\Delta\lambda = 1.5°, \Delta t = T/800$, (b) LPM with direct remeshing, (c) LPM with indirect remeshing, LPM used $\Delta\lambda = 2.16°, \Delta t = T/400$.

4.4. Velocity with non-zero divergence

To demonstrate the solution of a divergent velocity field, we use test case 6 from [37] with the Gaussian hills tracer (31). The flow velocity and divergence are given by

$$u(\lambda, \theta, t) = -\frac{5a}{T} \sin^2\left((\lambda - \Omega t)/2\right) \sin(2\theta) \cos^2(\theta) \cos(\Omega t/2) + a\Omega \cos(\theta), \tag{34a}$$

$$v(\lambda, \theta, t) = \frac{5a}{2T} \sin(\lambda - \Omega t) \cos^3(\theta) \cos(\Omega t/2), \tag{34b}$$

$$\nabla \cdot \boldsymbol{u}(\lambda, \theta, t) = -\frac{15}{T} \cos^2(\theta) \sin(\theta) \sin(\lambda - \Omega t) \cos(\Omega t/2). \tag{34c}$$

In this case the density ρ is no longer constant along particle trajectories and particle density values at a remeshing step $t > 0$ will not equal the sampled value from $\rho_0(\boldsymbol{x})$. We therefore use direct remeshing for ρ. Here, LPM-d implies that we also use direct remeshing for q. LPM-i denotes the use of indirect remeshing for q combined with direct remeshing of ρ.

Figure 10(a) presents the tracer error (l_∞, l_2) at time $t = T$ for for the Lin-Rood scheme and both LPM schemes. The results are similar to the figure 6 results from section 4.1. Again, LPM is more accurate than Lin-Rood for comparable grid spacings, and indirect remeshing is more accurate than direct remeshing. The l_∞ and l_2 errors for both LPM schemes converge at a rate close to fourth order. Figure 10(b) shows the tracer integral error l_Q for $0 \le t \le T$ using LPM-i. As in Figure 6(b), the tracer integral error grows initially to a saturation point, then does not show obvious growth in time.

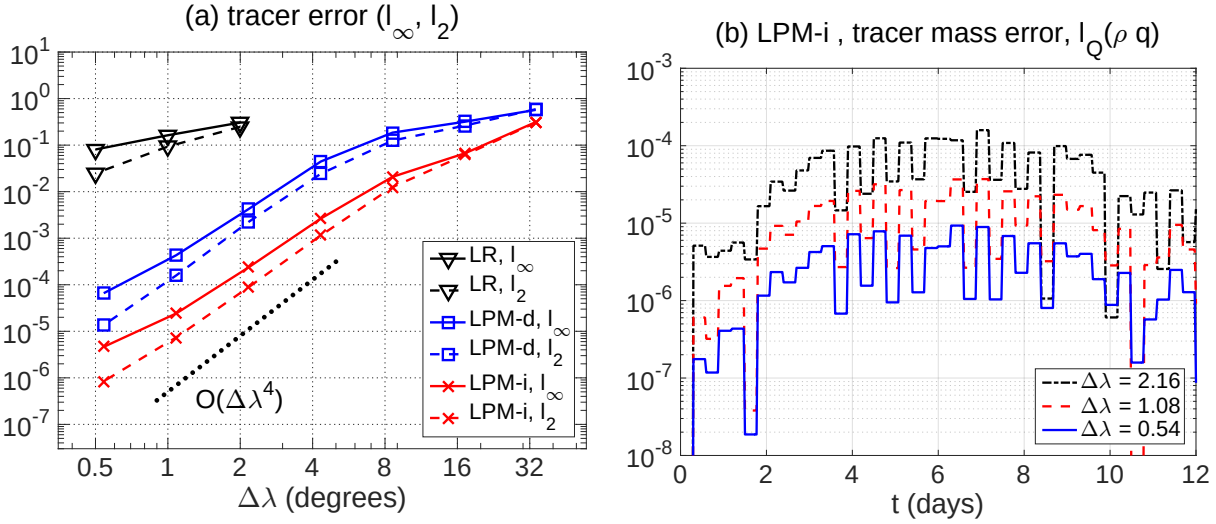

Figure 10: Reversing-deformational flow with non-zero divergence (34), Gaussian-hills tracer (31); (a) tracer error at $t = T$, l_∞ (solid lines), l_2 (dashed lines), Lin-Rood ($\triangledown$), LPM with direct remeshing ($\square$), LPM with indirect remeshing ($\times$), (b) tracer integral error l_Q for $0 \le t \le T$, LPM with indirect remeshing, grid spacing $\Delta\lambda$ decreases from top to bottom.

5. Adaptive refinement

In previous sections we employed LPM with uniform refinement, starting from the 20 panels of an icosahedron and applying a specified number of refinement steps to each panel. Here we demonstrate an adaptive scheme that starts from a coarse uniform mesh and recursively refines each panel until a certain criterion is satisfied [20]. This idea is applied to the moving-vortices flow with a refinement criterion that bounds the circulation in each panel,

$$\frac{|\zeta(\boldsymbol{x}_k)|\, A_k}{\Gamma_{\max}} < \epsilon, \quad k = 1, \ldots, N, \tag{35}$$

where $\zeta(\boldsymbol{x}_k)$ is the relative vorticity at the center particle of panel k, A_k is the panel area, $\Gamma_{\max}$ is the maximum circulation over all panels in the initial coarse mesh, and ϵ is a user-specified tolerance. This criterion is recommended by [6] for atmospheric modeling and shown to work well with LPM in [12]. The vorticity $\zeta(\boldsymbol{x})$ is computed analytically by differentiating the relative velocity, $(u_R, v_R) = (u - a\Omega\cos\theta, v)$, where (u, v) are defined in (27).

Figure 11 presents LPM results with indirect remeshing and adaptive refinement. In this computation, adaptive refinement is applied in each remeshing step starting from an icosahedral triangulation with $N = 1280$ panels. The tolerance is $\epsilon = 0.025$, leading to four

levels of refinement in the vortex core, with grid spacing from $\Delta\lambda = 8.64°$ to $\Delta\lambda = 0.54°$. The number of panels varies slightly at each remeshing step, but satisfies $5140 \leq N \leq 5174$ throughout the computation. The computed tracer rolls up into a smooth spiral and at time $t = T$ the computed results are visually indistinguishable from the exact solution in figure 3(c). The distribution of error is shown in figures 11(g,h,i) at $t = T/4, T/2, T$, respectively. Note that the scale of figure 11(i) whose maximum is 5.0E-03 is identical to the $N = 20480$ ($\Delta\lambda = 2.16°$) computation shown in figure 3(i). In the uniform computations of figure 3 the largest errors lie in the vortex core while the error in figure 11(i) is largest near the refinement boundaries and is much reduced in the vortex core. This shows that the refinement successfully reduced the largest error present in the uniform computations, and that the remeshing strategy is not adversely affected by changing spatial resolutions. Next we compare these two computations directly.

Figure 12(a) plots the tracer errors l_∞, l_2 for $0 \leq t \leq T$, comparing the present adaptive computation using $N \leq 5174$ panels with the uniform computation using $N = 20480$ panels shown in figure 3(f,i). The two computations have similar tracer errors even though the adaptive computation uses approximately 1/4 the number of panels. Figure 12(b) shows that the tracer integral error in the uniform computation is less than 10^{-13}, while in the adaptive computation it is less than 10^{-10}. As before, after the initial remeshing neither computation shows growth in the tracer integral error as time increases. These two plots show that LPM compares favorably to other adaptive methods, e.g., [44, figure 7] and [22, table 2].

6. Summary and conclusions

A Lagrangian particle method (LPM) based on the flow map was presented for tracer transport on the sphere. The particles carry tracer values and are located at the centers and vertices of triangular Lagrangian panels. The particles are advected in the given velocity field by fourth-order Runge-Kutta time-stepping. Remeshing is applied to avoid errors due to particle disorder and two schemes were considered; *direct remeshing* interpolates the tracer directly, while *indirect remeshing* interpolates the inverse flow map and then samples the initial tracer density. Remeshing is carried out using the STRIPACK/SSRFPACK libraries which are based on cubic Hermite interpolation [53, 54]. The test cases included a moving-vortices flow [44] and reversing-deformational flow with and without divergence using smooth and discontinuous tracers [37, 38]. We examined the tracer error, tracer integral error, and preservation of nonlinear correlation in a pair of tracers. We compared results obtained using LPM and the Lin-Rood finite-volume scheme [41, 42]. An adaptive particle/panel refinement scheme was demonstrated. The main results are summarized as follows.

- The time stepping portion of this advection scheme, integrating equations (17), (19), and (20) in time, is independent of the number of tracers. Direct remeshing interpolates each tracer independently whereas indirect remeshing interpolates the inverse flow map and resuses this data for all tracers.

- For the tracer mixing ratio, LPM with remeshing converges at a rate close to $O(\Delta\lambda^4)$, where $\Delta\lambda$ is the average grid spacing [figures 4(a), 6(a), 10(a)]. The tracer integral converges more slowly due to the use of a midpoint rule approximation (21), but the

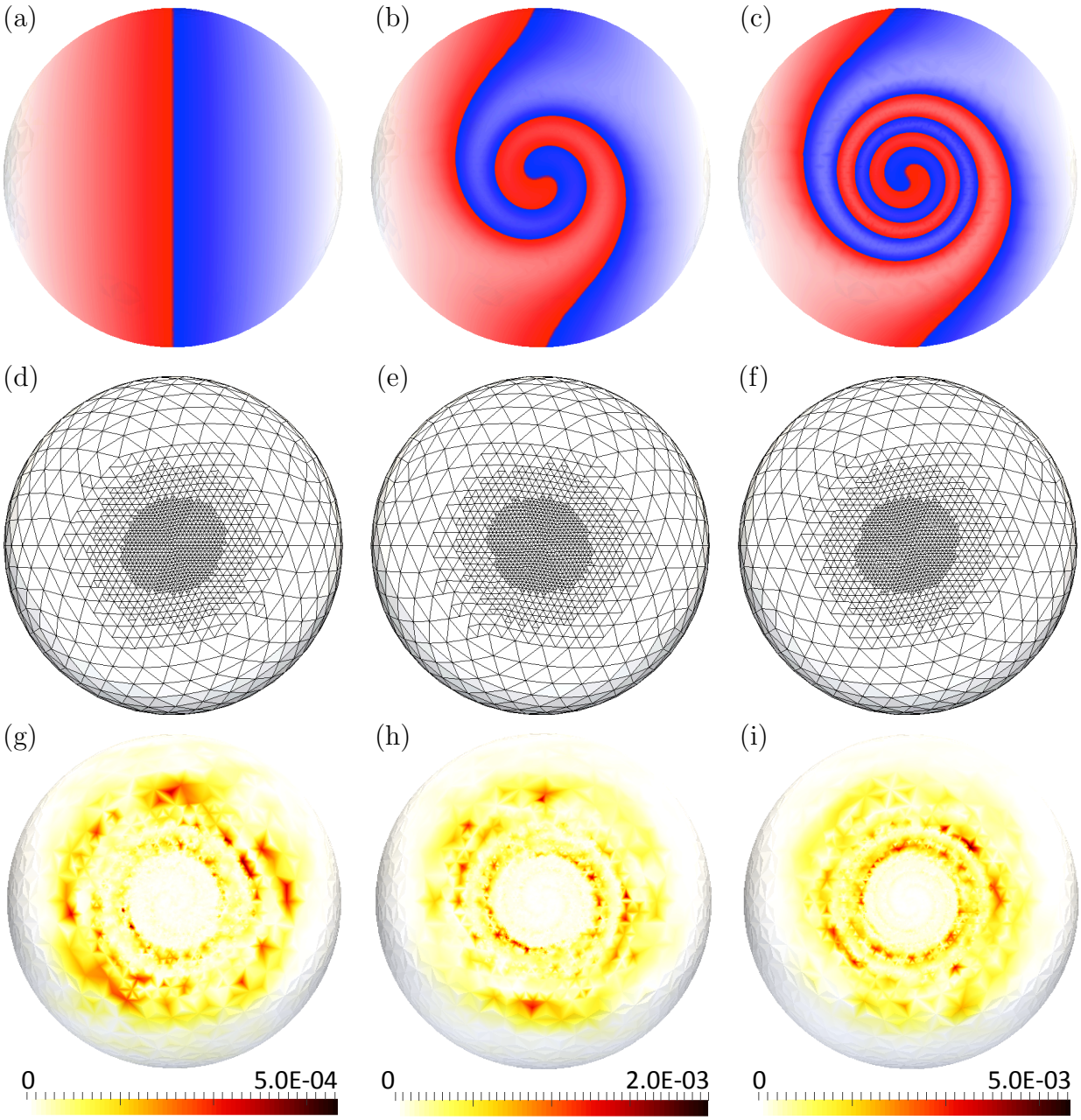

Figure 11: Moving-vortices flow, LPM with indirect remeshing and adaptive refinement, $0.54° \leq \Delta\lambda \leq 8.64°$; (a,b,c) tracer and (d,e,f) panels at (a,d) $t = 0, N = 5132$, (b,e) $t = T/4, N = 5174$, (c,f) $t = T, N = 5144$; (g) error at $t = T/4$, (h) error at $t = T/2$, (i) error at $t = T$.

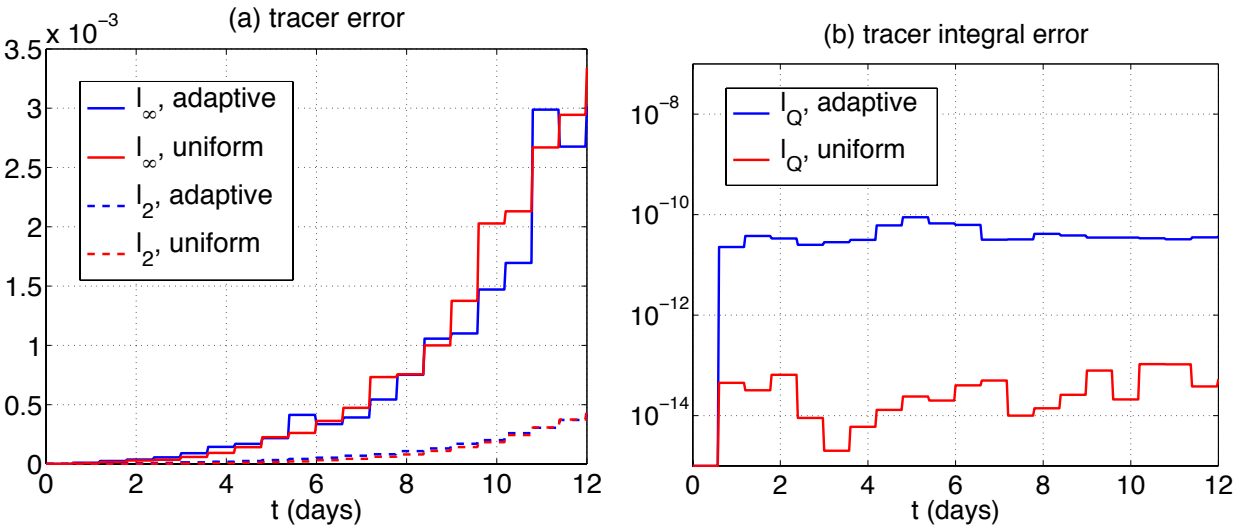

Figure 12: Moving-vortices flow, LPM with indirect remeshing, error as a function of time for $0 \le t \le T$, adaptive panels (blue, $N \le 5174$), uniform panels (red, $N = 20480$), (a) tracer error, l_∞ (solid lines), l_2 (dashed lines), (b) tracer integral error, l_Q.

error reaches saturation and does not significantly increase further in time [figures 4(b), 6(b), 10(b), 12(b)].

- The two LPM remeshing schemes gave similar results for the moving-vortices flow (figure 4), but indirect remeshing was more accurate than direct remeshing for the reversing-deformational flow (figures 5, 6, 8, 9, 10). The key property of indirect remeshing is that it samples tracer values instead of interpolating them; hence, important features such as the tracer maximum and minimum, and tracer correlations, are preserved more accurately (figures 8, 9).

- The LPM computations used the same time step Δt for a wide range of grid spacings $\Delta\lambda$. This was possible because Lagrangian schemes such as LPM are not subject to a CFL time step constraint of the form $\Delta t = O(\Delta\lambda)$. Due to the accuracy of the Runge-Kutta time-stepping scheme used, the temporal discretization error in the particle trajectories was negligible compared to the interpolation error in the tracer due to remeshing.

- For comparable values of the grid spacing $\Delta\lambda$, the computed tracer was more accurate for LPM than for the Lin-Rood scheme (figures 4, 6, 8, 9, 10). However while the tracer integral for LPM converges as the grid spacing is reduced [figures 4(b), 6(b), 10(b)], the Lin-Rood scheme preserves the tracer integral to machine precision.

- An example was presented showing that LPM with adaptive refinement has the potential to efficiently resolve small-scale features in the tracer density (figures 11, 12). Additionally, this example shows that indirect remeshing is not degraded by variable resolution.

Several topics are reserved for future investigation. These include upgrading the accuracy and conservation properties of the interpolation scheme used in remeshing, adapting LPM-i

to simulations with chemistry models and many tracers, and implementing a restart capability for long-time simulations. In addition, the LPM scheme for tracer transport described here can be combined with our particle method for the barotropic vorticity equation on a rotating sphere [12]. In that case the particles also carry vorticity values and the velocity is computed by the Biot-Savart integral. This will enable the study of tracer transport in flows with general vorticity distributions [50]. A further challenging goal is to extend these Lagrangian particle techniques to solve the shallow water equations.

Acknowledgements

This work was supported by a John von Neumann Postdoctoral Fellowship at Sandia National Laboratories, Office of Naval Research grants N00014-12-1-0509 and N00014-14-1-0075, National Science Foundation grant AGS-0723440, and Office of Science, Department of Energy Award No. DE-SC0003990. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2015-6438J.

References

[1] U. Ayachit, The ParaView Guide: A Parallel Visualization Application, Kitware, (2015).
[2] L.A. Barba, A. Leonard, C.B. Allen, Advances in viscous vortex methods - meshless spatial adaptation based on radial basis function interpolation, Int. J. Numer. Meth. Fl. 47 (2005) 387-421.
[3] J. Baumgardner, P. Frederickson, Icosahedral discretization of the two-sphere, SIAM J. Numer. Anal. 22 (1985) 1107-1115.
[4] J.T. Beale, A.J. Majda, High order accurate vortex methods with explicit velocity kernels, J. Comput. Phys. 58 (1985) 188-208.
[5] J. Behrens, K. Dethloff, W. Hiller, A. Rinke, Evolution of small-scale filaments in an adaptive advection model for idealized tracer transport, Mon. Weather Rev. 128 (2000) 2976-2982.
[6] J. Behrens, Adaptive Atmospheric Modeling: Key Techniques in Grid Generation, Data Structures, and Numerical Operations with Applications, Springer, 2006.
[7] J. Behrens, A. Iske, Grid-free adaptive semi-Lagrangian advection using radial basis functions, Comput. Math. Appl. 43 (2002) 319-327.
[8] M. Bergdorf, G.-H. Cottet, P. Koumoutsakos, Multilevel adaptive particle methods for convection-diffusion equations, Multiscale Model. Simul. 4 (2005) 328-357.
[9] M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1989) 64-84.
[10] P. Bochev, D. Ridzal, M. Shashkov, Fast optimization-based conservative remap of scalar fields through aggregate mass transfer, J. Comput. Phys. 246 (2013) 37-57.
[11] P. Bosler, Particle Methods for Geophysical Flow on the Sphere, Ph.D. thesis, University of Michigan (2013).
[12] P. Bosler, L. Wang, C. Jablonowski, R. Krasny, A Lagrangian particle/panel method for the barotropic vorticity equations on a rotating sphere, Fluid Dyn. Res. 46 (2014) 031406.
[13] A.J. Chorin, J.E. Marsden, A Mathematical Introduction to Fluid Mechanics, 3rd Edition, Springer, 1993.
[14] P. Colella, P. R. Woodward, The Piecewise Parabolic Method (PPM) for gas-dynamical simulations, J. Comput. Phys. 54 (1984) 174-201.
[15] C. Cotter, J. Frank, S. Reich, The remapped particle-mesh semi-Lagrangian advection scheme, Q. J. Roy. Meteorol. Soc. 133 (2007) 251-260.
[16] G.-H. Cottet, P. Koumoutsakos, Vortex Methods: Theory and Practice, Cambridge University Press, 2000.

[17] L. Dong, B. Wang, Trajectory-tracking scheme in Lagrangian form for solving linear advection problems: Interface spatial discretization, Mon. Weather Rev. 141 (2013) 324-339.
[18] D. Dritschel, M.H.P Ambaum, A contour-advective semi-Lagrangian numerical algorithm for simulating fine-scale conservative dynamical fields, Q. J. R. Meteorol. Soc. 123 (1997) 1097-1130.
[19] J.K. Dukowicz, J.W. Kodis, Accurate conservative remapping (rezoning) for Arbitrary Lagrangian-Eulerian computations, SIAM J. Sci. Stat. Comput. 8 (1987) 305-321.
[20] H. Feng, L. Kaganovskiy, R. Krasny, Azimuthal instability of a vortex ring computed by a vortex sheet panel method, Fluid Dyn. Res. 41 (2009) 051405.
[21] N. Flyer, G.B. Wright, Transport schemes on a sphere using radial basis functions, J. Comput. Phys. 226 (2007) 1059-1084.
[22] N. Flyer, E. Lehto, Rotational transport on a sphere: Local node refinement with radial basis functions, J. Comput. Phys. 229 (2010) 1954-1969.
[23] B. Fornberg, E. Lehto, Stabilization of RBF-generated finite-difference methods for convective PDEs, J. Comput. Phys. 230, (2011) 2270-2285.
[24] P.T. Haertel, D.A. Randall, Could a pile of slippery sacks behave like an ocean?, Mon. Weather Rev. 130 (2002) 2975-2988.
[25] C.W. Hirt, A.A. Amsden, J.L. Cook, An Arbitrary Lagrangian-Eulerian computing method for all flow speeds, J. Comput. Phys. 14 (1974) 227-253.
[26] E. Kaas, B. Sørensen, P.H. Lauritzen, A.B. Hansen, A hybrid Eulerian-Lagrangian numerical scheme for solving prognostic equations in fluid dynamics, Geosci. Model. Dev. 6 (2013) 2023-2047.
[27] J. Kent, J. Thuburn, N. Wood, Assessing implicit large eddy simulation for two-dimensional flow, Q. J. Roy. Meteor. Soc. 138 (2012) 365-376.
[28] J. Kent, C. Jablonowski, J.P. Whitehead, R.B. Rood, Downscale cascades in tracer transport test cases: An intercomparison of the dynamical cores in Community Atmosphere Model CAM5, Geosci. Model Dev. 5 (2012) 1517-1530.
[29] J. Kent, P.A. Ullrich, C. Jablonowski, Dynamical core model intercomparison project: Tracer transport test cases, Q. J. Roy. Meteor. Soc. 140 (2014) 1279-1293.
[30] P. Koumoutsakos, A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex methods, J. Fluid Mech. 296 (1995) 1-38.
[31] C. Jablonowski, M. Herzog, J.E. Penner, R.C. Oehmke, Q.F. Stout, B. Van Leer, K.G. Powell, Block-structured adaptive grids on the sphere: Advection experiments, Mon. Weather Rev. 134 (2006) 3691-3713.
[32] C. Jablonowski, D.L. Williamson, The pros and cons of diffusion, filters, and fixers in atmospheric general circulation models, in: P.H. Lauritzen, C. Jablonowski, M.A. Taylor, R.D. Nair (Eds.), Numerical Techniques for Global Atmospheric Models, Lecture Notes in Computational Science and Engineering 80 Springer, 2011, Ch. 13., 381-492.
[33] J.-F. Lamarque, L.K. Emmons, P.G. Hess, D.E. Kinnison, S. Tilmes, F. Vitt, C.L. Heald, E.A. Holland, P.H. Lauritzen, J. Neu, J.J. Orlando, P.J. Rasch, G.K. Tyndall, CAM-chem: description and evaluation of interactive atmospheric chemistry in the Community Earth System Model, Geosci. Model Dev. 5 (2012) 369-411.
[34] P.H. Lauritzen, R.D. Nair, P.A. Ullrich, A Conservative Semi-Lagrangian Multi-tracer transport scheme (CSLAM) on the cubed-sphere grid, J. Comput. Phys. 229 (2010) 1401-1424.
[35] P.H. Lauritzen, C. Jablonowski, M.A. Taylor, R.D. Nair (Eds.), Numerical Techniques for Global Atmospheric Models, Lecture Notes in Computational Science and Engineering 80 Springer, 2011.
[36] P.H. Lauritzen, P.A. Ullrich, R.D. Nair, Atmospheric transport schemes: Desirable properties and a semi-Lagrangian view on finite-volume discretizations, in: Numerical Techniques for Global Atmospheric Models, P.H. Lauritzen, C. Jablonowski, M.A. Taylor, R.D. Nair (Eds.), Lecture Notes in Computational Science and Engineering 80, Springer, 2011, Ch. 8., 185-250.
[37] P.H. Lauritzen, W.C. Skamarock, M.J. Prather, M.A. Taylor, A standard test case suite for two-dimensional linear transport on the sphere, Geosci. Model Dev. 5 (2012) 887-901.
[38] P.H. Lauritzen, P.A. Ullrich, C. Jablonowski, P.A. Bosler, D. Calhoun, A.J. Conley, T. Enomoto, L. Dong, S. Dubey, O. Guba, A.B. Hansen, E. Kaas, J. Kent, J.-F. Lamarque, M.J. Prather, D. Reinert, V.V. Shashkin, W.C. Skamarock, B. Sørensen, M.A. Taylor, M.A. Tolstykh, A standard test case suite for two-dimensional linear transport on the sphere: results from a collection of state-of-the-art schemes, Geosci. Model Dev. 7 (2014) 105-145.

[39] P.H. Lauritzen, J. Thuburn, Evaluating advection/transport schemes using scatter plots and numerical mixing diagnostics, Q. J. Roy. Meteor. Soc. 138 (2012) 906-918.
[40] J. Lin, D. Brunner, C. Gerbig, A. Stohl, A. Andreas, P. Webley (Eds.), Lagrangian Modeling of the Atmosphere, American Geophysical Union, 2012.
[41] S.J. Lin, R.B. Rood, Multidimensional flux-form semi-Lagrangian transport schemes, Mon. Weather Rev. 124 (1996) 2046-2070.
[42] S.J. Lin, A "vertically Lagrangian" finite-volume dynamical core for global models, Mon. Weather Rev. 132 (2004) 2293-2307.
[43] A. Magni, G.-H. Cottet, Accurate, non-oscillatory, remeshing schemes for particle methods, J. Comput. Phys. 231 (2012) 152-172.
[44] R.D. Nair, C. Jablonowski, Moving vortices on the sphere: A test case for horizontal advection problems, Mon. Weather Rev. 136 (2008) 699-711.
[45] R.D. Nair, S.J. Thomas, R.D. Loft, A discontinuous Galerkin transport scheme on the cubed sphere, Mon. Weather Rev. 133 (2005) 814-828.
[46] R.D. Nair, P.H. Lauritzen, A class of deformational flow test cases for linear transport problems on the sphere, J. Comput. Phys. 229 (2010) 8868-8887.
[47] NCAR Command Language (Version 6.3.0)(2015). http://dx.doi.org/10.5065/D6WD3XH5
[48] H.O. Nordmark, Rezoning for higher order vortex methods, J. Comput. Phys. 97 (1991) 366-397.
[49] M. Perlman, On the accuracy of vortex methods, J. Comput. Phys. 59 (1985) 200-233.
[50] R.A. Plumb, D.W. Waugh, M.P. Chipperfield, The effects of mixing on tracer relationships in the polar vortices, J. Geophys. Res. 105 (2000) 10047-10062.
[51] W. M. Putnam, S.-J. Lin, Finite-volume transport on various cubed-sphere grids, J. Comput. Phys. 227 (2007) 55-78.
[52] M.J. Prather, X. Zhu, S.E. Strahan, S.D. Steenrod, J.M. Rodriguez, Quantifying errors in trace species transport modeling, PNAS 105 (2008) 19617-19621.
[53] R. Renka, Algorithm 772: STRIPACK: Delaunay triangulation and Voronoi diagram on the surface of a sphere, ACM Trans. Math. Soft. 23 (1997) 416-434.
[54] R. Renka, Algorithm 773: SSRFPACK: Interpolation of scattered data on the surface of a sphere with a surface under tension, ACM Trans. Math. Soft. 23 (1997) 435-442.
[55] R.B. Rood, Numerical advection algorithms and their role in atmospheric transport and chemistry models, Rev. Geophysics 25 (1987) 71-100.
[56] A. Staniforth, J. Côté, Semi-Lagrangian integration schemes for atmospheric models - a review, Mon. Weather Rev. 119 (1991) 2206-2223.
[57] M.A. Taylor, A. Fournier, A compatible and conservative spectral element method on unstructured grids, J. Comput. Phys. 229 (17) (2010) 5879-5895.
[58] J. Thuburn, M. McIntyre, Numerical advection schemes, cross-isentropic random walks, and correlations between chemical species, J. Geophys. Res. 120 (1997) 6775-6797.
[59] D.W. Waugh, R.A. Plumb, Contour advection with surgery: A technique for investigating finescale structure in tracer transport, J. Atmos. Sci. 51 (1994) 530-540.
[60] D. Wee, A.F. Ghoniem, Modified interpolation kernels for treating diffusion and remeshing in vortex methods, J. Comput. Phys. 213 (2006) 239-263.
[61] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, P.N. Swarztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, J. Comput. Phys. 102 (1992) 211-224.
[62] J. Xiao, L. Wang, J. P. Boyd, RBF-vortex methods for the barotropic vorticity equation on a sphere, J. Comput. Phys. 285 (2015) 208-225.
[63] M. Zerroukat, N. Wood, A. Staniforth, SLICE: A Semi-Lagrangian Inherently Conserving and Efficient scheme for transport problems, Q. J. Roy. Meteor. Soc. 128 (2002) 2801-2820.