

Islet: Interpolation semi-Lagrangian element-based transport

Andrew M. Bradley¹, Peter A. Bosler¹, and Oksana Guba¹

¹Sandia National Laboratories, Albuquerque, New Mexico

Correspondence: Andrew M. Bradley (ambradl@sandia.gov)

Abstract. Advection of trace species(~~traeers~~), or tracers, also called tracer transport, in models of the atmosphere and other physical domains is an important and potentially computationally expensive part of a model’s dynamical core(~~dyeore~~). Semi-Lagrangian (SL) advection methods are efficient because they permit a time step much larger than the advective stability limit for explicit Eulerian methods, without requiring the solution of a globally coupled system of equations as implicit Eulerian
5 methods do. Thus, to reduce the computational expense of tracer transport, ~~dyeores~~ dynamical cores often use SL methods to advect ~~passive~~-tracers. The class of interpolation semi-Lagrangian (ISL) methods contains potentially extremely efficient SL methods. We describe a ~~set of ISL bases for element-based transport~~ finite-element ISL transport method that we call the
Islet method, such as for use with atmosphere models discretized using the spectral element (~~SE~~) method. ~~An ISL method that uses the natural polynomial interpolant on Gauss-Legendre-Lobatto (GLL) SE nodes of degree at least three is unstable~~
10 ~~on the test problem of periodic translational flow on a uniform element grid. We derive new alternative bases of up to order of accuracy nine that are stable on this test problem; we call these the Islet bases. Then we describe an atmosphere tracer transport method, the Islet method, that~~ method. The Islet method uses three grids that share an element grid: a dynamics grid supporting, for example, the ~~GLL~~ Gauss-Legendre-Lobatto basis of degree three; a physics parameterizations grid with a configurable number of finite-volume subcells per element; and a tracer grid supporting use of ~~our Islet bases~~ Islet bases, with
15 particular basis again configurable. This method provides extremely accurate tracer transport and excellent diagnostic values in a number of validation problems. ~~We conclude with performance results that use up to 27,600 NVIDIA V100 GPUs on the Summit supercomputer.~~

1 Introduction

~~Trace atmosphere species, or tracers, are an important part of atmosphere models for the study of climate. Thus, tracer transport~~
20 ~~is an important computational subcomponent of an atmosphere dynamical core (dyeore).~~ An atmosphere model has two parts: the dynamical core, which computes resolved fluid flow and resolved thermodynamics (Lauritzen, 2011); and the subgrid parameterizations, which compute unresolved chemistry and physics processes (Stensrud, 2009). In turn, the dynamical core also has two parts: first, the dynamics solver, which solves the equations of fluid motion; second, the tracer transport solver, which advects trace atmosphere species, or tracers, using the air density and flow fields from the dynamics solver.

25 Because of the large number of tracers in climate models, tracer transport can be computationally very expensive. ~~To address this cost, often semi-Lagrangian (SL) methods are used to carry out passive tracer transport. The Lagrangian form of the~~

advective form of the sourceless tracer transport equation is $Dq/Dt = 0$, where $Dq/Dt \equiv \partial q/\partial t + \mathbf{u} \cdot \nabla q$. For example, in the Dept. of Energy's Energy Exascale Earth System Model (E3SM) (E3SM Project, 2018) Atmosphere Model (EAM) version 1 (Golaz et al., 2019), configured with the default 40 tracers, tracer transport takes approximately 75% of the total dynamical core wall clock time and approximately 23% of the total atmosphere model wall clock time on a typical computer cluster (Golaz et al., 2022, Fig. 3). For EAM version 2, we developed a new tracer transport method that is 6.5 to over 8 times faster than in EAM version 1, in the cases of, q is a mixing ratio, and \mathbf{u} is the flow velocity, respectively, low and high workload per computer node (Golaz et al., 2022, Fig. 3). In addition, we developed remap operators to remap data between separate grids for physics parameterizations and dynamics, permitting the physics parameterization computations to run on a coarser grid and thus 1.6 to 2.2 times faster in version 2 than in version 1 (Hannah et al., 2021; Golaz et al., 2022).

There are a number of types of SL methods; see, e.g., Giraldo et al. (2003); Natarajan and Jacobs (2020); Bradley et al. (2019) for a review of these. This article focuses on remap-form interpolation methods. An interpolation SL (ISL) method discretizes the tracer transport equation by $q(x_i, t_2) = q(x_i^*, t_1)$. In this article, we describe methods that extend those we developed for EAM version 2 by providing the discretizations with natural parameters to trade between computational cost and accuracy in tracer transport and physics parameterizations for a given, fixed dynamical core configuration.

1.1 Problem statement

The mass continuity equation for the air density ρ is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (1)$$

where \mathbf{u} is the flow velocity. The tracer transport equation in continuity form and with a source term for a tracer mixing ratio q and corresponding tracer density ρq is

$$\frac{\partial (\rho q)}{\partial t} + \nabla \cdot (\rho q \mathbf{u}) = \rho f, \quad (2)$$

where x_i is an Eulerian arrival grid point and x_i^* is its departure point at $t_1 < t_2$. f is a source term for q has values only at nodal points x_i ; thus, evaluating $q(x_i^*, t_1)$ requires interpolation. A method in remap form directly remaps the tracer field on the Eulerian grid at the previous time step to the Lagrangian grid, in contrast to flux-form methods that integrate over swept regions (Lauritzen et al., 2010; Lee et al., 2016) or characteristic curves (Erath and Nair, 2014). The computational and communication costs of a remap-form method are very nearly independent of time step, whereas a flux-form method's costs grow roughly linearly with time step.

Interpolation is in contrast to cell-integrated methods; these latter integrate the basis of a target (e.g. Lagrangian) element against those of the source. This integration entails substantially greater cost for two reasons: first, larger computational cost due to sphere-to-reference point calculation and interpolant evaluations at many quadrature points; second, larger communication volume because all data from the target element must be made available to a source element. However, in trade for these additional costs, flux-form and exact cell-integrated remap-form methods have. Combining Eqs. (1) and (2) gives the advective

form of the tracer transport equation:

$$\frac{Dq}{Dt} \equiv \frac{\partial q}{\partial t} + \mathbf{u} \cdot \nabla q = f. \quad (3)$$

60 Because of time splitting in the atmosphere model, our focus is often on the sourceless advection equation,

$$\frac{Dq}{Dt} = 0. \quad (4)$$

At position \mathbf{x} and time t_1 , the exact solution of (4) in terms of the solution at another time t_0 is

$$q(\mathbf{x}, t_1) = q(\mathbf{X}(t_0; \mathbf{x}, t_1), t_0). \quad (5)$$

65 ~~$\mathbf{X}(t_0; \mathbf{x}, t_1)$ is the benefits of, first, local mass conservation and, second, stability obtained from the L^2 -projection solution of the ordinary differential equation~~

$$\frac{d}{dt} \mathbf{X}(t; \mathbf{x}, t_1) = \mathbf{u}(\mathbf{X}(t; \mathbf{x}, t_1), t), \quad (6)$$

with the initial condition $\mathbf{X}(t_1; \mathbf{x}, t_1) = \mathbf{x}$. In a semi-Lagrangian method, $\mathbf{X}(t_1; \mathbf{x}, t_1) = \mathbf{x}$ is called the *arrival* point and $\mathbf{X}(t_0; \mathbf{x}, t_1)$ is called the *departure* point. In this article, our focus is on the two-dimensional equations on the sphere. Thus, \mathbf{X} and \mathbf{u} each have two horizontal components and no vertical component. In various settings, the density variable ρ can be, for
70 example, mass per unit area, layer thickness, or layer pressure increment.

In this work, first, we give up local, but not global, mass conservation, and in trade seek maximum computational efficiency, where *computational efficiency* is the ratio of solution accuracy to computational or communication cost, each of these having multiple means of measurement. See Bradley et al. (2019, Sect. 1) for a more detailed discussion of these matters. Second, we seek ISL methods that satisfy a necessary condition for stability. An approximate numerical solution for q of Eq. (4) is said to be
75 *property preserving* if (possibly just a subset of) properties that hold for the exact solution also hold for the approximate one. Equation (5) implies that advection cannot introduce new extrema in the mixing ratio; advection is said to be *shape preserving*. Equation (2) with $f = 0$ implies the global mass is conserved. Although the focus of this article is not the continuity equation, we note that the Lagrangian form of the continuity equation, Eq. (B4) in Appendix B, implies that the total mass in a Lagrangian parcel, which is a parcel of fluid that moves with the flow, is constant. A final property that is a special case of the shape
80 preserving property relates to coupling a solver for Eq. (4) to a dynamics solver: mass-tracer consistency. This property means that if q is constant in space at time t_0 , then it remains constant in space at every other time. In other words, the dynamics solver and transport solver use the same air density. The methods in this article conserve global mass, do not introduce new nodal extrema, and provide mass-tracer consistency when coupled to a dynamics solver.

1.2 Motivation and applications

85 Our primary objective in developing efficient ~~There is a large body of work on~~ tracer transport methods ~~is to provide one for the Dept. of Energy Exascale Earth System Model (E3SM) (E3SM Project, 2018) Atmosphere Model's (EAM) dycore, HOMME~~

(Dennis et al., 2005, 2012). HOMME uses the spectral element (SE) method (SEM) to discretize the equations governing atmospheric flow. *Element-based* methods permit extremely flexible discretization of domains: for example, E3SM’s regionally refined model (RRM) configurations (Tang et al., 2019), in which the atmosphere element grid is refined in a particular region of the earth. Our methods are fully element-based. As a result, researchers have developed a large number of accuracy, property preservation, and other diagnostics for use in comparisons of methods (Lauritzen et al., 2012, 2014; Lauritzen and Thuburn, 2012; Lauritzen

In addition to developing ISL methods that can share a spectral element grid, we develop methods to remap among multiple subelement grids. Our motivation is to enable extremely highly resolved tracer filamentary structure for a fixed dynamics resolution. Another quality of a transport method is important: its computational efficiency. Computational efficiency is some measure of solution accuracy for a given set of computational resources.

Thus, when developing a new transport method, the objective is to obtain high efficiency, as measured by diagnostic values and computational cost, constrained by the need to couple to specific dynamics and physics grids. Our objective in this article is to extend our highly efficient tracer transport method for EAM version 2 by introducing, first, parameters whose values trade between solution accuracy and computational cost and, second, algorithms associated with these parameters.

Our motivation is to support the modeling computationally efficient simulation of strongly tracer-dependent models such as those of aerosols by enabling extremely highly resolved tracer filamentary structure for a fixed dynamics resolution. A recent report from the National Academies of Sciences, Engineering, and Medicine (Field et al., 2021) includes the recommendation to “explore whether global aerosol optical depth (AOD) distribution is significantly affected by plume-scale effects” and asks: “Are nested grids needed to represent plume processes? What spatial resolution is needed to faithfully represent the radiative forcing and impact outcomes?”

Because the Islet method includes algorithms to remap among subelement grids to address this first motivation, it can in addition couple to element-based discretizations other than SE. In this article, we present algorithms that can help to address these questions.

1.2 Related work Overview of the E3SM Atmosphere Model

Baptista and Oliveira (Baptista, 1987; Oliveira and Baptista, 1995) analyzed the stability of a number of SL methods, including ISL. They differentiate between compact and non-compact interpolants. A compact interpolant is one whose support is only the nodes of an element; a non-compact one includes nodes in the interiors of adjacent elements in its support. In this article, we focus on compact interpolants. Let a *natural* interpolant be the unique polynomial interpolant that uses all available nodes in an element as its support, and similarly for a natural basis. Our findings are consistent with theirs: an ISL method using a compact natural interpolant is unstable for elements having degree three and higher. The ISL method on spectral elements, in particular, has been explored in EAM uses the High-Order Method Modeling Environment (HOMME) dynamical core (Dennis et al., 2005, 2012). HOMME uses the spectral element method to discretize the equations of atmospheric flow. HOMME’s grid consists of horizontally unstructured quadrilateral elements extruded as columns in the vertical dimension.

120 Element-based methods permit extremely flexible discretization of domains: for example, E3SM’s regionally refined model (RRM) configurations (Tang et al., 2019), in which the atmosphere element grid is refined in a particular region of the earth.

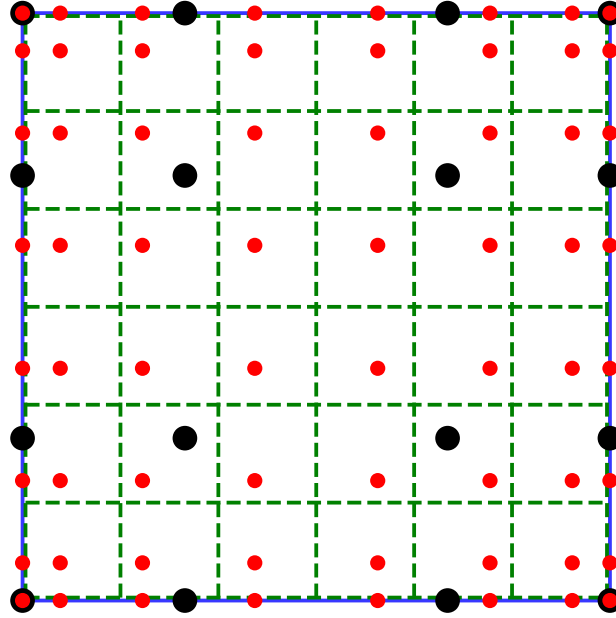


Figure 1. Diagram of subelement grids. One spectral element (blue solid line outlining the full square) with dynamics ($n_p^v = 4$, black large circles), tracer ($n_p^t = 8$, small red circles), and physics ($n_p^f = 6$, green dashed lines) subelement grids.

In the horizontal direction, each quadrilateral element has an $n_p \times n_p$ subgrid. HOMME uses the past (Giraldo, 1998; Giraldo et al., 2003), but these studies did not completely analyze the stability implications of using high-order bases. In Sect. ??, we study the stability of compact interpolants and devise new interpolation bases founded on this analysis. Gauss-Legendre-Lobatto (GLL) spectral element polynomial basis and nodes. The two-dimensional (2D) basis functions are tensor products of one-dimensional (1D) GLL basis functions. n_p is the number of subgrid nodes in a 1D basis function; the degree of the polynomial is $n_p - 1$. Figure 1 shows one spectral element, outlined in blue. The large black dots are the 4×4 tensor product grid of GLL nodes, with $n_p = 4$.

Remapping data among multiple component grids is common in many applications of PDE-based modeling because it is a direct means to permit each component to run in its most efficient configuration. The CFL condition for the advection operator in spectral element fluid dynamics scales inversely with the minimum distance between GLL nodes. In turn, this minimum distance scales roughly quadratically in n_p . For example, most whole-earth, fully-coupled earth system models use different grids for ocean, land, and atmosphere components. Our focus is on using multiple grids within a component to enhance efficiency of each subcomponent model. In atmosphere models, the term nodes for $n_p = 6, 8$, and 12 have minimum distances respectively 2.4, 4.3, and 10.0 times smaller than nodes for $n_p = 4$. Thus, developers of a dynamics solver choose a

value for n_p that is large enough to provide good accuracy and wave representation but small enough to permit a large time step, i.e., one determined by accuracy rather than stability considerations. EAM always sets $n_p^v = 4$, where we use *physics parameterizations* ~~is often used to refer to the collection of microphysics, macrophysics, chemistry, aerosols, radiation, and other sub-element-grid parameterizations, also called subgrid closures (e.g. Stensrud, 2009). These act together as the physics~~ *parameterizations subcomponent, and the dycore is the other subcomponent of v* to refer specifically to the dynamics solver's value of n_p . We refer to the spectral element grid, including the spectral element nodes, as the *dynamics grid*, and the element grid excluding subgrid nodes as the *element grid*.

For efficiency, the EAM version 2 physics and chemistry parameterizations run on a different grid than the dynamics; we call this grid the *physics grid*. To couple efficiently to the dynamics grid, the *atmosphere component*. ~~Recent examples in~~ *atmosphere models of remapping data among multiple subcomponent grids include separate physics parameterizations* physics grid has the same element grid as the dynamics grid but its own subelement grid (Herrington et al., 2019; Hannah et al., 2021). The physics grid divides each horizontal quadrilateral element into a regular $n_f \times n_f$ subgrid of finite-volume quadrilateral subcells. In Fig. 1, the dashed green lines outline the 6×6 physics subgrid, with $n_f = 6$. EAM version 2 uses $n_f = 2$. Compared with EAM version 1, in which the physics and dynamics grids ~~in the atmosphere (Herrington et al., 2019; Hannah et al., 2021)~~ *adaptive mesh refinement (AMR) of* were the same, EAM version 2 has 4/9 as many physics grid points, speeding up the parameterizations part of the model by up to 2.25 times, with a small cost for remapping between dynamics and physics grids. The shared element grid and the element-local remap operators minimize interprocess communication costs.

1.3 Overview of the Islet method

While our tracer transport method is based on spectral elements and, like the physics grid, uses the same element grid as the dynamics grid, it is not restricted by the CFL condition. This important fact motivates making n_p^t , where we use t to refer specifically to the transport solver's value of n_p , a *tracer* (Chen et al., 2021; Semakin and Rastigejev, 2020); ~~and local vertical refinement in physics parameterizations relative to the shared background vertical grid~~, parameter of the tracer transport module and independent of n_p^v , to trade between computational speed (maximum at $n_p^t = n_p^v$) and higher accuracy (increasing with n_p^t). In Fig. 1, the small red circles are GLL nodes for $n_p^t = 8$. We refer to this approach as *tracer transport p-refinement*. In the finite element method, p -refinement means increasing the basis polynomial degree. In the Islet method, we increase n_p^t relative to n_p^v to represent the mixing ratio fields at higher resolution.

In summary, we use one element grid, three subelement grids, and three corresponding parameters: the dynamics grid, n_p^v ; the *Framework for Improvement by Vertical Enhancement (FIVE)* (Yamaguchi et al., 2017; Lee et al., 2020). We use the ideas in Hannah et al. (2021) directly in this article. An important part of remapping fields between grids is conserving mass, not ~~introducing new mixing ratio extrema, and—as a specific case of not introducing new extrema—maintaining a constant mixing ratio. We summarize these requirements as~~ *property preservation*; sometimes *shape preservation* is used to refer to physics grid, n_f ; and the tracer transport grid, n_p^t . For efficiency, EAM version 2 already uses two grids, one with $n_p^t = n_p^v = 4$ and the other with $n_f = 2$. This article considers the case $n_p^t > n_p^v$. Figure 1 shows the three subgrids for one spectral element.

The tracer transport solver couples to the dynamics solver through the flow velocity \mathbf{u} , the air density ρ , and possibly a small subset of tracers, such as specific humidity. It couples to the properties other than mass conservation. The Islet method uses both local-element and global-grid versions of algorithms described and analyzed in Bradley et al. (2019); see that article for a detailed discussion of the problem of property preservation.

This article focuses on the horizontal dimensions or, more generally, two-dimensional Lagrangian levels in a three-dimensional discretization. Future work using methods similar to those in Sect. ?? will address the vertical dimension. For example, it is necessary to increase simultaneously both the horizontal and vertical resolutions of a plume to capture and maintain its structure (Eastham and Jacob, 2017) parameterizations through the tracers. Coupling requires remapping a field between grids. It is natural to give responsibility for these remap procedures to the tracer transport solver module, as the remap operators share many of the same requirements and computations as the transport solver. As in the case of remapping quantities between the physics grid and the dynamics grid in EAM version 2, the shared element grid and element-local remap procedures minimize communication when remapping among the three grids. For brevity, we refer to this overall spectral element, three-grid scheme as the *Islet method*.

1.4 Outline

The rest of this article is structured as follows. Section ?? describes a necessary condition for stability of an ISL transport method on a test problem. In Sect. ??, we combine this necessary condition with two quality heuristics and a search procedure to derive element basis sets, the *Islet bases*, for ISL element-based transport. In The Islet method depends on a number of lower-level algorithms. Section 2 and Appendix A describe these; mathematical details can be skipped on a first reading. Section 3 describes each step of the Islet method over the course of one time step. Descriptions are compact, relying on the details in Sect. ??, we use these basis sets as part of the overall *Islet method* to couple accurate tracer transport to a spectral element dynamical grid and a finite-volume physics parameterization grid². Section 4 describes numerical validation problems and presents results for the Islet method. Section ?? shows performance data on the GPU-powered supercomputer Summit and presents the results of a number of validation problems. Finally, Sect. 5 concludes.

2 Stability Overview of algorithms

A necessary and sufficient condition for stability of a linear method is uniform power boundedness (Strikwerda, 2004): the norm of

This section describes the details of low-level algorithms on which the Islet method depends, associated concepts, and related work for context. Our objective is to isolate the notation for, and details of, these low-level algorithms to this section, then focus on the higher-level details of the method's space-time operator raised to any integer power $n \geq 0$, and independent of (i.e. "uniform in") discretization parameters, is bounded by a constant. Islet method in Sect. 3 without lower-level clutter. Section 2.1 discusses semi-Lagrangian transport, the finite-element interpolation semi-Lagrangian method in particular, and the bases that the Islet method uses. Section 2.1 describes our approach to solving the property preservation problem. Section

2.1 describes algorithms to remap data between grids. Section 2.2 describes the spectral element direct stiffness summation operator.

Consider the pure advection *test problem* of periodic uniform translation on a uniform element grid with no source terms. A necessary condition for stability of a space-time operator for this problem is that the maximum amplitude of an eigenvalue of the space-time operator is at most one. A discrete space-time operator for pure advection in general has at least one eigenvalue of magnitude one.

2.1 Semi-Lagrangian transport

The Islet method uses semi-Lagrangian (SL) transport. In an SL method, following Eqs. (5) and (6), each Eulerian arrival grid point at time t_1 is advected backward or forward in time to its generally off-grid departure point at time t_0 . The resulting Lagrangian departure grid then samples the field at time t_0 , and by some means uses these data to construct the advected field at time t_1 .

In the problem of tracer transport, for which the flow velocity u is provided, SL methods can take much longer time steps than Eulerian methods. Data dependencies are local to a small domain of dependence; thus, unlike an implicit method, SL tracer transport does not require the solution of a global linear equation. This long time step requires many fewer interprocess communication rounds, and sometimes fewer floating point operators, per unit of simulation time than Eulerian methods, making SL methods substantially more efficient than Eulerian methods for tracer transport.

For the class of discretizations we develop in this article, we do not attempt to prove satisfaction of this necessary condition analytically. For doing so is equivalent to finding a bound of one, analytically, on the magnitudes of eigenvalues of an infinite set of $d \times d$ matrices for d as large as 12, where the matrices have at least one eigenvalue with magnitude one. We are not aware of a means to do this. Thus, our approach is purely numerical: we describe a standard method to compute eigenvalues numerically efficiently, and then we apply this method to numerically test methods at a large but finite number of points in parameter space and with finite precision results. Transport methods often are the composition of a number of operators. Two key operators are the linear advection operator and the nonlinear property preservation operator; we describe ours in Sects. 2.0.1 and 2.1, respectively.

Let λ_{\max} denote the maximum eigenvalue magnitude we find in our search. In infinite precision, the necessary condition implies $\lambda_{\max} = 1$ for the test problem. In finite precision, we expect error in the computation and require $\lambda_{\max} \leq 1 + \epsilon$, with ϵ a tolerance near machine precision. We say that a method is *test-problem stable* (t.p.s.) if this numerical bound holds for the method applied to the test problem. For brevity, we also say that a basis is or is not t.p.s., by which we mean that the ISL method that uses the basis is or is not t.p.s.

2.1.1 Types of SL methods

For a broad survey of atmosphere tracer transport methods, including a large number of Lagrangian and semi-Lagrangian methods, we refer the reader to Lauritzen et al. (2014, Sect. 2). In addition, Giraldo et al. (2003); Natarajan and Jacobs (2020); Bradley et al. review types of SL methods. This article focuses on remap-form interpolation methods.

2.2 Instability in the classical cubic ISL method

We neither seek nor expect to find methods that satisfy the necessary condition in more general settings. For already classical methods do not. In this subsection, we provide two examples in which the classical cubic ISL method is not stable. To find each example, we construct a parameterized problem and discretization, then run a randomized search over parameter values to find non-t.p.s. instances. Finally, to simplify reproducibility, we round the parameter values to not more than five digits while still obtaining a maximum eigenvalue magnitude substantially greater than one. A method in *remap form* directly remaps the tracer field on the Eulerian grid at the previous time step to the Lagrangian grid, in contrast to flux-form methods that integrate over swept regions (Lauritzen et al., 2010; Lee et al., 2016) or characteristic curves (Erath and Nair, 2014). The computational and communication costs of a remap-form method are very nearly independent of time step, whereas a flux-form method's cost grows roughly linearly with the time step.

First, consider the classical cubic ISL method on a nonuniform one-dimensional grid for the problem of periodic uniform translational flow. Let the periodic grid have the five unique points $\{0, 0.11242, 0.44817, 0.78392, 0.88737, 1\}$, where 0 is identified with 1 by periodicity. The cubic interpolant uses the four source points nearest a target departure point as its support. Let the flow speed and time step be such that in each time step, the field translates by distance 0.33575. The space-time operator for this configuration has maximum eigenvalue magnitude exceeding $1 + 10^{-3}$, violating the necessary condition for stability. An interpolation SL (ISL) method directly discretizes the tracer transport equation, Eq. (5): $q(x_i, t_1) = q(\bar{x}_i, t_0)$, where x_i is an Eulerian arrival grid point and \bar{x}_i is its departure, generally off-grid, point at $t_0 < t_1$. q has exact values only at grid points x_i ; thus, evaluating $q(\bar{x}_i, t_0)$ requires interpolation.

Second, consider again the classical cubic ISL method. Interpolation is in contrast to *exactly cell-integrated* methods, which accurately integrate the basis of a target (e.g. Lagrangian) element against those of the source; see, e.g., but now on a doubly periodic, two-dimensional grid $[0, 1]^2$, where again 0 is identified with 1 by periodicity. The grid consists of uniformly sized rectangles. The flow velocity, while still nondivergent and constant in time, is shear in space: $s \equiv 1 + \cos(2\pi(0.342 + x - y))$, $(u, v) = (s, s)$. With 15 unique grid points in the x direction and 13 in the y and time step 0.2761, the space-time operator has maximum eigenvalue magnitude exceeding $1 + 10^{-2}$. Bosler et al. (2019). (In some cases, an inaccurate cell-integrated method can be interpreted as an interpolation method; see Appendix B for an example.) Exactly cell-integrated methods have substantially greater cost than interpolation methods for three reasons.

Figure ??, described in Sect. ??, plots $\log_{10}(\lambda_{\max} - 1)$ as a function of step size for a set of bases. We need to proceed further in this article before we can describe this figure in detail. However, one qualitative fact is apparent now. In one basis, with corresponding line pattern red with small circles, λ_{\max} is substantially above 1 for almost all step sizes. In the other two, green with \times markers and black with large circle marker, $\lambda_{\max} - 1$ is at machine precision for almost all values, with one or a small number of spikes to larger values. We refer to this second pattern of λ_{\max} values as *spiky*. In our two examples, the classical cubic First, to obtain smoothness in the integrand, integration is over facets computed by geometric intersection of a target element against source elements; intersection calculations are not needed in interpolation methods. Typically, to minimize computational geometry complexity, departure cell edges are approximated by great arcs rather than flow-distorted curves,

limiting the method to second-order accuracy; however, Ullrich et al. (2013) describe a higher-order edge reconstruction that yields a third-order accurate advection method. In contrast, achieving arbitrarily high order in an ISL method's λ_{\max} is spiky with respect to step size and other parameters. We speculate all high-order ISL methods have at least a spiky λ_{\max} pattern when applied to some problems, and practical high-order ISL methods must have no worse than a spiky pattern or else be unstable in practice. Sect. ?? compares the consequences of the two λ_{\max} patterns on validation problems. linear advection operator does not entail any additional complexity.

2.2 Notation and definitions

Vectors and matrices are written in boldface; continuum fields are not. Thus, \mathbf{f} could be the nodal values that, combined with a basis, represent the continuum field f . Indexing starts at 0. To minimize nested subscripting, in some cases we denote entries of a vector parenthetically, and we use colon notation to indicate an index list. For example, $\mathbf{x}(i)$ is element i of \mathbf{x} , and $\mathbf{x}(i:i+2)$ produces the 3-element vector containing elements i through $i+2$ inclusive. The list $n-1:-1:0$ decrements the index; thus, $\mathbf{x}(n-1:-1:0)$ is a vector with elements in the opposite order of those in \mathbf{x} . It is sometimes useful to annotate a matrix or vector with its size; for example, each of the matrix of zeros $\mathbf{0}^{m \times n}$ and the identity matrix $\mathbf{I}^{m \times n}$ has size $m \times n$, and \mathbf{x}^n is an n -vector. Often in equations we annotate only the first instance of the vector or matrix to provide useful data while minimizing clutter. Let \mathbf{e}_i be column i of the identity matrix. Second, accurate integration has a larger computational cost because it requires sphere-to-reference point calculation and interpolant evaluations at many quadrature points.

We refer to various grids. An *element grid* is distinguished from the *point* or *subelement* grid. The point grid is derived from the element grid by populating each element with either subelement grid points associated with the method's basis or finite-volume subcells. When *grid* is used without qualification, we mean the point grid. Third, an exactly cell-integrated method requires a larger communication volume because all data from a source element are used to integrate against each target basis function.

2.2 Maximum eigenvalue magnitude

The ISL space-time operator for an element-based method applied to the test problem has the following structure. Let a matrix $\mathbf{B}^{d \times (d+1)} \equiv (\bar{\mathbf{B}}^{d \times d} \mathbf{b}^{d \times 1})$. Let \mathbf{C}_k be the permutation matrix that circularly shifts its left operand k columns to the right and its right operand k rows up. Let

$$\mathbf{A} \equiv \mathbf{C}_{-r'} \left[\left(\mathbf{I}^{N \times N} \otimes \bar{\mathbf{B}} \right) + \left(\mathbf{I}^{N \times N} \otimes (\mathbf{b} \mathbf{e}_{d-1}^T) \right) \mathbf{C}_1 \right],$$

where \otimes denotes the Kronecker product, T the matrix transpose, and N the number of elements. In trade for these additional costs, exactly cell-integrated methods are locally mass conserving, and the fact that they are L^2 projections can be used to prove stability. Local mass conservation means that one can identify numerical, possibly Lagrangian, fluid parcels on the grid that have constant tracer mass. *Local* is in contrast to *global* mass conservation; the latter means that the mass of the tracer fluid is conserved over the whole domain but not necessarily in any identifiable parcels smaller than the domain. Although an

exactly cell-integrated method is locally mass conserving, coupling it to a dynamics solver still generally requires additional measures to obtain mass-tracer consistency.

This structure arises as follows. Consider a continuous discretization using a nodal n_p -basis, $n_p = d + 1$, with n_p the number of nodes. The grid has N elements. Each row of the space-time matrix corresponds to a target node. The target node's departure point is in a source element, and the source element's n_p nodes provide the interpolant's support for that target node. Because the test problem has uniform flow and

2.1.1 Spectral element ISL transport

Our objective in this work is to use the design freedom provided by giving up local, but not global, mass conservation to maximize computational efficiency. We develop an ISL method that uses the *Islet bases*, described in Bradley (2022) and Bradley et al. (2021, Sect. 2 and 3) and summarized in Appendix A, to satisfy a necessary condition for stability.

Figure 2(a) illustrates the linear advection step in the Islet method. First, the grid is uniform, each source element contains $n_p - 1$ target node departure points. The upper-left element is advected backward in time from time t_1 to time t_0 , using velocity data on the $n_p^v = 4$ dynamics grid to advect each dynamics grid point. Second, the $n_p^v = 4$ isoparametric map interpolates the locations of the Lagrangian $n_p^t = 6$ tracer grid points from the Lagrangian $n_p^v = 4$ dynamics grid points. Third, the tracer data at time t_0 in the bottom-right Eulerian element is used to interpolate the tracer mixing ratio at t_0 to each green-starred Lagrangian point. Interpolation uses the 2D tensor-product $n_p^t = 6$ nodal basis. Fourth, these values are then the values in the upper-left Eulerian element at time t_1 .

Figure 2(b) shows two bases, the GLL basis, called the *natural* basis when a distinguishing name is needed, and the Islet GLL basis. The ISL method using the natural GLL basis is unstable for the test problem of uniform flow on a uniform grid, but the ISL method using the Islet GLL basis is stable for this problem (Bradley et al., 2021; Bradley, 2022). In this article, all of the algorithms accommodate either basis type, and we use the Islet GLL bases to obtain stability; see Appendix A for details.

Each basis is a *nodal* basis: a basis function has value 1 at one node and 0 at every other node. Thus, for each source element, there is a $d \times (d + 1)$ block \mathbf{B} in the matrix. Again because the test problem has uniform flow and the grid is uniform, this block is the same for each source element. Translation distance determines the number of rows r' these blocks are circularly shifted. The final column of each block, b , overlaps the first column of the next because elements share element-boundary nodes.

We seek the maximum eigenvalue magnitude of the matrix \mathbf{A} . Diagonalization of a matrix having this form can be understood in a number of ways: the discrete Fourier transform diagonalizes a circulant matrix, and block diagonalization of a block circulant matrix follows with some algebra (e.g. Vichnevetsky and Bowles, 1982; Idelsohn et al., 1995); by Bloch-wave decomposition (e.g. Cohen, 2001; Ainsworth, 2004); and by consideration of the problem's symmetries. The result is that the spectrum of \mathbf{A} can be obtained by solving eigenvalue problems associated with \mathbf{B} . For fixed N , one must solve N independent eigenvalue problems of size $d \times d$. Thus, they can be solved in parallel, and each has cost that is a function of just d rather than Nd .

Appendix ?? provides details. Each basis function is associated with a node. For example, in Fig. 2(b), the blue basis function is associated with the third node of six. The basis is symmetric; basis function $k \in \{0, \dots, n_p - 1\}$ is the mirror image of basis function $n_p - k - 1$. Thus, the blue and cyan functions are mirror images around reference coordinate 0.

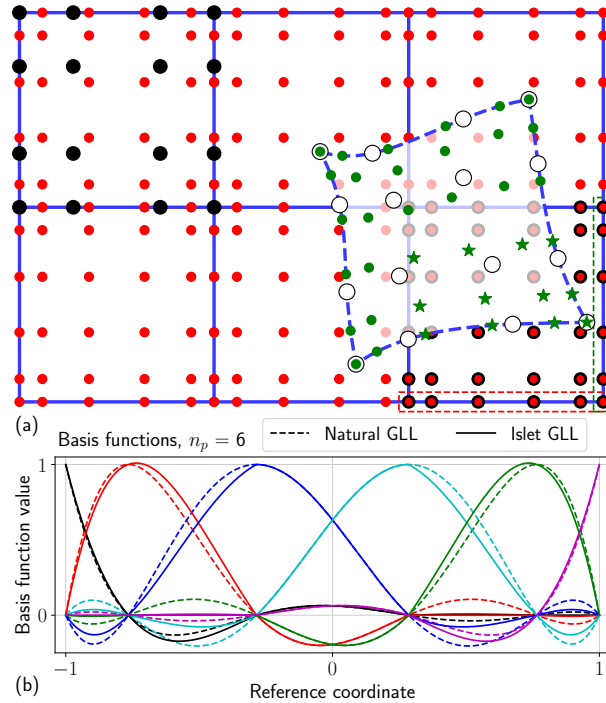


Figure 2. Diagram of the linear advection operation in a finite-element interpolation semi-Lagrangian time step. (a) Six spectral elements (blue solid lines) with an $n_p^t = 6$ tracer grid (red small circles). The black large circles in the upper-left element show the $n_p^v = 4$ dynamics grid. The upper-left Eulerian arrival element is advected backward in time to the right and down, resulting in the Lagrangian departure element (blue dashed outline, open-circle dynamics grid points, green tracer grid points). The domain of dependence of each green-starred tracer grid point is the bottom-right Eulerian element, with interpolation grid points (red small circles) associated with this domain of dependence outlined in black. (b) The six 1D $n_p^t = 6$ element basis functions, a different color for each, for the natural GLL (dashed curves) and Islet GLL (solid curves) bases. In the bottom-right Eulerian element, a set of 1D basis functions is conceptually associated with each of the red horizontal and green vertical rectangles outlining an element side, and a 2D basis function over the element is a tensor product of one 1D basis function from each set.

2.2 Test-problem stability

The free parameter in the test problem is the scalar flow speed times time step, or simply \mathbf{B} . There are two conventional ways to enumerate finite element nodal degrees of freedom and basis functions, where for a scalar field, one degree of freedom is associated with each basis function. First, in this article, by a basis we always mean an *translation-distance*. A block \mathbf{B} corresponds to one particular flow speed. Let \mathcal{N} be the collection of data describing the basis according to which \mathbf{B} is formed. To approximate $\lambda_{\max}(\mathcal{N})$, the maximum eigenvalue magnitude corresponding to basis \mathcal{N} , we must discretize not only the search over values of number of elements N , but also the translation distance. *element* basis unless stated otherwise. A basis

340 function in an element basis set has non-zero values only within an element. As a result, nodes in the interior of an element have single-valued field values, while nodes at an element boundary may have multi-valued field values.

We can use two symmetries to reduce the translation distance set. First, on a uniform element grid having element size one, any translation $\Delta x + n$ for integer n yields the same maximum eigenvalue magnitude as simply Δx . Integer increments to Δx correspond to row shifts of the space-time operator, to which the maximum eigenvalue magnitude is invariant. Second, basis symmetry implies that only $\Delta x \in (0, 1/2]$ needs to be searched, for λ_{\max} is the same for translations $\pm \Delta x$, and then it is the same for $-\Delta x + 1$ by the first. Second, the viewpoint of *global* continuous degrees of freedom and *global* basis functions always assigns only one degree of freedom of the field to each grid point. The basis function associated with a grid point in the interior of an element is the same as in the element basis. A grid point on the element boundary has a basis function that spans all the elements that share that grid point. This global basis function is the sum of the element basis functions associated with that grid point.

350 In summary, computing an approximation to $\lambda_{\max}(\mathcal{N})$ requires solving eigenvalue problems for a discrete subset of the two-dimensional space $\Delta x \in (0, 1/2]$, $\theta \in [0, 2\pi)$, where, as explained in Appendix ??, θ is the parameter in the eigenvalue computation corresponding to N . If in the course of this search it is found that $\lambda_{\max}(\mathcal{N}) > 1 + \varepsilon$, with ε a tolerance near machine precision, then \mathcal{N} is determined not to be t.p.s.

355 If a basis is t.p.s. on the one-dimensional test problem and has tensor-product structure, then the same holds on the multidimensional test problem, too, because the multidimensional test problem decouples by dimension. Our interpolants have tensor-product structure. In the case of continuous finite element methods, the two viewpoints are equivalent, and convenience determines which viewpoint is adopted. In the case of a discontinuous finite element method, only the element viewpoint is possible.

360 3 The Islet bases

The natural basis at Gauss-Lobatto-Legendre (GLL) nodes — for brevity, subsequently the *natural GLL basis* — is not t.p.s. for $n_p \geq 4$, as we illustrate in Sects. ?? and ??. In this section, we design compact modified bases that are t.p.s.

2.1 Definitions

The polynomial interpolant through values \mathbf{y}^n at points \mathbf{x}^n , evaluated at x , in Lagrange form is

$$365 \quad \mathcal{L}(x; \mathbf{y}^n, \mathbf{x}^n) \equiv \sum_{i=0}^{n-1} \mathbf{y}(i) \prod_{j=0, j \neq i}^{n-1} \frac{x - \mathbf{x}^n(j)}{\mathbf{x}^n(i) - \mathbf{x}^n(j)}.$$

If the second argument to \mathcal{L} is a matrix $\mathbf{Y}^{n \times k}$ rather than a vector \mathbf{y}^n , then the output is a vector whose element j corresponds to column j of $\mathbf{Y}^{n \times k}$. If the first argument is a vector \mathbf{z}^m , then the output is an $m \times k$ matrix whose i th row corresponds to $\mathbf{z}^m(i)$ and j th column corresponds to column j of $\mathbf{Y}^{n \times k}$. \mathcal{L} provides a basis for degree- d polynomials. These are supported by $n = d + 1$ points, each an element in the n -vector \mathbf{x}^n . The natural basis functions associated with \mathbf{x}^n are $\mathcal{L}(x; \mathbf{I}^{n \times n}, \mathbf{x}^n)$.

370 In this work we take \mathbf{x} to be points in the reference domain $[-1, 1]$. Often $\mathbf{x}^n = \mathbf{x}_G^n$, the GLL points, or a subset of these. *Region r of the domain $[-1, 1]$ is $[\mathbf{x}^n(r), \mathbf{x}^n(r+1)]$. A middle region is region $r = n/2 - 1$ if* The Islet method accommodates both continuous and discontinuous variants; in this article, we present only the continuous variant. We always take the element viewpoint, regardless of the continuity of a field. Thus, for a scalar field, there is one degree of freedom associated with each grid point in an element, and thus n is even; if n is odd, then there is no middle
 375 region.

2.1 Interpolants

The coefficients of the ISL space-time operator correspond to interpolation at departure points. We consider piecewise-polynomial interpolants having the following constraints. In Appendix ?? we state these constraints precisely; here, we summarize them sufficiently to motivate the subsets we choose to study in this article.

- 380 1. Given a departure point \mathbf{x} , the interpolant is a linear operator in the function values \mathbf{y} at basis points \mathbf{x} .
2. Within each region, the interpolant is a polynomial elements. Index i is associated with grid point location \mathbf{x}_i . Again, to be clear, for a grid point at an element edge, multiple index values are associated with the same location \mathbf{x} . Some steps in the Islet method introduce discontinuities in the field across elements. Section 2.2 describes the operator that restores continuity to a discontinuous field.
- 385 3. The interpolant must recover a degree- p polynomial with specified $p = s - 1$ and $2 \leq s \leq n$. We call this the *order constraint*.
4. The interpolant must interpolate nodes $r, r + 1$; that is, it must have values $\mathbf{y}(r)$

2.0.1 Definitions and notation

The tracer mixing ratio degrees of freedom are written q_i ; the air density, ρ_i . The vector of all scalar values is bold-faced, e.g.,
 390 $\mathbf{y}(r+1)$, respectively, at these nodes. We call this the *region interpolation constraint*. One consequence is that the resulting interpolant is continuous.

5. The nodal basis function associated with node i of n must have mirror symmetry around 0 with the basis function associated with node $n - i - 1$.

2.1 Nodal subset bases

395 In this article, with one exception, we limit our attention to two nested proper finite subsets of the infinite set of all basis sets that satisfy constraints 1–5: *nodal subset bases* (n.s. bases) and, a proper subset of these, \mathbf{q} . A subset of these, with the index set given by \mathcal{E} , is written $\mathbf{q}(\mathcal{E})$. When needed, a superscript letter indicates on which grid the field is approximated: t , tracer; offset nodal subset bases (o.n.s. bases), each with $\mathbf{x}^{n_p} = \mathbf{x}_G^{n_p} \mathbf{v}$, dynamics; f , physics.

A nodal subset n_p -basis \mathcal{N} consists of two data sets. First, it has $n_p - 1$ sets of support nodes, $\mathcal{I}_r = \mathcal{N}(r)$, one for each
 400 region $r \in \{0, \dots, n_p - 2\}$, with $(n_p^{\text{sub}})_r = |\mathcal{I}_r|$ the number of nodes in region r 's support. For brevity, we call a set of support

nodes \mathcal{I}_r a support. The region interpolation constraint requires $r, r+1 \in \mathcal{I}_r$. Second, it has the points $\mathbf{x}_C^{n_p} = \mathcal{N}_x$. Arithmetic between vectors applies entry-wise. For two vectors \mathbf{x} and \mathbf{y} , we write $\mathbf{x} \odot \mathbf{y}$ for element-wise multiplication and $\mathbf{x} \oslash \mathbf{y}$ for element-wise division.

We write the vector of reference domain coordinates of the support nodes for a region, the region's support points, as $\mathbf{x}^{n_p}(\mathcal{I}_r)$. Given these data, the basis functions take values $\mathcal{L}(x; \mathbf{I}^{n_p \times n_p}(\mathcal{I}_r, \cdot), \mathbf{x}^{n_p}(\mathcal{I}_r))$ in region r , i.e. $x \in [\mathbf{x}^{n_p}(r), \mathbf{x}^{n_p}(r+1)]$. A software implementation can use the barycentric form of the polynomial interpolant by first precomputing coefficients for each region.

Offset nodal subset bases in addition have consecutive indices in each \mathcal{I}_r , and thus a region's support can be defined by its number of support nodes and an offset index giving the smallest index node in the support.

So far, these definitions satisfy the first four constraints. Using the symmetry constraint, basis data for regions $r \geq \lfloor n_p/2 \rfloor$ are constructed by symmetry of the basis set from regions $r < \lfloor n_p/2 \rfloor$. In addition, when n_p is even, need to define a map between a reference element with coordinates $\mathbf{r} \in [-1, 1]^2$ and element e on the manifold with coordinates \mathbf{x} , e.g., the sphere. Let this map be $\mathbf{x} = m_e(\mathbf{r})$ and its inverse be $\mathbf{r} = m_e^{-1}(\mathbf{x})$. Additionally, let $e = E(\mathbf{x})$ return the element index for the symmetry constraint implies that the middle region support, \mathcal{I}_r with $r = n_p/2 - 1$, must correspond to support points that are symmetric about $\mathbf{x} = \mathbf{0}$ element that contains e . In this work, if \mathbf{x} is on the boundary of multiple elements, $E(\mathbf{x})$ can be any of the corresponding element indices without affecting the solution.

Basis functions for the Islet $n_p = 6$ GLL nodal subset basis listed in Table A1. Each curve's color corresponds to a basis function. Each line pattern corresponds to a basis type, as listed in the legend. The green span shows region 1. The red arrows point to the nodes in the support of region 1; the red \times is beneath the one node not in region 1's support. Similarly, we need to map between (element-)local and global indices. Local indices are in the set $\{0, \dots, (n_p^g)^2 - 1\}$, where $g \in \{v, t\}$. Global index i is in element $\text{elm}(i; g) \equiv \lfloor i / (n_p^g)^2 \rfloor$. The corresponding local index is $\text{lcl}(i; g) \equiv i \bmod (n_p^g)^2$. Given the local index j and element index e , the global index is $\text{glb}(j; g) \equiv (n_p^g)^2 e + j$. Let $\mathcal{E}(e; g)$ be the set of global indices for the grid points in element e . We omit g when it is not necessary to specify a grid. For convenience, let $\mathbf{q}_e \equiv \mathbf{q}(\mathcal{E}(e))$.

n_p OOA n_p^{sub} Supports 4 2 see text see text 5 2 {3, 4} offsets {0, 0} 6 4 {5, 5, 6} nodal subsets {{0, 1, 2, 3, 4}, {0, 1, 2, 3, 5}, {0, 1, 2, 3, 4, 5}} 7 4 {5, 5, 6} offsets {0, 0, 0} 8 5 {6, 6, 7, 6} offsets {0, 0, 0, 1} 9 6 {7, 8, 8, 7} nodal subsets {{0, 1, 2, 3, 4, 5, 8}, {0, 1, 2, 3, 4, 5, 7, 8}, {0, 1, 2, 3, 4, 5, 6, 8}, {1, 2, 3, 4, 5, 6, 7}} 10 6 {7, 7, 7, 8, 8} offsets {0, 0, 0, 0, 1} 11 7 {8, 9, 8, 9, 8} offsets {0, 0, 0, 0, 1} 12 8 9, 9, 10, 10, 9, 10 offsets {0, 0, 0, 0, 1, 1} 13 9 10, 10, 10, 10, 11, 10 offsets {0, 0, 0, 0, 0, 1} Islet GLL nodal subset bases. Each row provides a formula for the row's n_p value. Columns are n_p , order of accuracy (OOA), the support sizes n_p^{sub} for each region ordered left to middle, and the supports. For offset nodal subset bases, supports are given by offsets. For general nodal subset bases, supports are given by nodal subsets, again ordered from left region to middle. The case $n_p = 4$ is described in Sect. ?? In all cases, the support points are GLL points. Element basis functions and reference-coordinate nodes are indexed locally. Thus, global grid point i corresponds to the 2D tensor-product element basis function $\phi_{\text{lcl}(i; g)}$ and reference-coordinate node $\mathbf{r}_{\text{lcl}(i; g)}$.

Figure ?? shows an example of a n.s. basis for $n_p = 6$. Each color corresponds to a basis function. Dashed curves are the natural GLL basis functions. Solid curves are Occasionally we write the tensor product in the n.s. basis functions. This basis

corresponds to the entry for $n_p = 6$ in Table A1. There are five regions. The green span shows region 1. The red arrows point to the nodes in the support of region 1; the red \times is beneath the one node not in region 1's support. Referring now to Table A1, regions 0 to 2 have $(n_p^{\text{sub}})_r$ values, in region order, of $\{5, 5, 6\}$, implying the basis provides order of accuracy 4. Region 1 has support $\mathcal{I}_1 = \{0, 1, 2, 3, 5\}$. Because this support omits node 4 and includes nodes 3 and 5, this basis is not an o.n. s. basis; \mathcal{I}_1 does not contain fully consecutive nodes. By symmetry, region 3 has support $\mathcal{I}_3 = \{0, 2, 3, 4, 5\}$. Region 2 has support $\mathcal{I}_2 = \{0, 1, 2, 3, 4, 5\}$ and thus uses all nodes. The support nodes for region 2 are symmetric around the element center, thus satisfying the symmetry constraint. Consider now the basis function values as they pass through region 1. Each basis function can be identified with the node at which it takes value 1. Basis functions 0, 1, 2, 3, and 5 all have nonzero values in region 1 because these nodes are in \mathcal{I}_1 . Basis function 4 has function values 0 throughout region 1 because node 4 is not in \mathcal{I}_1 . As an example of one basis function, basis function 2 has the values $\mathcal{L}(x; (0 \ 0 \ 1 \ 0 \ 0)^T, \mathbf{x}_G^6([0, 1, 2, 3, 5]))$ in region 1.

Let $r = \text{region}(x; \mathbf{x}^n)$ be the region containing x . If $x = \mathbf{x}^n(r)$, $\text{region}(x; \mathbf{x}^n)$ can return r or $r - 1$; the value does not affect calculations because the interpolant is continuous. Let n_p^{submin} be the minimum value of $|\mathcal{I}_r|$ over r . When present, a subscript \mathcal{N}_{n_p} gives the total number of unique support nodes for the basis, and a superscript $\mathcal{N}_{n_p}^{\text{submin}}$ gives the minimum value of $|\mathcal{N}(r)|$ in the basis. Thus, for example, \mathcal{N}_4^4 denotes the natural GLL basis. 2D basis explicitly. Let a 1D element basis function set be
 $\psi_j, j \in 0, \dots, n_p - 1$. Let $\mathbf{r} \equiv (r_1, r_2)$. Then $\phi_i(\mathbf{r}) \equiv \psi_{\lfloor i/n_p \rfloor}(r_1) \psi_{i \bmod n_p}(r_2)$. Figure 2(b) shows the $n_p = 6$ basis functions ψ_j .

2.1 Accuracy heuristic

Let $p_d(x)$ be The function normalize(\mathbf{x}) $\equiv \mathbf{x}/\|\mathbf{x}\|_2$.

Finally, we write the particular map $\mathbf{x} = m_e(\mathbf{r})$ we use. The map uses just the corner nodes of an element; it is the isoparametric map for a degree- d polynomial interpolant defined by support points $\mathbf{x}^n, n = d + 1$, in $[-1, 1]$ and including the endpoints, and nodal values $f(\mathbf{x}^n)$, with $f(x) \in C^{d+1}$ for $x \in [-1, 1]$. A standard result (e.g. Lozier, 2003, Sect. 3.3(i)) is that the approximation error is bounded by

$$|p_d(x) - f(x)| \leq \max_{\xi \in [-1, 1]} \left| \frac{f^{(d+1)}(\xi)}{(d+1)!} \prod_{i=0}^d (x - \mathbf{x}^n(i)) \right|.$$

Based on this inequality, we define the following accuracy heuristics for bilinear basis. In element e , let $\text{corner}_e(j; \mathbf{v})$ be the
 l_1 global index of corner j of the element, here referenced to the dynamics grid for concreteness. Let $\phi_j^{\text{bilin}}(\mathbf{r})$ be the bilinear basis functions, i.e., l_2 , and l_∞ norms. Let

$$e(x; \mathcal{N}) \equiv \left| \frac{1}{n!} \prod_{i=0}^{n-1} (x - \mathbf{x}(\mathcal{I}_r(i))) \right|,$$

where $r = \text{region}(x; \mathbf{x}^n)$, $n = |\mathcal{I}_r|$.

The error heuristics for the basis are

$$\begin{aligned} \underline{a_1(\mathcal{N})} &\equiv \underline{\int_{-1}^1 e(x; \mathcal{N}) \, dx}, \\ \underline{a_2(\mathcal{N})} &\equiv \underline{\left(\int_{-1}^1 e(x; \mathcal{N})^2 \, dx \right)^{1/2}}, \\ \underline{a_\infty(\mathcal{N})} &\equiv \underline{\max_{x \in [-1,1]} e(x; \mathcal{N})}. \end{aligned}$$

The relation Eq. is insensitive to Runge’s phenomenon. For this reason, the Lebesgue constant is often used when designing interpolants, as it bounds worst-case error. However, in our application, we also have the requirement of test-problem stability, thus filtering out node placements that would produce Runge’s phenomenon.

2.1 Metric of instability on a perturbed uniform mesh

Based on the results in Sect. ??, for any t.p.s. \mathcal{N} , we expect $\lambda_{\max}(\mathcal{N}) > 1 + \epsilon$ for some nonuniform element grids. To quantify this instability, we create the following *perturbed uniform mesh metric*, denoted $\lambda_{\max}^{\text{PUM}}(\mathcal{N})$ the GLL basis with $n_p = 2$. Then

$$\underline{x(\mathbf{r}) = \text{normalize} \left(\sum_{j=0}^3 \mathbf{x}_{\text{corner}(j;\mathbf{v})}^{\mathbf{v}} \phi_j^{\text{bilin}}(\mathbf{r}) \right)}. \quad (7)$$

This is the map described in Guba et al. (2014, Appendix A).

To approximate $\lambda_{\max}(\mathcal{N})$ for the test problem, recall that we create the matrix \mathbf{B} corresponding to \mathcal{N} and use the methods described in Sect. ?. In this case, \mathbf{B} represents the action of the ISL operator on one target element. To approximate $\lambda_{\max}^{\text{PUM}}(\mathcal{N})$, we construct \mathbf{B} to represent the action of the ISL operator on multiple contiguous nonuniform target elements, as follows:

First, choose an element subgrid having n_e^{sub} elements

2.0.1 The linear advection operator

Now we write the steps illustrated in Fig. 2. First, each grid point $\mathbf{x}_i^{\mathbf{v}}$ is advected from time t_1 to time t_0 to give

$$\underline{\bar{\mathbf{x}}_i^{\mathbf{v}} \equiv \mathbf{X}(t_0; \mathbf{x}_i^{\mathbf{v}}, t_1)}. \quad (8)$$

In this article, we assume this step is performed nearly exactly, i.e., with numerical error substantially smaller than any other source of error, because this article focuses on the spatial part of the advection discretization. Second, perturb each element boundary by $U(-\delta/2, \delta/2)$, where $U(a, b)$ is the uniform probability distribution over $[a, b]$ and δ is fraction of the subelement size. Thus, a subelement has size $U(1-\delta, 1+\delta)$ relative to the unperturbed size. Third, compute \mathbf{B} as one matrix for all subelements. The result corresponds to periodic translation on a nonuniform element grid tiled by the n_e^{sub} -subelement grid. Fourth, approximate $\lambda_{\max}(\mathbf{B})$ almost the same as usual. The one modification is the set of Δx values; we explain this modification in Sect. ?. To approximate $\lambda_{\max}^{\text{PUM}}(\mathcal{N})$, repeat these four steps multiple times for a set of n_e^{sub} values and take the maximum over the λ_{\max} values. In practice we choose $n_e^{\text{sub}} \in \{3, \dots, 15\}$. The other parameter we must determine is δ . In practice we choose a fixed $\delta = 0.01$, the choice of and insensitivity to which we explain in Sect. ?.

2.1 Search

We wrote software to search for t.p.s.-n.s.-bases. Because there are many of these, especially as n_p grows, we use $a_{1,2,\infty}(\mathcal{N})$ and $\lambda_{\max}^{\text{PUM}}(\mathcal{N})$ to choose among the t.p.s.-ones. ~~tracer grid points are computed from the dynamics grid points using the n_p^v -basis~~
 495 ~~isoparametric map followed by normalization:~~

$$\bar{\mathbf{x}}_i^t \equiv \text{normalize} \left(\sum_{j \in \mathcal{E}^v(\text{elm}(i;t))} \bar{\mathbf{x}}_j^v \phi_{\text{icl}(j;v)}^v(\mathbf{r}_{\text{icl}(i;t)}^t) \right). \quad (9)$$

Third, an interpolated value from time t_0 is assigned to each advected tracer grid point:

$$e \equiv E(\bar{\mathbf{x}}_i^t) \quad (10)$$

$$q_i^t(t_1) = \sum_{j \in \mathcal{E}^t(e)} q_j^t(t_0) \phi_{\text{icl}(j;t)}^t(m_e^{-1}(\bar{\mathbf{x}}_i^t)). \quad (11)$$

500 For a given n_p ,

2.1 Property preservation

Shape preservation and mass conservation on their own are not necessarily difficult to achieve. Rather, the ~~procedure first enumerates all possible o.n.s.-bases, grouped in sets by decreasing n_p^{submin} . All bases use GLL node points. Once it finds a value of n_p^{submin} for which there are t.p.s.-bases and has tested all bases in this set, the procedure switches to enumerating all~~
 505 ~~possible n.s.-bases having this or larger value of n_p^{submin}~~ hard problem in the development of an efficient transport method is the consistent coupling of the transport scheme to the dynamical core, or mass-tracer consistency. Mass-tracer consistency means that the transport solver and dynamics solver agree exactly on the air density field. Achieving both mass-tracer consistency and a long transport time step requires redistributing tracer mass.

In both phases, bases are filtered as follows. Three sets of uniformly spaced bins are constructed over a range of $\log_{10}(\lambda_{\max}^{\text{PUM}} - 1)$ values, roughly 0 down to \log_{10} of machine precision, one set of bins for each of $a_{1,2,\infty}$. Each bin starts empty. Consider a candidate basis. A sequence of analysis steps is performed. If a step fails, the analysis stops and the procedure moves to the next candidate.

1. The basis's nodal weights, where a node's weight is the integral of its associated basis function, are computed; a non-positive weight is a failure. This step is fast. ~~It is possible to couple a tracer transport method to a dynamical core consistently and preserve properties using only local operations. Lauritzen et al. (2017) describe the coupling of a finite-volume, cell-integrated SL method to the HOMME spectral element dynamical core using Lagrangian grid adjustments, where each adjustment is a function of only data in the adjusted entity's local neighborhood. However, in their method, the time step is limited to a Courant number of at most 1, limiting efficiency.~~

2. Its $a_{1,2,\infty}$ values are computed. If no value is better than the best in each bin, the step fails. An empty bin implies
 520 ~~success. This step is fast~~ Our approach to consistent coupling is to use what is sometimes called a *global fixer*, although often

this term applies to only the problem of global mass conservation. In Bradley et al. (2019), we describe several classes of methods that we call Communication-Efficient Density Reconstructors (CEDR). Each CEDR can solve a variety of property preservation problems, including shape preservation and global mass conservation. Each CEDR provides a tight and practically meaningful bound on mass redistribution. A solution is assured if the input data meet necessary and sufficient conditions.

525 Depending on the algorithm, a CEDR uses either one or two global reductions or reduction-like communication rounds. See Bradley et al. (2019, Sect. 1.2) for a discussion of how CEDRs relate to previous global fixers, in particular with respect to solution guarantees and number of communication rounds.

3. The basis is tested for test-problem stability. As we described previously, many small eigenvalue problems are solved in parallel. If any has $\lambda_{\max} > 1 + \epsilon$, the step terminates and fails. A speedup is to search Δx so that the range $(0, 1/2]$ is covered quickly, coarsely at first, and then filled in at a roughly uniform rate everywhere, and the same for θ . This enumeration of these sets encourages fast failures if a basis is not t.p.s. This step is slow if it succeeds but almost always is fast if it fails.

530

4. The largest $\lambda_{\max}^{\text{PUM}}$ value that would permit acceptance of the candidate based on its $a_{1,2,\infty}$ values is computed. This step is fast.

5. $\lambda_{\max}^{\text{PUM}}$ is computed, given the threshold from step 4. If a value is found that exceeds this threshold, this step terminates and fails. This step is slow.

535

6. The basis is added to the bin corresponding to its $\lambda_{\max}^{\text{PUM}}$ value and to every bin with larger $\lambda_{\max}^{\text{PUM}}$ value for which the corresponding $a_{1,2,\infty}$ value is better than the best in that bin. This step is fast.

As the program proceeds, it can filter candidate bases more quickly, since the bins are populated with increasingly good bases. In addition, by starting with o.n.s. bases, the bins get populated quickly with good bases, making the search through the much more numerous general n.s. bases faster. Because we use a CEDR for consistent coupling, we are then free to exploit it in other parts of our transport method to maximize efficiency. For example, we can use an interpolation SL method, which is not mass conserving, because the mass will be corrected at the end of each time step.

540

We have completed this procedure for bases through $n_p = 10$. For bases having $n_p > 10$, we have completed this procedure for o.n.s. bases but only incompletely for general n.s. bases. Exact shape preservation uses mixing ratio bounds computed from each arrival grid point's domain of dependence and does not permit any violation of a domain's nodal extrema. As a result, exact shape preservation clips smooth extrema, limiting the overall transport method to an order of accuracy generally not more than a little above two. Nonetheless, higher order linear advection is still useful because it increases the absolute accuracy of the overall method even if not its order of accuracy, as examples in Sect. 4 will demonstrate.

545

The final step is to select a basis from the set of bases in the bins. We have not developed a rule to select the basis since there is a trade-off between $\lambda_{\max}^{\text{PUM}}$ and accuracy. Instead, we follow a few guidelines to select the final one to recommend. First, we always choose a basis having the maximum n_p^{submin} we have found. Second, within that set, we prefer small $\lambda_{\max}^{\text{PUM}}$ values. A factor of roughly three difference in $\lambda_{\max}^{\text{PUM}}$ values does not matter, but a factor of roughly ten does. We then choose the most accurate basis within the tight cluster of $\lambda_{\max}^{\text{PUM}}$ values we have selected, weighting a_2 more heavily than $a_{1,\infty}$. For $n_p \leq 10$, only $n_p = 6$ and 9 have significantly better non-offset than offset bases. In these two cases, $\lambda_{\max}^{\text{PUM}}$ for the best o.n.s. basis is substantially higher than for the best non-offset one.

550

555

2.1.1 CLIPANDASSURED SUM

In this article, we use the simplest CEDR: CLIPANDASSURED SUM, subsequently CAAS, Algorithm 3.1 in Bradley et al. (2019)

~

2.2 Results

560 Table A1 lists the final set of Islet GLL n.s. bases, subsequently *Islet bases*, one for each value of n_p between 5 and 13. Section ?? describes the basis for $n_p = 4$, which is not a n. s. basis. Each row provides a formula for the row's n_p value. Columns are n_p , order of accuracy (OOA) $n_p^{\text{submin}} - 1$, the support sizes n_p^{sub} for each region ordered left to middle, and the supports. For o.n.s. bases, supports are given by offsets. For general n.s. bases, supports are given by nodal subsets, again ordered from left region to middle. Supports for the right half of To explain CAAS, we need to define the weight associated with a grid

565 point, since weights are needed to define the mass. Conceptually, a weight is an integral over the element of the product of the reference element are determined by the symmetry constraint. The bases for $n_p = 6$ grid point's basis function and the Jacobian determinant of $\mathbf{x} = m(\mathbf{r})$:

$$\bar{\theta}_i \equiv \int_{-1}^1 \int_{-1}^1 \phi_{\text{lcl}(i)}(\mathbf{r}) \left| \frac{\partial m_{\text{elm}(i)}(\mathbf{r})}{\partial \mathbf{r}} \right| d\mathbf{r}. \quad (12)$$

Details of the discretization modify this value. Our manifold is the sphere, and 9 are not offset, we use HOMME's definition

570 of a weight. HOMME discretizes the weight integral using the n_p -basis GLL quadrature. Let

$$v_i \equiv \int_{-1}^1 \psi_i(r) dr.$$

Because $\phi_{\text{lcl}(i)}(\mathbf{r}_{\text{lcl}(j)}) = 1$ when $j = i$ and is 0 for every other value of j , GLL quadrature applied to Eq. (12) gives

$$\begin{aligned} \theta_i &\equiv \sum_{j \in \mathcal{E}(\text{elm}(i))} \left[\phi_{\text{lcl}(i)}(\mathbf{r}_{\text{lcl}(j)}) \left| \frac{\partial m_{\text{elm}(i)}(\mathbf{r}_{\text{lcl}(j)})}{\partial \mathbf{r}} \right| \right] \times \\ &\quad v_{\lfloor \text{lcl}(j)/n_p \rfloor} v_{\text{lcl}(j) \bmod n_p} \\ &= \left| \frac{\partial m_{\text{elm}(i)}(\mathbf{r}_{\text{lcl}(i)})}{\partial \mathbf{r}} \right| v_{\lfloor \text{lcl}(i)/n_p \rfloor} v_{\text{lcl}(i) \bmod n_p} \\ &\equiv J_i w_i. \end{aligned}$$

575

In the final line, we define

$$J_i \equiv \left| \frac{\partial m_{\text{elm}(i)}(\mathbf{r}_{\text{lcl}(i)})}{\partial \mathbf{r}} \right| \quad (13)$$

and w_i as the product of the two 1D basis function integrals. In practice, HOMME modifies each node's Jacobian determinant J_i by multiplying it by a constant very close to 1 such that the global sum $\sum_i J_i w_i = 4\pi$; that is, the sum of all the products of w_i and adjusted J_i gives exactly the area of the unit sphere. This is an optional and minor implementation detail, and we neglect this scaling factor subsequently. Note that in general J is discontinuous across elements.

As an example of an o.n.s. basis, consider the $n_p = 8$ basis. Regions 0 to 3 have $(n_p^{\text{sub}})_r$ values, in region order, of $\{6, 6, 7, 6\}$. There are seven regions total, and region 3 is a middle region because n_p is even. By symmetry, Now we describe CAAS, independent of two sets of details: first, the means by which time- t_1 data are obtained; second, the $(n_p^{\text{sub}})_r$ values for regions 0 to 6 are $\{6, 6, 7, 6, 7, 6, 6\}$. Because n_p^{submin} is 6, corresponding to a degree-5 interpolating polynomial, the basis yields an ISL method having order of accuracy 5. The offsets for regions 0 to 3 are $\{0, 0, 0, 1\}$. Thus, region 0 has support $\{0, 1, 2, 3, 4, 5\}$, and the middle region, region 3, has support $\{1, 2, 3, 4, 5, 6\}$. The support nodes for region 3 are symmetric around the element center to satisfy the symmetry constraint. By symmetry with region 0, region 6 has support $\{2, 3, 4, 5, 6, 7\}$.

$\lambda_{\max}(\Delta x) - 1$ (solid lines) and $\lambda_{\max}^{\text{PUM}}(\Delta x) - 1$ (markers) for the natural GLL (red, small circles), uniform-points offset nodal subset (green, \times), and Islet GLL nodal subset (black, large circle) $n_p = 8$ bases. Green dotted vertical lines mark multiples of $1/(n_p - 1) = 1/7$.

Figure ?? illustrates λ_{\max} -index set \mathcal{E} over which CAAS is applied. Later, we will specify these details for each application of CAAS. At time t_1 and grid point i , we are given air density ρ_i , preliminary tracer mixing ratio \bar{q}_i , and weight θ_i . In addition, we are given a target total mass b over all $i \in \mathcal{E}$. Finally, we have lower and upper bounds, q_i^{\min} and $\lambda_{\max}^{\text{PUM}}$ for three $n_p = 8$ bases. Color corresponds to basis: natural GLL, red; an o.n.s. basis with uniform points, green; Islet, black. The uniform-points basis is an o.n.s. basis with n_p^{sub} values $\{4, 4, 4, 4\}$ and offsets $\{0, 0, 1, 2\}$. Solid lines shows the maximum eigenvalue magnitude of q_i^{\max} , on the target q_i . These are the space-time operator as a function of translation $\Delta x \in (0, 1]$, where again a translation of 1 corresponds to one full element. Note the symmetry around $\Delta x = 1/2$ of these curves. Markers show $\lambda_{\max}^{\text{PUM}}$ as a function of translation Δx , with $\delta = 0.01$, inputs to CAAS. The output is a set of modified values $q(\mathcal{E})$ that solves the 1-norm minimization problem

$$\min \sum_{i \in \mathcal{E}} |q_i - \bar{q}_i|$$

$$\text{subject to } \sum_{i \in \mathcal{E}} \theta_i \rho_i q_i = b \quad (\text{mass conservation})$$

$$q_i^{\min} \leq q_i \leq q_i^{\max} \text{ for } i \in \mathcal{E} \quad (\text{shape preservation}).$$

This problem has a solution, i.e., the constraint set is non-empty, if and only if

$$\sum_{i \in \mathcal{E}} \theta_i \rho_i q_i^{\min} \leq b \leq \sum_{i \in \mathcal{E}} \theta_i \rho_i q_i^{\max}. \quad (14)$$

For details, see Bradley et al. (2019, Sect. 2).

The natural GLL basis is not t.p.s.; thus, $\lambda_{\max}^{\text{PUM}}$ values do not add new information and agree closely with λ_{\max} values except at $\Delta x = 1$, where $\lambda_{\max} = 1$ but $\lambda_{\max}^{\text{PUM}} > 1$. For the Islet basis, the only $\lambda_{\max}^{\text{PUM}}$ value above 1 occurs at $\Delta x = 1$. In the

610 ease of the t.p.s. basis having uniform points, $\lambda_{\max}^{\text{PUM}}$ values above 1 occur at translations that are clustered around multiples of $1/7$ and larger than $1/2$. In general, we observe $\lambda_{\max}^{\text{PUM}} > 1$ clustered around values of Δx at which most of the translated nodes overlie the Eulerian nodes in a different element. For the Islet bases, these are integer values of Δx other than 0. For a t.p.s. uniform-points basis, these are multiples of $1/(n_p - 1) \geq 1/2$. Thus, When there is a solution, there are usually an infinite number, and CAAS efficiently finds one as follows. First, clip the mixing ratios for $i \in \mathcal{E}$:

615 $\hat{q}_i \equiv \min(q_i^{\max}, \max(q_i^{\min}, \bar{q}_i)).$

Let the preliminary mass be

$$\hat{m} \equiv \sum_{i \in \mathcal{E}} \theta_i \rho_i \hat{q}_i.$$

If $\hat{m} = b$, then set $q_i = \hat{q}_i$ for t.p.s. bases, we compute $\lambda_{\max}^{\text{PUM}}$ at Δx values that follow these observations, rather than the values $\Delta x \in (0, 1/2]$ used when computing λ_{\max} . Since the discrete set size is much smaller than for the computation of λ_{\max} , we can run more trials of randomly perturbed subelements in each test. $i \in \mathcal{E}$ and terminate. Otherwise, suppose $\hat{m} < b$. Second, compute the capacity:

620 $c \equiv \sum_{i \in \mathcal{E}} \theta_i \rho_i (q_i^{\max} - \hat{q}_i).$

Third, compute the final values for $i \in \mathcal{E}$:

$$q_i = \hat{q}_i + (b - \hat{m}) \frac{q_i^{\max} - \hat{q}_i}{c}.$$

625 As a check, note that

$$\begin{aligned} \sum_{i \in \mathcal{E}} \theta_i \rho_i q_i &= \sum_{i \in \mathcal{E}} \theta_i \rho_i \hat{q}_i + (b - \hat{m}) \frac{\sum_{i \in \mathcal{E}} \theta_i \rho_i (q_i^{\max} - \hat{q}_i)}{c} \\ &= \hat{m} + (b - \hat{m}) \frac{c}{c} \\ &= b. \end{aligned}$$

If $b - \hat{m} \leq c$, then $q_i \leq q_i^{\max}$ for $i \in \mathcal{E}$; these conditions hold if and only if Eq. (14) does. The case $\hat{m} > b$ is similar. Let

630 $q(\mathcal{E}) = \text{CAAS}(\mathcal{E}, \rho, \theta, b, q^{\min}, q^{\max}, \bar{q})$

carry out these steps.

T.p.s. uniform-points n.s. bases having OOA 3 exist for $n_p \geq 8$, and in terms of accuracy, these are potentially useful. However, because uniform-points n.s. bases exhibit much larger $\lambda_{\max}^{\text{PUM}}$ than GLL-node n.s. bases, we do not consider them further in this article. CAAS is a CEDR because the global version, i.e., the problem in which \mathcal{E} is over all grid points, can be implemented using a single global reduction. For details, see Bradley et al. (2019)[Sect. 6].

635 $\lambda_{\max}^{\text{PUM}}(\delta) = 1$ for the bases in Table A1 with $n_p \geq 6$. The triangle provides a δ^4 reference slope.

2.1.1 Infeasible problems

Figure ?? plots $\lambda_{\max}^{\text{PUM}} - 1$ vs. subelement perturbation δ for the bases in Table A1, excluding those for $n_p \leq 5$ because those curves have a number of drops to machine precision values. Empirically, $\lambda_{\max}^{\text{PUM}} - 1$ is proportional to δ^k , $k = 4$, for these bases, and for a fixed value of δ , it tends to increase with n_p . Not all t.p.s. GLL n.s. bases fall off at this rate $k = 4$. Some fall off more slowly, $k < 4$, indicating greater instability on nonuniform grids. We have not specifically focused a search on attempting to find a t.p.s. basis with rate $k > 4$, but we have not observed any. Based on this plot, when computing $\lambda_{\max}^{\text{PUM}}(\mathcal{N})$ in the basis search procedure, we choose the specific value $\delta = 0.01$, with search results extremely insensitive to this value over a large range of δ . In some situations, the constraint set is empty. There is a sequence of up to two relaxations to the constraints to produce a non-empty constraint set. Algorithm 3.4 in Bradley et al. (2019), RECONSTRUCTSAFELY, which wraps a lower-level algorithm like CAAS, formalizes this sequence. Suppose the original constraint set is empty and $\hat{m} < b$. RECONSTRUCTSAFELY first tries to compute a one-norm-minimal relaxation to q^{\max} such that the maximum value $\max_{i \in \mathcal{E}} q_i^{\max}$ is not exceeded:

$$q(\mathcal{E}) = q^{\max}(\mathcal{E}) + \text{CAAS}(\mathcal{E}, \rho, \theta, b - \sum_{i \in \mathcal{E}} \theta_i \rho_i q_i^{\max}, 0, \max_{i \in \mathcal{E}} q_i^{\max} - q, q^{\max}).$$

This relaxed problem provides mass-tracer consistency and mass conservation but does not solve the exact shape preservation problem. If this relaxation has an empty constraint set, then RECONSTRUCTSAFELY returns $q(\mathcal{E})$ with uniform values that satisfy the mass conservation constraint.

2.2 Optimized interpolant

For the $n_p = 4$ GLL nodes,

2.1.1 Local and global problems

We use CAAS to solve problems at two levels: first, within each element separately; second, over the t.p.s. n.s. basis has n_p^{sub} values $\{3, 4\}$ with offsets $\{0, 0\}$, referred to subsequently as \mathcal{N}_4^3 . In this subsection, we find a t.p.s. interpolant more accurate than this n.s. one whole grid. In addition, at the global level, there are two natural choices for scalar values in the vectors passed to CAAS: grid point values and integrals over elements. In practice, some local problems may not have assuredly non-empty constraint sets, but the global problem always has a non-empty constraint set.

The middle region already has $n_p^{\text{sub}} = n_p$ and thus is not modified. In the left region, a convex combination $\alpha(x)$ is sought so that the basis functions are

$$(1 - \alpha(x)) \mathcal{L}(x; \mathbf{I}^{4 \times 4}(0:2, :), \mathbf{x}_G^4(0:2)) + \alpha(x) \mathcal{L}(x; \mathbf{I}^{4 \times 4}, \mathbf{x}_G^4)$$

for $x \in [x_G^4(0), x_G^4(1)]$. The right region is symmetric to the left, as usual. In the first approach to the global problem, which we refer to as *CAAS-point*, the inputs to CAAS are simply the grid point values. Global and local applications of CAAS are then identical except for the index set \mathcal{E} .

We set $\alpha(x)$, $x \in [x_G^4(0), x_G^4(1)]$, to be a quadratic polynomial having values $c \in [0, 1]^3$ at left, middle, and right points of the region and use the heuristics described in Sections ?? and ??, as well as the necessary condition for stability, to search for the optimal value of c . For the left region, the accuracy heuristic is modified to be $\bar{c}(x) \equiv (1 - \alpha(x))c(x; \mathcal{N}_4^3) + \alpha(x)c(x; \mathcal{N}_4^4)$, where \mathcal{N}_4^4 . In the alternative approach, which we refer to as *CAAS-CAAS*, the global problem is solved over element-level scalar values, and then a second step applies CAAS separately to each element. The global step's task is to provide each element with a sufficient mass adjustment so that each element's local CAAS step has a non-empty constraint set. The global problem's inputs are formulated as follows. Weight e is element e 's area: $A_e \equiv \sum_{i \in \mathcal{E}(e)} w_i J_i$. Air density ρ_e is the standard GLL 4-basis, and similarly for the right region. Results are insensitive to small perturbations to c , so the final numbers are rounded to $c = (1, 0.306, 0)$.

Illustration of the optimized Islet GLL $n_p = 4$ basis (solid line) compared with the natural (dotted) and the best nodal subset (dashed) $n_p = 4$ bases. Each basis function in a basis has its own color. The top panel shows the convex combination parameter value as a function of reference coordinate that is used to combine the natural and best nodal subset bases to form the optimized basis.

Figure ?? illustrates this basis. The top panel shows $\alpha(x)$ in the left region and its mirror image in the right region. The bottom panel shows the natural (dotted), o.n.s. (dashed), and optimized (solid) basis functions. element's average density. Let the element's mass be $m_e \equiv \sum_{i \in \mathcal{E}(e)} w_i J_i \rho_i$. Then $\rho_e = m_e / A_e$. The minimum mixing ratio is the minimum tracer mass over the element divided by the total mass: $q_e^{\min} \equiv \sum_{i \in \mathcal{E}(e)} w_i J_i \rho_i q_i^{\min} / m_e$, and similarly for q_e^{\max} . Finally, the initial value \bar{q}_e is $\bar{q}_e \equiv \sum_{i \in \mathcal{E}(e)} w_i J_i \rho_i q_i / m_e$. This global CAAS step returns element-level mass adjustments. The adjusted mass in an element is then b in the input to the local CAAS step.

3 The Islet method

Our primary application of the Islet bases is tracer transport p -refinement (TPPR). In p -refinement, the element grid is fixed, while the element basis parameter n_p is increased relative to a baseline value to increase solution accuracy. Importantly, both the dynamical equations and the physics parameterizations use their own original discretizations and basis sets. We refer to the set of algorithms composing this application as the *Islet method*. In some configurations we study, not every algorithm is active.

The tracer transport equation in conservation form is

$$\frac{\partial(\rho q_i)}{\partial t} + \nabla \cdot (\mathbf{u} \rho q_i) = \rho f_i(q_1, \dots, q_n; \dots)$$

and in advective form is

$$\frac{Dq_i(\mathbf{y}, t)}{Dt} \equiv \frac{\partial q_i}{\partial t} + \mathbf{u} \cdot \nabla q_i = f_i(q_1, \dots, q_n; \dots)$$

for $i \in \{1, \dots, n\}$, where ρ is total density, \mathbf{y} is the spatial coordinate on the sphere, \mathbf{u} is the flow velocity provided by the dynamics solver, there are n mixing ratios q_i , and f_i are the physical parameterization source terms. Each f_i is a function of the mixing ratios as well as other quantities such as time, space, and variables from the dynamical equations, indicated by the trailing ellipses in the argument list to f_i . The two forms of the equation are related by the mass continuity equation $\partial \rho / \partial t + \nabla \cdot (\mathbf{u} \rho) = 0$. For convenience, we use base-1 indexing when labeling the mixing ratios and source terms.

2.0.1 Relaxed problems

One spectral element (blue solid line outlining the full square) with dynamics (black large circles), tracer (small red circles), and physics (green dashed lines) subelement grids. In the Islet method, multiple local, and one global, property preservation problems are solved in each time step. Not every problem must enforce strict property preservation; strictness is required only in certain outputs, e.g., at the end of a time step. Relaxing the extremal bounds on the mixing ratio grid point values reduces the amount of mass redistribution. More subtly, solving relaxed element-local problems before solving the strict global problem increases intra-element, and decreases inter-element, mass redistribution, thus increasing the locality of mass redistribution. Finally, problems such as the toy chemistry problem described in Sect. 4.3.2 that are sensitive to finite-precision round-off errors benefit from relaxed bounds because the perturbation to the mixing ratio is decreased. Thus, in some element-local CAAS applications in a time step, for each index i , we widen each pair of bounds by 1% of $q_i^{\max} - q_i^{\min}$ in each direction. The global CAAS-point application uses strict bounds, assuring the tracer field is property preserving at the end of a time step.

The Islet method operates on three point grids sharing one element grid. Figure 1 shows one element, outlined by the solid blue line. The *dynamics grid*, black large circles, belongs to the dynamical core, excluding tracer transport, which solves the equations of motion for the atmosphere to provide \mathbf{u} .

2.1 Grid remap

Remapping data among multiple component grids is common in many applications of PDE-based modeling because it is a direct means to permit each component to run in its most efficient configuration. For example, many whole-earth, fully coupled earth system models use different grids for ocean, land, and ρ in Eq. . Occasionally in figures we shorten the name to *v grid*. It is shown using the standard $n_p = 4$ GLL grid; we refer to this value subsequently as n_p^v . The Islet method modifies only tracer transport in the dynamical core. The tracer transport module uses the p -refined *tracer grid*, small red circles, and the Islet basis from Table A1 corresponding to these GLL nodes to evolve the mixing ratios q_i . In Fig. 1, the

tracer-grid n_p value is 8; subsequently, we refer to the tracer-grid n_p as n_p^t . Finally, following Hannah et al. (2021), we couple
 725 the dynamics and tracer transport modules to a separate finite-volume (FV) *physics grid*, shown by the green dashed lines
 outlining an $n_f \times n_f$ grid of squares, $n_f = 6$. The subgrid-scale physics and chemistry parameterizations compute the source
 terms f_i on the physics-grid atmosphere components. Remapping among subcomponent grids is less common. Recent examples
 include separate physics parameterizations and dynamics grids in the atmosphere (Herrington et al., 2019; Hannah et al., 2021)
 ; adaptive mesh refinement (AMR) of a tracer (Chen et al., 2021; Semakin and Rastigejev, 2020); and local vertical refinement
 730 in physics parameterizations relative to the shared background vertical grid, the Framework for Improvement by Vertical
 Enhancement (FIVE) (Yamaguchi et al., 2017; Lee et al., 2020).

The ISL algorithm and the grid transfers are property preserving. First, global, and in some steps also In the Islet method,
 grid remap operators transfer data among the dynamics, tracer, and physics grids. In all of these grid transfers, there is one key
 property: the linear remap operators use only element-local , mass are conserved. Second, tracer mass mixing ratio extrema
 735 are locally (and therefore also globally) constrained from growing data. Thus, a constant mixing ratio is preserved, implying
 in turn that the method is mass-tracer consistent. Property preservation steps apply the local or global algorithms described in
 Bradley et al. (2019); see that reference for further details of property preservation remap between grids is extremely efficient.

In this section, we describe each algorithm that runs in a tracer transport time step, advancing the simulation from time
 step n to $n + 1$. In summary, first, total mass density ρ and trajectory data are remapped Remapping a field between a GLL
 740 grid, dynamics or tracer, and the physics grid is described in detail in Hannah et al. (2021, Sect. 2), and these details are not
 reproduced here. Remapping from the dynamics grid to the tracer grid. Second, tracer tendencies $f_i \Delta t$ are remapped from
 the physics grid to the tracer grid, where Δt is the physics parameterization time step. Third, the remapped tracer tendencies
 are added to the tracer-grid tracers. Fourth, the tracers are advected on the tracer grid. Fifth, tracer states are remapped to the
 physics and, optionally, dynamics grids. We also describe interprocess communication procedures to maximize efficiency.

745 2.2 Preliminaries

This subsection describes notation, definitions, and mathematical details that we use subsequently grid is simple: a physics
 grid subcell's value is assigned the average value of the density over the subcell. Remapping from the physics grid to the
 dynamics grid requires a high-order reconstruction of the data on the physics grid, in addition to maintaining an element-local
 mass-conservation constraint, and most of the details in Hannah et al. (2021, Sect. 2) focus on this high-order reconstruction.

750 Arithmetic between vectors applies entry-wise. All arithmetic applies to a single element of the element-grid unless stated
 otherwise. Sub- or superscript letters in the set $\{v, f, t\}$ indicate which grid the quantity is on, respectively dynamics, finite-volume
 physics, or tracer grids. Location of this letter has no meaning; it is placed where there is room and may switch position for the
 same quantity depending on context.

$\mathcal{I}^{n_p^v \rightarrow n_p^t}$ interpolates Remapping a field between the dynamics and tracer grids, in either direction, is simple; each grid hosts
 755 a field already in a high-order representation, and thus no high-order reconstruction is needed. The first step is interpolation.
 Let $\mathcal{I}^{v \rightarrow t}$ interpolate a field in an element on the dynamics grid, represented by the natural GLL $n_p^v n_p^v$ -basis, to one on the

tracer grid, represented by the Islet $n_p^t n_p^t$ -basis, using the natural $n_p^v n_p^v$ -basis functions to interpolate. $\mathcal{I}^{n_p^t \rightarrow n_p^v}$ does. Let $\mathcal{I}^{t \rightarrow v}$ do the opposite, using the Islet $n_p^t n_p^t$ -basis functions.

760 The weight associated to each node in the subelement grid is, as usual, the integral of its associated basis function, whether natural or not, over the reference element; we denote it as w_i . Unless otherwise indicated, sums of the form \sum_i are over the grid points in an element. We can write these linear operators explicitly using the notation introduced in Sect. 2.1.1. Let f be the grid-point values of a scalar field. Consider element e . Then for local index $i \in \{0, \dots, (n_p^t)^2 - 1\}$,

$$(f_e^t)_i = (\mathcal{I}^{v \rightarrow t} f_e^v)_i \equiv \sum_{j \in \mathcal{E}^v(e)} f_j^v \phi_{\text{icl}(j;v)}(r_i^t),$$

where we have used the notation $f_e \equiv f(\mathcal{E}(e))$. $\mathcal{I}^{t \rightarrow v}$ is written the same way but with t and v switched.

765 The interpolation of a field f from the dynamics grid to the tracer grid, $f^t = \mathcal{I}^{n_p^v \rightarrow n_p^t} f^v$, has the useful property that it is conservative, $\sum_i w_i^t f_i^t = \sum_i w_i^v f_i^v$, within each element e despite not being in all cases an L^2 projection. First, for any Islet basis for which $n_p^{\text{submin}} \geq n_p^v$, the continuum field f is the same on each grid, since the tracer grid can exactly represent polynomials of degree $n_p^v - 1$. Second, for n_p^v even, such as the standard $n_p^v = 4$, if $n_p^{\text{submin}} = n_p^v - 1$, then the continuum field f is in general different on each grid but the integral of each over an element has the same value. This is because for polynomial degree d odd and $f(x) \equiv \sum_{i=0}^d a_i x^i$, $\int_{-1}^1 f(x) dx = \int_{-1}^1 (f(-x) + f(x))/2 dx$, and $g(x) \equiv (f(-x) + f(x))/2$ has degree at least one less than $f(x)$. Consider $f_e^t = \mathcal{I}^{v \rightarrow t} f_e^v$. Then

$$\sum_{i \in \mathcal{E}^t(e)} w_i^t f_i^t = \sum_{i \in \mathcal{E}^v(e)} w_i^v f_i^v. \quad (15)$$

See Appendix A2 for a proof of Eq. (15). $g(x)$ is exactly represented on the tracer grid, and thus the integral of f , which is the same as the integral of g , is the same on the tracer grid as on the dynamics grid. These two reasons assure that $\mathcal{I}^{n_p^v \rightarrow n_p^t}$ is conservative for the Islet bases in Table A1 when $n_p^v = 4$.

Each grid point needs a value for the Jacobian determinant of the map from reference element to sphere, denoted J . Remapping between grids requires, as a second step, applying CAAS for property preservation; thus, we need grid point weight data on each grid. We define J^t using Eq. (13). On the dynamics grid, in practice this value is the usual isoparametric element Jacobian determinant, possibly modified so that the sum over the whole unit sphere is 4π . However, note that what follows is independent of the definition of J^v . On the tracer grid, we instead tracer grid, rather than use Eq. (13), we interpolate the Jacobian determinant values from the dynamics grid,

$$J_e^t = \mathcal{I}^{n_p^v \rightarrow n_p^t} J_e^v, \quad (16)$$

for two reasons. First, this operation conserves the area of the element, the sum $\sum_i w_i^v J_i^v$, as we just discussed

$$\sum_{j \in \mathcal{E}^t(e)} w_j^t J_j^t = \sum_{j \in \mathcal{E}^v(e)} w_j^v J_j^v,$$

785 by Eq. (15). We describe the second reason when discussing ρ in Sect. 3.1. On the physics grid, we follow a similar procedure, explained in Hannah et al. (2021, Sect. 2.2.1), that is specialized to the finite-volume physics grid. Here again, the area of an element on the physics grid is the same as on the dynamics grid. Thus, all three subgrid definitions of the element area agree on element area values.

In a SEM

790 2.2 Direct stiffness summation

In a spectral element method, most operations are performed independently in each element, often leading to discontinuities in a field across element boundaries. The global direct stiffness summation (DSS) operator (Dennis et al., 2012) (Fischer and Patera, 1989; Dennis et al., 2012) restores continuity. At each grid point on an edge of an element, the multi-valued solution is restored to a single value by weighted summation of contributions from each element sharing the grid point. An element's weight at that grid point grid point i is the value $w_i J_i$ normalized by the sum of all contributing elements' values. We will need a generalization of this operator in which an extra factor σ_i is absorbed into the weight in the weighted sum. Let $\delta_{ij}^{\text{grid}}$ be 1 if global indices i and j are associated with the same grid point and 0 otherwise. For global index i , the generalized DSS applied to a preliminary field \bar{y} is written

$$y_i = \text{DSS}(\theta, \sigma, \bar{y}, i) \equiv \frac{\sum_e \sum_{j \in \mathcal{E}(e)} \delta_{ij}^{\text{grid}} \theta_j \sigma_j \bar{y}_j}{\sum_e \sum_{j \in \mathcal{E}(e)} \delta_{ij}^{\text{grid}} \theta_j \sigma_j}, \quad (17)$$

800 where \sum_e is the sum over all elements. In our use of the generalized DSS, $\sigma = \rho$ and $\bar{y} = \bar{q}$. In the standard DSS, σ is all one and $\bar{y} = \rho \odot \bar{q}$. These two use cases give the same value for q_i if ρ is continuous across elements. We use the generalized DSS when we want to make q continuous but leave ρ discontinuous and, in particular, unmodified from its original value.

3 The Islet method

Now that we have described the problem setting and the core algorithms we use, we describe each algorithm that runs in one time step of the Islet method, advancing the simulation from time step n to $n + 1$.

As part of an earth system model, the advection equation Eq. (3) has a source term. f is a function of possibly all the variables in a simulation. The physics parameterizations compute f on the physics grid. In practice there are multiple equations of the form Eq. 3 to solve, e.g., 40 in the E3SM version 2 water cycle simulations (Golaz et al., 2022); because they decouple during one transport time step, we can focus on just one in this section.

810 In summary, first, air density ρ and trajectory data are remapped from the dynamics grid to the tracer grid. Second, $\Delta q \equiv f \Delta t$ is remapped from the physics grid to the tracer grid, where Δt is the physics parameterization time step. Either of f or Δq is sometimes called a *tendency*. Third, the remapped tracer tendencies are added to the tracer-grid tracers. Fourth, the tracers are advected on the tracer grid. Fifth, tracer states are remapped to the physics and, optionally, dynamics grids.

3.1 Algorithms

815 ~~We~~ There are a number of algorithmic steps in a time step. To organize these algorithms, we name each algorithm using the format **Step::algorithm-name** and sometimes **Step::algorithm-name::sub-algorithm-name**. Unless stated otherwise, each operation acts within a single element **Step::algorithm-name** and sometimes **Step::algorithm-name::sub-algorithm-name**.

Step::density-d2t. Interpolate $J\rho^{n+1}$ **Step::density-d2t**. Interpolate $J \odot \rho^{n+1}$ from the dynamics grid to the tracer grid using the natural GLL n_p^v -basis interpolant: $\rho_t^{n+1} = (\mathcal{I}^{n_p^v \rightarrow n_t^t}(J_v \rho_v^{n+1})) / J_t$. This quantity will not be needed for several steps, but
820 we describe it here because $J_t \rho_t^n$ is needed in **Step::tendency-f2t**. In each element e , compute

$$(\rho_e^t)^{n+1} = (\mathcal{I}^{v \rightarrow t}(J_e^v \odot (\rho_e^v)^{n+1})) \odot J_e^t.$$

This grid transfer is conservative because it uses $\mathcal{I}^{n_p^v \rightarrow n_t^t} \cdot J_t \rho_t^{n+1}$ conserves $J_e \odot \rho_e^{n+1}$ by Eq. (15). $J^t \odot (\rho^t)^{n+1}$ is not continuous across element boundaries because $J \cdot J^t$ is not, but continuity is not needed. In the case that ρ Equation (16) implies that if ρ_e is constant on the dynamics grid, then it is constant on the tracer grid, as follows from Eq. . This is not a necessary property
825 to have too. Mapping a constant air density exactly is not necessary, but since it is possible, we use it; it do; this property is the second reason to define $J^t \cdot J^t$ according to Eq. (16).

Step::tendency-f2t **Step::tendency-f2t**. Map the tracer tendencies Δq^n tendency Δq^n from the physics grid to the tracer grid. This step involves multiple sub-algorithms.

Step::tendency-f2t::bounds **Step::tendency-f2t::bounds**. In each element e , compute and store the minimum and maximum mixing-ratio-state q_t^n values values of $(q_e^f)^n$, subsequently extrema. Let an element's neighborhood contain itself and
830 its immediate element neighbors every other element that shares a vertex with it. Augment an element's extrema with the extrema over its neighborhood. Finally, augment an element e 's extrema again with the extremal values of the tracer-grid mixing ratio state $q_t^n(q_e^f)^n$. This final extrema update assures that if $\Delta q_t^n = 0$, then q_t^n is unmodified $\Delta(q_e^f)^n = 0$, then $(\bar{q}_e^f)^n$ is unmodified in **Step::tendency-f2t::CAAS**. These final extrema in an element are the bounds used in subsequent property
835 preservation corrections.

Step::tendency-f2t::linear-remap **Step::tendency-f2t::linear-remap**. In each element e , apply the linear, element-local element-local, conservative, panel-reconstruction (PR) panel-reconstruction remap operator described in Hannah et al. (2021, Sect. 2.2.3) to map the tendency from the physics grid to the tracer grid. In the Islet method, but unlike in Hannah et al. (2021), the basis used in the mass matrix of the L^2 projection is the Islet basis. From the previous step's application of
840 **Step::density-d2t** **Step::density-d2t**, we have $\rho_t \rho_e^t$. The operator uses this quantity and $\rho_t \Delta q_t^n$ to compute $\rho_t \Delta q_t^n \rho_e^f \odot \Delta(q_e^f)^n$ to compute $\rho_e^t \odot \Delta(q_e^f)^n$.

Step::tendency-f2t::CAAS **Step::tendency-f2t::CAAS**. In each element e , apply CAAS, apply CLIP AND ASSURED SUM, subsequently CAAS, Algorithm 3.1 in Bradley et al. (2019), to $\bar{q}_t^n \equiv q_t^n + \Delta q_t^n$. Each node weight is the integral of the corresponding Islet basis function times J . In this application of the element-local CAAS algorithm, the described in Sect. 2.1.1, to $(\bar{q}_e^f)^n \equiv (q_e^f)^n + \Delta(q_e^f)^n$
845 using the weight vector $\theta_e^t \equiv w_e^t \odot J_e^t$ and air density $(\rho_e^f)^n$. Target mass b is set to the current total element mass because **Step::tendency-f2t::linear-remap** is conservative; thus, no adjustment to the total element mass is needed. The same up-

per and lower bounds apply to each GLL node in the element. In this step, the bounds that **Step::tendency-f2t::bounds** **Step::tendency-f2t::bounds** computed are relaxed in each direction by 1% of the difference between upper and lower bounds: ~~This relaxation is one part of obtaining good toy chemistry diagnostic values, as we shall describe, as discussed~~ in Sect. 4.3.22.0.1.

850 The exact bounds will be enforced in **Step::CEDRStep::CEDR::global**. The constraint set of mass conservation and bounds nonviolation is non-empty because the constant mixing ratio is a solution, as explained in Hannah et al. (2021, Sect. 2.3).

Step::tendency-f2t::DSSStep::tendency-f2t::DSS. At this point, $\bar{q}_t^n - (\bar{q}^t)^n$ is discontinuous across element boundaries. Neither order of accuracy nor property preservation requires continuity, but we find ~~the toy chemistry diagnostic value (see Sect. 4.3.2) is large without restoration of continuity prior to the ISL step.~~ ρ_t^n that continuity improves the solution quality.

855 $(\rho^t)^n$ can remain discontinuous. ~~Because of this, the DSS arithmetic is slightly different than usual. Consider an element boundary node indexed by global ID g and having DSS weights w_g^e —the product of reference-to-sphere Jacobian determinant and Islet basis function weights—for contributing elements indexed by e . In element e , $(\rho_t^n)_g^e$ is the restriction of ρ_t^n to node g . Then the post-DSS value at the node is~~

$$(\hat{q}_t^n)_g^e \equiv \frac{\sum_e w_g^e (\rho_t^n)_g^e (\bar{q}_t^n)_g^e}{\sum_e w_g^e (\rho_t^n)_g^e},$$

860 ~~where the sum \sum_e is over elements that contain node g . In contrast, in the standard DSS, $(\rho_t^n)_g^e$ is absorbed into the DSSed quantity, so that the tracer density, rather than the mixing ratio, is made continuous. Thus, we apply the generalized DSS, Eq. (17), to $(\bar{q}^t)^n$, using $(\rho^t)^n$ to obtain the continuous field $(\hat{q}^t)^n$.~~

Step::advect-interpStep::advect-interp. Compute and apply the linear ~~space-time ISL operator~~ advection ISL operator described in Sect. 2.0.1. This step involves two substeps: computing the grid point trajectories and computing the interpolants.

865 **Step::advect-interp::trajectoryStep::advect-interp::trajectory**. Compute Eq. (8). The dynamics component supplies velocity data at the dynamics GLL grid points. Any of a number of algorithms can compute departure points at time n backward in time from dynamics-grid arrival GLL points at time $n + 1$. The Islet method takes as input these departure points as 3D Cartesian departure points. Next, compute Eq. (9). In each element, the natural GLL interpolant $\mathcal{I}^{n_p^v \rightarrow n_p^t} \mathcal{T}^{v \rightarrow t}$ is applied to each of the Cartesian components separately to ~~obtain~~ provide departure points on the tracer grid. This procedure implies that ;

870 ~~first, adjacent elements compute identical departure points at shared boundaries and, second, at GLL points common to the n_p^v - and n_p^t -bases, the resulting departure points are identical.~~ Finally, for simplicity in the subsequent sphere-to-reference map computations, the 3D Cartesian points are normalized to the sphere. For $n_p^v = 4$, ~~the $n_p^t n_p^v = 4$, the n_p^t -basis~~ departure points are obtained at OOA order of accuracy 4.

Step::advect-interp::mixing-ratioStep::advect-interp::mixing-ratio. Compute Eq. (10). Each departure tracer-grid GLL

875 ~~point node~~ is mapped to the containing element, subsequently the *source* element. Details of finding the source element depend on host-model implementation details and are omitted here; possibilities include octree search, $O(1)$ arithmetic for quasi-uniform cubed-sphere element grids having certain reference-to-sphere maps, and search within a predefined element neighborhood whose size is proportional to maximum wind speed times advection time step. Then the corresponding reference coordinates within the source element are computed using Newton’s method. Next, compute Eq. (11). The mixing ratio value

880 is computed at the departure point using the Islet basis interpolant and the source element’s ~~\hat{q}_t^n values.~~ $(\hat{q}^t)^n$ values. In addi-

tion, the source element's stored extrema are associated to this point as bounds. Finally, the mixing ratio value and bounds are assigned to the target GLL point-node on the arrival tracer grid. Departure points and interpolant weights are calculated once and then are reused for each tracer.

Step::CEDR. ~~Apply local and then global~~ **Step::CEDR.** Apply Communication-Efficient Density Reconstructors (CEDR) (Bradley et al., 2019) on the tracer grid. ~~As usual, node weights correspond to Islet basis functions and not to the natural GLL basis, first to each element and then globally. In this work we use CAAS, described in Sect. 2.1.1. Node weights are $w^l \odot J^l$.~~

Step::CEDR::local **Step::CEDR::local.** First, apply the element-local CAAS algorithm. ~~Neither order of accuracy nor property preservation requires this step. But this step~~ This step is not required, but it reduces the amount of global mass redistribution in **Step::CEDR::global** **Step::CEDR::global**, is computationally inexpensive, ~~and~~ involves no interprocess communication, and thus is worth including ~~in the overall procedure. Again.~~ As in **Step::tendency-f2t::CAAS**, we relax bounds in each direction by 1% of the difference between upper and lower bounds. Unlike in **Step::tendency-f2t::CAAS** **Step::tendency-f2t::CAAS**, in this application of the element-local CAAS algorithm, any two target GLL nodes in an element may have different bounds; the bounds depend on the source element for the target GLL node, as detailed in **Step::advect-interp::mixing-ratio** **Step::advect-interp::mixing-ratio**. At this point, the global mass has not yet been corrected; thus, this local CAAS application's mass constraint is to maintain the element's current tracer mass. The constraint set is not assuredly ~~feasible~~ non-empty. Thus, **Algorithm 3.4 in Bradley et al. (2019)**, RECONSTRUCTSAFELY, described in Sect. 2.1.1, wraps the call to CAAS. ~~This algorithm relaxes the bounds according to degree of infeasibility before calling CAAS.~~

Step::CEDR::global. ~~In this work, we use global CAAS applied at the tracer grid GLL gridpoint level; we refer to the procedure as CAAS-point~~ **Step::CEDR::global.** Apply CAAS-point, described in Sect. 2.1.1, on the global tracer grid. The exact, rather than relaxed, bounds are applied to each node. The inputs to CAAS-point are as follows: the air density $(\rho^l)^{n+1}$; the weights $\theta \equiv w^l \odot J^l$; $b = \sum_i w_i^l J_i^l (\rho_i^l)^n (\hat{q}^l)^n$, the global tracer mass after the tendency update; the mixing ratio bounds computed in **Step::tendency-f2t::bounds**; and the current mixing ratio values from **Step::CEDR::local**. Let the output values be $(q^l)^{n+1}$.

Continuity across element boundaries was restored to the mixing ratio field in **Step::tendency-f2t::DSS** ~~and was maintained in subsequent steps; this step also maintains it.~~ In **Step::tendency-f2t::DSS**. Subsequent steps maintain it in exact arithmetic. However, in finite precision, continuity ~~does not hold to machine precision. However, first, no step of the overall algorithm is sensitive to this level of error; second, the error in $(q^l)^{n+1}$ holds only to a little above machine precision and not exactly. This numerical discontinuity~~ does not grow in time because, in each step, **Step::advect-interp::mixing-ratio** restores time step, both **Step::tendency-f2t::DSS** and **Step::advect-interp::mixing-ratio** restore exact continuity in finite precision. ~~The inputs to,~~ although only one restoration per time step is necessary to prevent growth of the numerical discontinuity. No step of the overall algorithm is sensitive to this small and roughly temporally constant numerical discontinuity.

At this point, the global CAAS algorithm are as follows: $J_l \rho_l^n$ and \hat{q}_l^n to provide the global tracer mass after the tendency update, $J_l \rho_l^{n+1}$ as part of the global tracer mass at time step $n + 1$, the mixing ratio bounds computed in **Step::tendency-f2t::bounds**, and the current mixing ratio values from **Step::CEDR::local**. The final tracer-grid values are q_l^{n+1} time step is complete; the remaining computations remap the tracer grid data between grids.

Step::state-t2f **Step::state-t2f**. Remap the mixing ratio state to the physics grid. This is a purely element-local operation. First, the linear operator described in Hannah et al. (2021, Sect. 2.2.1) is applied to the tracer density. Second, the element-local CAAS algorithm is applied on the physics grid, with the extremal mixing ratio values in the element on the tracer grid as the bounds. For the same reason as in **Step::tendency-f2t::CAAS** **Step::tendency-f2t::CAAS**, the constraint set is assuredly
 920 non-empty.

Step::state-t2v **Step::state-t2v**. The dynamics solver needs one or more mixing ratios on the dynamics grid, e.g., specific humidity. In addition, in our numerical results [in Sect. 4](#), we compute all errors, except as indicated, on the dynamics grid, so we use this step to obtain those errors. First, in an element, ~~the Islet basis is used to interpolate~~ $\mathcal{T}^t \rightarrow v$ ~~interpolates~~ the tracer-grid mixing ratio to the $n_p^v n_p^v$ -basis. Second, the element-local CAAS algorithm is applied on the dynamics grid to preserve
 925 shape and conserve mass; ~~the details are as in~~ **Step::state-t2f** **Step::state-t2f** but with GLL nodes instead of **FV-subcells**. ~~finite-volume subcells. The result is~~ $(q^v)^{n+1}$. Third, the standard DSS is applied to $(\rho^v)^{n+1} \odot (q^v)^{n+1}$, where $(\rho^v)^{n+1}$ is the ~~continuous air density from the dynamical core, to~~ obtain continuous tracer density and mixing ratio fields.

Step::state-v2t **Step::state-v2t**. In validation problems in Sect. 4, we need to remap a mixing ratio initial condition from the dynamics grid to the tracer grid. The algorithm is the same as **Step::state-t2v** **Step::state-t2v** except that the DSS follows the
 930 procedure in **Step::tendency-f2t::DSS** ~~since the mass~~ **Step::tendency-f2t::DSS** ~~because the air~~ density on the tracer grid is and remains discontinuous at element boundaries.

If $n_p^t = n_p^v n_p^t = n_p^v$, then tracer transport p -refinement (**TTPR**) is not enabled. In this case, identity maps replace a subset of the algorithms described in this subsection: **Step::density-d2t** **Step::density-d2t**, the interpolation part of **Step::advect-interp::trajectory**,
Step::state-t2v, and **Step::state-v2t** **Step::advect-interp::trajectory**, **Step::state-t2v**, and **Step::state-v2t**. When describing
 935 numerical experiments, we indicate when TTTPR is not enabled.

3.2 Interprocess communication

~~To realize the full computational efficiency of an ISL method, we must limit the interprocess communication to the union of the discrete domains of dependence of the grid points on an owning process. In addition, our methods must run on GPU architectures; therefore, the assembly and parsing of messages must be highly parallel within a process. This subsection~~
 940 ~~discusses interprocess communication for the Islet method, including some high-level details relevant to an efficient GPU implementation.~~

~~A process owns one or more elements and the grid points within those elements. A process communicates with remote processes, or remotes. We say that an element that is owned by a process or remote is on that process or remote. Each element has a halo. A 0-halo is the element; an $(h+1)$ -halo, $h \geq 0$, is the union of the h -halo and the elements adjacent to elements~~
 945 ~~in the h -halo. The time step is restricted according to a configurable maximum h ; in our implementation, we set $h = 2$. Recall that a grid point's source element is the element containing its Lagrangian departure point; its target element is the element containing its Eulerian arrival point.~~

3.1.1 Advection step

Step::advect-interp::mixing-ratio has two communication rounds.

950 The first communication round exchanges lists of departure points. Each process prepares one message for each of its potential partners, where a second process is a potential partner if it owns at least one element in the h -halo of the first process's owned elements. We call this message an x -message.

An x -message contains two sections: first, metadata to identify the source element for each departure point on the remote; second, the bulk data list of departure points whose source elements are on that remote. For efficiency, during model initialization,
955 each process computes its remotes' local IDs for each element in its halo; then during time stepping, the source element identification data in the x -message is the remote's local ID for the element. This procedure means that only direct array lookups rather than hash- or tree-based map lookups are needed during time stepping. If a potential remote has no source elements in a step, then the message is essentially empty but is still sent since it is expected. A process is a potential partner of itself. In this special case, the list is implicit in data structures reserved for departure points that remain on the owning process;
960 we call this list the x -self-list.

In a GPU implementation, assembling these lists involves parallel-scan and parallel-for loops as well as atomic-access writes to bookkeeping data structures. The parallel-scan loop writes the x -message metadata and computes pointers into the x -message and future received q -message bulk data. The parallel-for loop copies departure points into the x -message bulk data.

965 While a process waits to receive its x -messages, it computes extrema in each owned element.

The second communication round exchanges interpolated mixing ratios and source element extrema. A process receives one x -message from each of its potential remotes. For each departure point in an x -message, it computes the interpolated mixing ratio. Then it writes two sets of data to a new message, the q -message corresponding to the x -message. The first set contains mixing-ratio extrema for each active source element. It writes these data once per active source element per remote. The second
970 set contains the interpolated mixing ratio data, one for each departure point. If an x -message is empty, the process neither writes nor sends a corresponding q -message.

In a GPU implementation, a parallel-scan loop, one per potential remote, is used to parse an x -message's metadata and compute pointers into the q -messages. Then one parallel-for loop over all requested departure points is used to compute and write the q -messages, using these pointers. Atomic accesses are not needed in this communication round.

975 After a process sends all of its q -messages to remotes, and while it is waiting to receive its q -messages, it computes interpolated mixing ratio data corresponding to entries in its x -self-list. This step requires one parallel-for loop and no atomic accesses.

When a process receives all of its q -messages, it copies the data to its internal data structures. Bookkeeping that was done while forming and writing x -messages permits this step to be done with one parallel-for loop and no atomic accesses.

980 Finally, the process waits on its sent q -messages to clear the MPI buffers. When the waits are done, **Step::advect-interp::mixing-ratio** is complete.

3.1.1 Communication volume

Communication volume, sometimes just *volume* when the context is clear, is the volume of data transmitted during interprocess communication. Consider the case of one element per process. The worst-case volume per element of both the x -messages and q -messages is proportional to the number of grid points, $(n_p^t)^2$. For example, at most $(n_p^t)^2$ total interpolated mixing ratio values must be communicated from the source elements to the target element, regardless of halo depth. In contrast, a naive h -halo exchange of full element mixing ratio data would communicate $c_h(n_p^t)^2$ values, $c_h = (2h + 1)^2 - 1$; accounting for continuity, the number could be reduced to a little more than $c_h(n_p^t - 1)^2$. Thus, the volume reduction factor is at least $[(2h + 1)^2 - 1](n_p^t - 1)^2 / (n_p^t)^2$. As examples, for $h = 1$, $n_p^t = 4$, the factor is 4.5; for $h = 2$, $n_p^t = 4$, 13.5; for $h = 2$, $n_p^t = 8$, over 18.

The volume of extrema data depends on the number of source elements a departure element overlaps and very little on n_p^t , since only one pair of extrema data per source element is communicated to a partner. Thus, the proportion of the q -message that contains extrema data decreases with increasing n_p^t .

Other communication rounds require only standard SEM halo exchanges that are already part of a SEM host model. These include exchanging element-edge data, proportional to n_p^v , when computing departure points; element single-scalar data, thus independent of subgrid parameters, to share source element extrema in **Step::tendency-f2t::bounds**; and another element-edge exchange, proportional to n_p^t , in the DSS in **Step::tendency-f2t::DSS**.

CAAS-point requires a reduction. Many climate host models expect answers to be invariant to the number of MPI processes and thus provide a reproducible reduction routine. CAAS-point can maintain reproducibility while keeping the reproducible reduction independent of n_p^t by reducing arrays within an element, then calling the global reproducible reduction routine with element-level values. See Bradley et al. (2019, Sect. 6.1) for further details on CEDR communication.

In summary, only x - and q -messages have volume proportional to $(n_p^t)^2$; all other messages have volume proportional to n_p^t or n_p^v or independent of subgrid parameters. In addition, x - and q -messages have volume substantially smaller than would occur with full halo exchanges.

4 Numerical results

This section presents results for a number of validation problems. Except in Sect. 4.3, the equation is the sourceless advection equation, $f_t = 0$ in Eq. ?? Time-dependent flow u is imposed(4). Two-dimensional, time-dependent flow, $u(x, t)$, is prescribed on the sphere.

In most figures, we show results for $n_p^t = n_p^t = 4, 6, 8, 9, 12$. The $n_p^t = n_p^v = 4, n_p^t = n_p^v = 4$ case provides a reference because it does not use the TTPR algorithms. $n_p^t = 6$ is of interest because it is The $n_p^t = 6$ case has the smallest value of n_p^t with basis OOA n_p^t providing order of accuracy (OOA) greater than 2, in this case 4. $n_p^t = 8$ has The $n_p^t = 8$ case provides OOA 5 and has four times as many nodes as the $n_p^t = 4, n_p^t = 4$ basis in two dimensions. $n_p^t = 9$ has The $n_p^t = 9$ case provides OOA 6. Finally, $n_p^t = 12$ has the $n_p^t = 12$ case provides OOA 8 and has four times as many nodes as the $n_p^t = 6, n_p^t = 6$ basis.

Tests follow the procedures detailed in Lauritzen et al. (2012). Results can be compared with those from many models described in Lauritzen et al. (2014). We refer to these articles frequently and thus abbreviate them as TS12 (“test suite”) and TR14 (“test results”), respectively. Initial conditions are generated on the dynamics grid. Similarly, error diagnostics are computed on the dynamics grid in most cases; we state the exceptions when they occur. In most cases we omit results for simulations without property preservation, as tracer transport modules in earth system models are expected to be property preserving. We have not attempted to make this section self-contained, as describing details of the large number of validation problems would take too much space. We recommend the reader not familiar with these problems read TS12. In addition, we refer to specific figures in TR14 and sometimes TS12 so the reader can compare our results with those from previously documented methods.

We briefly summarize the key characteristics of the validation problems. There are two prescribed flows: a nondivergent one and a divergent one. Each prescribes a flow that lasts for $T = 12$ days and such that at time T , the exact solution is the same as the initial condition. This 12-day prescribed flow can be run for multiple cycles to lengthen the simulation. The nondivergent flow creates a filament of maximum aspect ratio at time $T/2$. The divergent flow tests treatment of divergence. There are four initial conditions that share the feature of placing two circular shapes at two points along the equator: the C^∞ Gaussian hills, the C^1 cosine bells, the correlated cosine bells used in the mixing diagnostic, and the discontinuous slotted cylinders. Assessing the behavior of a transport method on tracers having various degrees of continuity-smoothness is important because atmosphere tracers can be smooth or nonsmooth.

This article does not study time integration methods to generate the dynamics-grid departure points. Thus, to remove temporal errors due to time integration algorithms, we use an adaptive Runge-Kutta method (Dormand and Prince, 1980; Shampine and Reichelt, 1997) with a tight tolerance (10^{-8} , with an exception noted later) to integrate trajectories very accurately. All tests for $n_p^t > 4$ use TTPR with $n_p^v = 4$ unless we state otherwise. Since $n_p^v = 4$ in these validation problems, the $n_p^t = 4$ configuration does not use TTPR.

~~We use a quasiuniform cubed-sphere element grid. Let a cube face of the cubed-sphere element grid have $n_e \times n_e$ elements. Long tracer time steps correspond to $6n_e$ steps per T ; short, $30n_e$. These are the same time step settings as the two CSLAM (Lauritzen et al., 2010) model configurations used in TR14.~~

~~An alternative to running CAAS-point is to run the global CAAS method over elements rather than grid points, then to apply element-local CAAS to each element; As we discussed in Sect. 2.1.1, there are two natural approaches when applying CAAS globally, which we refer to this procedure as CAAS-CAAS. See Bradley et al. (2019, Sect. 7.2) for further details and CAAS-point. CAAS-CAAS tends to give slightly more accuracy than CAAS-point. But However, it does not permit the bound relaxation-relaxations in the element-local CAAS application that we find is necessary to obtain good results for the toy-chemistry diagnostic applications that improve the solution quality when simulating source terms sensitive to round-off errors, as discussed in Sect. 2.0.1. However, because it provides slightly more accurate results, we use it for $n_p^t = 4$ CAAS-CAAS for $n_p^t = 4$, except where noted, while using CAAS-point for all other n_p^t values, thus maximizing the accuracy obtained with $n_p^t = 4$ to provide the best baseline performance ($n_p^t = n_p^v = 4$) accuracy.~~

The ~~total-mass-density~~ ρ ~~air density on the dynamics grid,~~ ρ^v , is not needed for ~~the~~ advection of mixing ratios, but it ~~is-and its remapped versions are~~ needed for property preservation~~of these~~. In these validation problems, we have no independent computation of density ρ^v , as occurs when transport is coupled to a full dynamical core. For the nondivergent flow ~~problem~~, we could ~~of-course just set~~ ρ ~~set~~ ρ^v to a constant, but for the divergent flow ~~, that would be problem, a constant is~~ incorrect. Thus, ~~within our standalone validation code,~~ we need a means to compute ρ ~~on the dynamics grid. We discretize the Lagrangian formulation of the continuity equation,~~ $\int_{A_e(t_2)} \rho(x, t_2) dx = \int_{A_e(t_1)} \rho(x, t_1) dx$, using $\rho(x_i, t_2) J_e(x_i, t_2) = \rho(x_i^*(t_1), t_1) J_e(x_i^*(t_1), t_1)$. In these expressions, x_i is an Eulerian grid point, x_i^* is its departure point at t_1 , $A_e(t)$ is the domain of Lagrangian element e , $A_e(t_2)$ is the domain of the element in its arrival Eulerian position, and J_e is the isoparametric determinant of the map from reference element to spherical element e . J_e is a function of time because $J_e(\cdot, t_2)$ corresponds to its arrival Eulerian configuration and $J_e(\cdot, t_1)$ to its departure Lagrangian configuration. First, ρ is advected on the dynamics grid using ~~Step::advect-interp::m~~. Second, the resulting value at grid point i in element e is multiplied by the density factor $J_e(x_i^*(t_1), t_1) / J_e(x_i, t_2)$. Third, ~~because this discretization~~ ρ^v as a surrogate for a dynamics solver. Appendix B describes the linear advection part of computing ρ^v . After the linear advection operator is applied, because it is not mass conserving, the ~~mass-density~~ is corrected by adding $\Delta m / a_{\text{total}}$ to each grid point, where Δm is the global mass discrepancy after advection and a_{total} is the total area of the grid. Negative density does not occur in these validation problems. ~~The algorithm for~~ ρ^v has OOA 2 for $n_p^v = 4$.

4.1 Unstable and stable integration

Stability of the Islet method with the Islet GLL bases, compared with the instability of the method with the natural GLL bases. The x -axis is average dynamics grid-point spacing at the equator in degrees for the quasiuniform cubed-sphere grid. The y -axis is $\log_{10} l_2$ relative error. A curve's line pattern corresponds to basis type and number of cycles, as listed in the top legend. A curve's marker corresponds to n_p^l , as listed in the bottom legend. The case is divergent flow, Gaussian hills ICs, property preservation, TTPR, and long time steps. We use a quasiuniform, equiangular cubed-sphere element grid. Let a cube face of the cubed-sphere element grid have $n_e \times n_e$ elements. Each element has an $n_p^v \times n_p^v$ tensor grid of GLL nodes and thus $n_p^v - 1$ intervals between GLL nodes along each direction of an element. GLL nodes are mapped to the sphere using the isoparametric map for a bilinear element, as described in Appendix A of Guba et al. (2014) and Eq. (7). Long tracer time steps correspond to $6n_e$ steps per T ; short, $30n_e$. For the test flows, these correspond to, respectively, approximately 5.5 and 1 times the maximum Courant number. These are the same time step settings as the two CSLAM (Lauritzen et al., 2010) model configurations used in TR14. Essentially all SL methods, particularly when given exact trajectory data, exhibit greater error with smaller time step, e.g., CSLAM in TR14. This is because the only source of error, given exact trajectories, is the remap error. Smaller time steps correspond to more remaps to reach a fixed simulation time.

Figure A1 compares accuracy and stability between natural and Islet bases. The divergent flow, Gaussian hills IC, property preservation, TTPR, and long time steps are used. Figures 3–9 and Fig. 15 show convergence plots, and we explain the format of these figures [here](#). A curve's marker corresponds to $n_p^l n_p^l$, as listed in the legend. To maximize font size and minimize notational clutter in figures, in figures we use n_p rather than n_p^l , n_p rather than n_p^l , and omit “=”, e.g., $n_p^l = 8$ is written n_p , “ $n_p^l = 8$ ” is written “ n_p , 8”; additionally, sometimes we omit n_p entirely, as in the legend of this figure”. Additionally,

the legend omits “ n_p ” entirely. The x -axis is the average dynamics-grid point-node spacing at the equator in degrees for the a cubed-sphere grid with north and south cube faces centered at the poles. Thus, for example, $n_e=5$ corresponds to $360^\circ/(4 \text{ faces} \times n_e \text{ elements/face} \times (n_p^v-1) \text{ regions/element})=6^\circ/\text{region}$. $n_e=5$ and $n_p^v=4$ correspond to the resolution

$$\frac{360^\circ}{4 \text{ cube faces}} \cdot \frac{1 \text{ cube face}}{5 \text{ elements}} \cdot \frac{1 \text{ element}}{(4-1) \text{ intervals}} = \frac{6^\circ}{\text{interval}}.$$

The y -axis is $\log_{10} l_2$ relative error. A curve’s line pattern corresponds to basis type and number of cycles: solid, Islet for 1 cycle; dashed, Islet for 100 cycles (or $12 \times 100 = 1200$ days); dash-dotted, natural for 1 cycle. The dotted straight line is a reference \log_{10} of the relative error, with the norm or norms indicated in each figure. The title of a plot lists details of the configuration: the test flow, the initial condition (IC), the time step (short or long), and, if property preservation is disabled, an extra line stating that. Many plots have a dotted line showing the slope for OOA 2. For each n_p^l , the l_2 norm of the solution using the natural basis diverges with increasing resolution within the first cycle, demonstrating that the basis leads to an unstable Islet method. In the case of $n_p^l=4$, we see the start of the curve’s divergence, but further element-grid refinement is needed to see the curve fully diverge. In contrast, the curves for the Islet method with the Islet bases converge at OOA 2.

4.1 Accuracy for C^∞ and C^1 tracers

4.1.1 Time integration check

4.1.1 Accuracy limited only by trajectories

The first experiment tests the OOA limit due to computing trajectories on the tracer grid using velocities from the dynamics grid. Because the dynamics grid uses $n_p^v=4$ and $T^{v \rightarrow h}$ uses the natural GLL basis, we expect this OOA limit to be 4. Property preservation is turned off to expose this OOA limit.

Sometimes validation problems have unexpected features that interact with a method to produce higher-accuracy solutions than would occur in a more realistic problem. Of particular concern is the symmetry of the flow in time around the midpoint time, 6 days. To be sure our trajectory interpolation procedure is not interacting with this symmetry to produce artificially higher accuracy, we run a test in which in this experiment we compute the solution error is computed at the midpoint time as well as the final time.

Comparison of relative errors calculated at the test simulation’s midpoint time of 6 days (1/2 cycle, dashed lines) and endpoint time of 12 days (1 cycle, solid lines). Each number at the right side of the plot is the empirical OOA computed using the final two points of the 1-cycle result.

The test uses the nondivergent flow, the Gaussian hills IC, and long time steps. Property preservation is turned off to expose fully the temporal error the long time step. The midpoint reference solution is computed using one 6-day step and the natural GLL basis. The time integrator’s relative error tolerance is set to 10^{-14} rather than its usual 10^{-8} for this step. One remap step

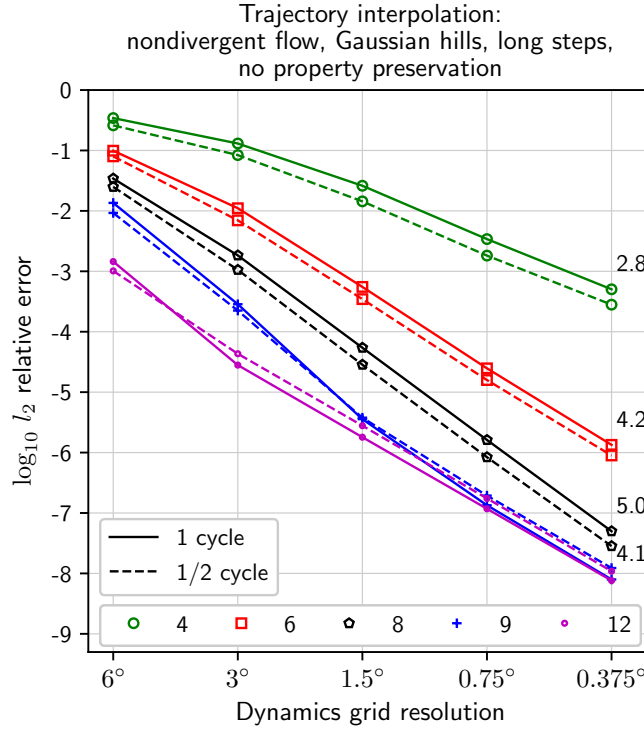


Figure 3. Comparison of relative errors calculated at the test simulation’s midpoint time of 6 days (1/2 cycle, dashed lines) and endpoint time of 12 days (1 cycle, solid lines). Each number at the right side of the plot is the empirical OOA computed using the final two points of the 1-cycle result.

with the natural GLL basis provides a midpoint solution much more accurate than the time-stepped case, thus serving as an appropriate reference.

Figure 3 shows the results. The dashed curves show the error errors measured at the midpoint, half of a cycle of the 12-day problem; the solid curves, the usual endpoint. The numbers at the right side of the plot show empirical OOA computed using the final two points of the one-cycle results. Numbers for the half-cycle curves are omitted because each pair of curves has almost exactly parallel lines for resolution at least as fine as 0.75° . In addition, the empirical OOA for $n_p^t = 12$ is omitted because the curves nearly overlap those for $n_p^t = 9$. For fine enough resolution, all curves should converge with OOA have empirical OOA limited to 4 because the tracer-grid trajectories are interpolated using the $n_p = 4$ natural interpolant from the dynamics grid, but for $n_p^t = 8$, the full convergence regime is not reached in this plot, thus giving OOA 5 for $n_p^t = 8$. The $n_p^t = 4$ curves have OOA less than 4 due to a spatial OOA limit of 2 in the full convergence regime. We see OOA at about 4 for $n_p^t = 9$ and 12. The $n_p^t = 9$ curve shows at coarse resolution higher OOA, governed by the spatial error, and then a drop to 4 as the temporal error becomes dominant. Importantly, the half- and full-cycle errors are very close for each value of n_p^t , demonstrating that the endpoint error metrics are valid measurements for the Islet method.

1125 4.1.2 Empirical order of accuracy

Empirical verification of the order of accuracy of the Islet GLL bases. Each number at the right side of the plot is empirical OOA computed using the final two points of the l_∞ curve.

Figure ?? empirically verifies the OOA of the Islet bases. The test uses the nondivergent flow, Gaussian hills IC, and short time steps. To expose the OOA of the basis, this test does not use TTPR — the trajectories are computed nearly exactly at each quadrature point, and thus $n_p^v = n_p^l$ — and property preservation is turned off. Errors are reported in the l_2 (solid lines) and l_∞ (dashed lines) norms. The numbers on the right side of the plot are empirical OOA calculated using the final two points of the l_∞ -norm curves; the values for the l_2 -norm curves are about the same. The $n_p^l = 12$ curve has one fewer element-grid refinement point because the 0.375° one is slightly influenced by the machine-precision limit on relative accuracy. Each empirical OOA is at least as large as, but not much larger than, the expected OOA listed in Table A1.

1135 4.1.2 Accuracy heuristic

l_2 norm on the nondivergent flow problem using basis \mathcal{N}_{n_p} vs. $a_2(\mathcal{N}_{n_p})$, for a large number of t.p.s. bases and $n_p = 6$ to 10. The legend lists the marker type for each n_p . Large red circles outline the bases in Table A1. The configuration uses the Gaussian hills IC and no property preservation. Same as Fig. ?? except that the configuration uses the cosine bells IC with property preservation.

To select Islet bases for each value of n_p^l , recall that we used accuracy heuristics a_i , $i \in \{1, 2, \infty\}$, to choose accurate bases, and the perturbed uniform mesh metric to filter out bases that might be unstable in practice. The accuracy heuristics are functions of the basis data only, not any specific problem. Figures ?? and ?? plot accuracy on a validation problem vs. the value of the a_2 heuristic for a large number of bases. Each point in a scatter plot corresponds to a basis. The y -axis is accuracy of bases when used to simulate the nondivergent flow at resolution 1.5° , in two different cases: Gaussian hills with no property preservation (Fig. ??) and cosine bells with property preservation (Fig. ??). The x -axis is the value of a_2 for the basis. The marker pattern corresponds to n_p^l . For each n_p^l , one red circle outlines the basis that is in Table A1. The a_2 accuracy heuristic is a good predictor of relative accuracy on the validation problems. We chose the bases in Table A1 using the heuristics without reference to validation problem results, as we do not want accuracy on any particular problem to influence the selection of recommended bases.

1150 4.1.2 Accuracy data for other standard configurations

The next figures show accuracy for standard configurations used in TR14. ~~Figure titles provide the test configuration details.~~ Although we explain how each figure can be compared with corresponding ones in TR14, these figures also stand on their own as simply convergence plots for various test cases.

Figures 4 and 5 can be compared with Figs. 1 ~~, and~~ 2 in TR14. They evaluate error on an infinitely smooth IC. The Islet method with a high-order basis compares extremely favorably with the methods in TR14. For example, the most accurate shape-preserving method in TR14 for the nondivergent flow with the Gaussian hills IC is HEL-ND-CN1.0, by a substantial

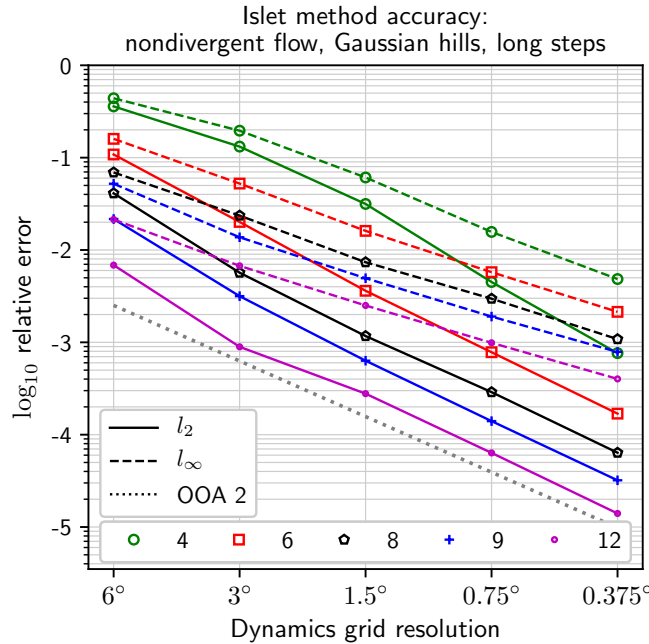


Figure 4. Accuracy diagnostic. Compare with Figs. 1, 2 in TR14.

margin (cyan curves in Fig. 1, bottom right, of TR14). The $n_p^t = 12$ Islet scheme with the long time step, Fig. 4, is approximately three times more accurate than HEL-ND-CN1.0 in the l_2 norm at resolution 0.375° and approximately twice as accurate at resolution 3° . Yet HEL-ND is, quoting TR14, an “unphysical” method. It is run for comparison with the practically useful HEL scheme. After HEL-ND, the next most accurate method in the l_2 norm at 0.375° resolution is CSLAM-CN5.0. The Islet method with the long time step, the same as that of CSLAM-CN5.0, is at least as accurate for $n_p^t \geq 8$. With the short time step, the same as that of CSLAM-CN1.0, the Islet method is at least as accurate as CSLAM-CN1.0 for $n_p^t \geq 6$. At 3° resolution, no method other than HEL-ND-CN1.0 provides l_2 norm below 10^{-2} ; the Islet method does for $n_p^t \geq 8$ with the long time step and $n_p^t \geq 10$ (only $n_p^t = 12$ is shown) with the short time step.

Now, we must remind the reader that, in this article, we use nearly exact trajectories on the dynamics grid because the description of practical trajectory methods is outside the scope of this article. However, first, many of the schemes in TR14 use nearly exact trajectories, as well, e.g., CSLAM. Second, when using a practical trajectory algorithm, highly accurate, even if not exact, trajectories are possible because a trajectory over a tracer time step can be computed from multiple dynamics time steps. Thus, the use of exact trajectories is only slightly unrealistic. Third, the diagnostic values in Sect. 4.1.3, 4.1.4, and 4.3.2 are roughly independent of temporal errors.

Figures 6 and 7 provide data that can be compared with the top panel of Fig. 3 in TR14. They evaluate error on a C^1 IC. The horizontal dash-dotted line provides the relative l_2 -error-norm value of 0.033 by which the “minimal resolution” diagnostic value is determined; the coordinate of the intersection between the l_2 -norm curve and this reference line is the value. A larger

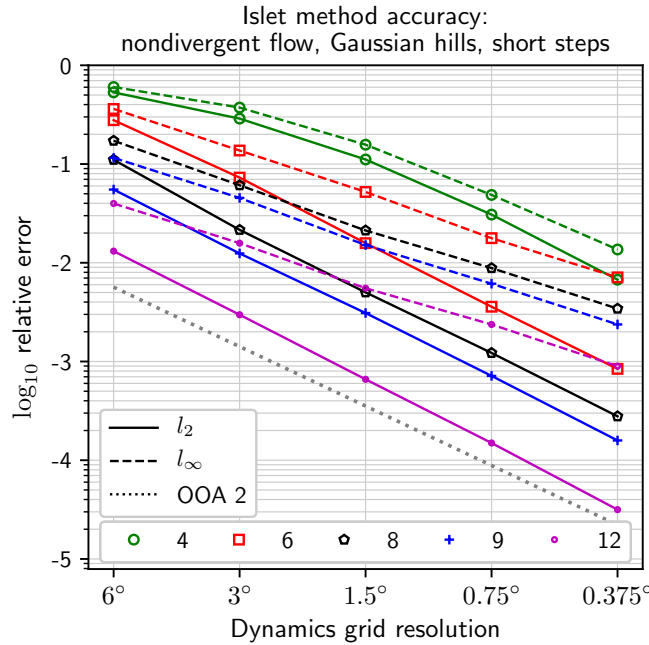


Figure 5. Accuracy diagnostic. Compare with Figs. 1, 2 in TR14.

value is better. For example, with a long time step, for $n_p^t = 8$, this value is a little coarser than 3° ; for $n_p^t = 12$, approximately 6° .

The Islet method with a high-order basis compares extremely favorably with the best of the methods in TR14. For example, the most accurate shape-preserving method in TR14 for the nondivergent flow with Gaussian hills IC is HEL-ND-CN1.0 (cyan curves in Fig. 1, bottom right, of For comparison, no model in TR14). The $n_p^t = 12$ Islet scheme with long time steps, Fig. 4, has even more accuracy than HEL-ND-CN1.0 in the l_2 norm reports a value larger than 2.5° .

Figures 8 and 9 provide data that can be compared with the top two panels in Fig. 16 of TR14, given the minimum resolutions plotted in Fig. 3 of TR14. They are like Figs. 6 and 7, but here the divergent flow is used.

4.1.3 Filament diagnostic

Figure 10 shows results for the filament diagnostic described in Sect. 3.3 of TS12, to compare with Fig. 5 in TR14. The two dynamics grid resolutions are as prescribed in TR14. We used the code distributed with TS12 to compute the results. The diagnostic uses the nondivergent flow with and the cosine bells IC. In this test, the midpoint solution is analyzed to determine the quality of the filamentary structure. See Fig. 13 for an illustration of the filamentary structure at the simulation midpoint, although with the slotted cylinders IC. For each value of the possible value τ of the tracer mixing ratio at the initial time, $\tau \in [0.1, 1]$, the area over which the mixing ratio is at least τ at the midpoint time is computed. For the cosine bells IC, $\tau \in [0.1, 1]$. The diagnostic is then this area divided by the correct area, which for any nondivergent flow is the area at the

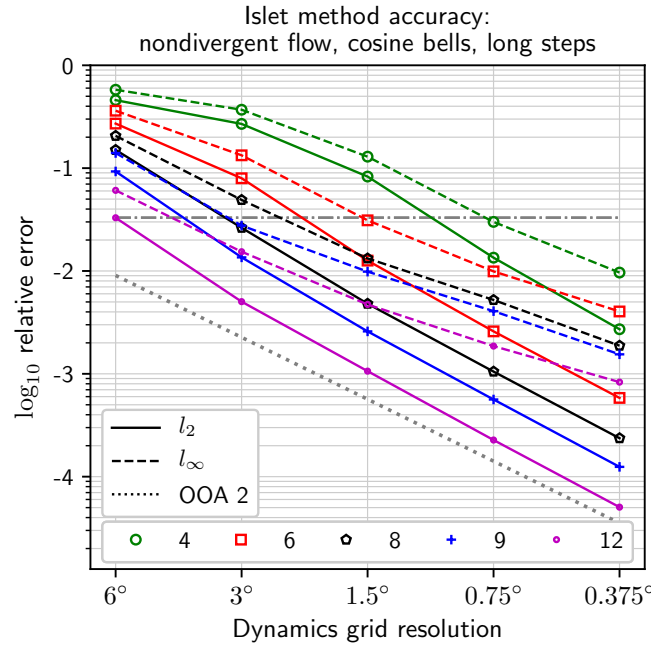


Figure 6. Accuracy diagnostic. Compare with Fig. 3 in TR14.

1190 initial time. The perfect diagnostic value is 100% for all $\tau \in [0.1, 1]$ and 0 otherwise. In each plot in Fig. 10, the x -axis is τ and the y -axis is the diagnostic value. The diagnostic is computed for the dynamics-grid resolutions and time step lengths listed in the legend. Note that the y -axis limits are tighter with increasing n_p^t .

A subtlety with this diagnostic is that the area calculation must use the quadrature method of the discretization. On the dynamics grid, the resulting curve is noisier than is implied by the underlying solution on the tracer grid. Thus, in Fig. 10, we
1195 show the diagnostic as computed on the dynamics grid in the top row of plots; on the tracer grid, in the bottom row.

There is no summary number that can be compared directly with the results in Fig. 5 of TR14, but, visually, the curves for $n_p^t \geq 8$, the resolution 1.5°, and on the tracer grid are among the best of those in TR14 at the resolution 1.5°.

4.1.4 Mixing diagnostic

Figures 11 and 12 show results for the mixing ~~diagnostic~~ diagnostics described in Sect. 3.5 of TS12, to compare with ~~Figures~~
1200 Figs. 11–14 in TR14. ~~Like the filament diagnostic, the analysis is done at the solution midpoint rather than the endpoint. Also as for the filament diagnostic, we used the code distributed with TS12 to compute the results. The problem is~~ Again the two dynamics grid resolutions are as prescribed in TR14.

The test uses the nondivergent flow with two ICs:- The diagnostics assess preservation of the nonlinear correlation of two tracers; one mixing ratio is a nonlinear function of the other. The mixing ratio of each tracer is on an axis, cosine bells on
1205 the x -axis ~~of each plot~~, and the correlated cosine bells field on the y -axis. Each dot ~~in a plot~~ is a grid-point sample from the

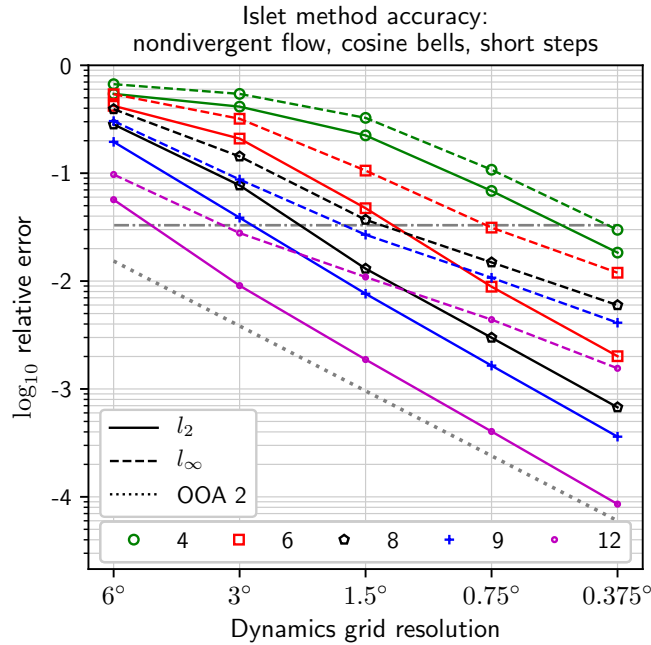


Figure 7. Accuracy diagnostic. Compare with Fig. 3 in TR14.

dynamics grid of the two mixing ratios. A perfect diagnostic has the red dots all on the convex-upward black reference curve. n_p^t and Like the filament diagnostic, the analysis is done at the solution midpoint rather than the endpoint. Also as for the filament diagnostic, we used the code distributed with TS12 to compute the results. n_p^t and the time step length is are printed in each plot. Diagnostic The diagnostic values l_r and l_u are, explained in a moment, are also printed in each plot; the parenthesized “(v)” suffix means the value is measured on the dynamics grid, while the un-suffixed values are measured on the tracer grid. Because we show results only for simulations with property preservation on, $l_o = 0$ always.

At the initial time, all points are on the curve. Perfect nonlinear correlation corresponds to staying on the upper curve as the simulation proceeds. Points that drop into the convex hull of the starting points can be interpreted as physical mixing of air parcels, since such mixing results in a linear combination of two points on the curve. The diagnostic value l_r measures this type of error; smaller is better. In Figs. 11–14 in TR14, the smallest value of l_r at 1.5° among the property-preserving methods is 2.15×10^{-4} , by the UCISOM-CN5.5 method, except for a value of 0 by HEL-ND, which, again, cannot be used in practice. For the long time step, the Islet method gives at least as small a value for $n_p^t \geq 6$; for the short time step, $n_p^t \geq 8$.

Values outside the triangle are overshoots, possible only if the method is not strictly shape preserving. Since our method is, this diagnostic value, l_o , is always 0 and is not displayed in the figures.

Values outside the convex hull cannot be described as physical mixing of parcels and thus are purely numerical artifacts of a method. The corresponding diagnostic is l_u , and again smaller is better. This diagnostic is more difficult to compare than l_r because very dissipative methods tend to have a large value of l_r and consequently a very small value for l_u . In contrast, a very

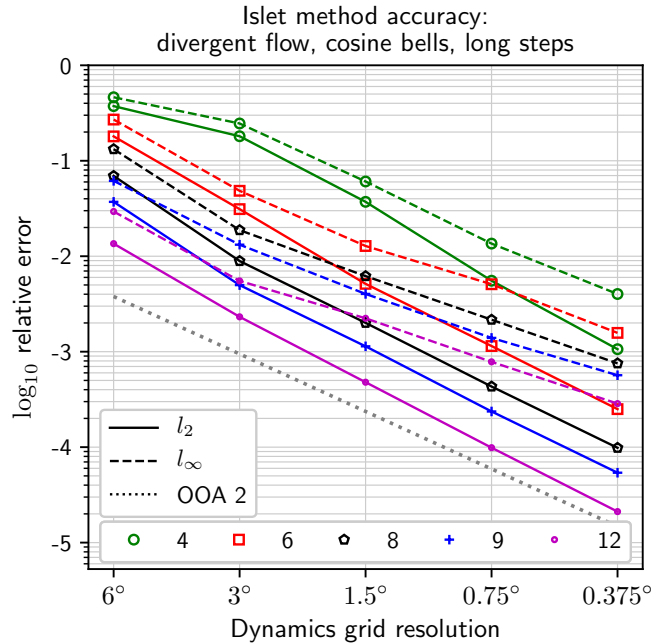


Figure 8. Accuracy diagnostic. Compare with Fig. 16 in TR14.

accurate method, for which l_r is small, can have a larger l_u value than a very dissipative method. One means of comparison is to consider the best l_u values among methods that obtain, say, $l_r < 5 \times 10^{-4}$. In Figs. 11–14 in TR14, the smallest value of l_u at 1.5° under this restriction is 0, obtained by the HEL-CN1.0 and so we do not print those values. We omit a description of these scalar diagnostics and refer the reader to TS12HEL-CN5.5 methods. These HEL variants are practically usable, unlike HEL-ND, and are designed to preserve tracer correlations exactly. Other than the HEL methods, the next best value is 4.80×10^{-5} , again by the UCISOM-CN5.5 method. For both the long and short time steps, at 1.5° , the Islet method gives at least as small a value for $n_p \geq 8$. However, even with the constraint on l_r , comparison is not straightforward, as the UCISOM methods are not strictly shape-preserving and so have $l_o > 0$.

4.2 Slotted cylinders

We observe that in both the filament and mixing diagnostics of TS12, $n_p^l \geq 6$ gives excellent results; $n_p^l = 12$, nearly perfect. Figures 13 and 14 show solution quality using latitude-longitude images and further underscore these observations. The problem is the nondivergent flow with the slotted cylinders IC, at resolutions 1.5° and 0.75° , with the long and short time steps. The text in the individual images in Fig. 14 provides normwise accuracy at the end of one cycle and deviation from the initial extrema. $\phi_{\min} \geq 0$ and $\phi_{\max} \leq 0$ are consistent with no global extrema violation. Compare Fig. 13 with Figs. 7–10 in TR14 and both figures with Fig. 7 in TS12. Although TR14 does not provide error norm values for this problem, those in

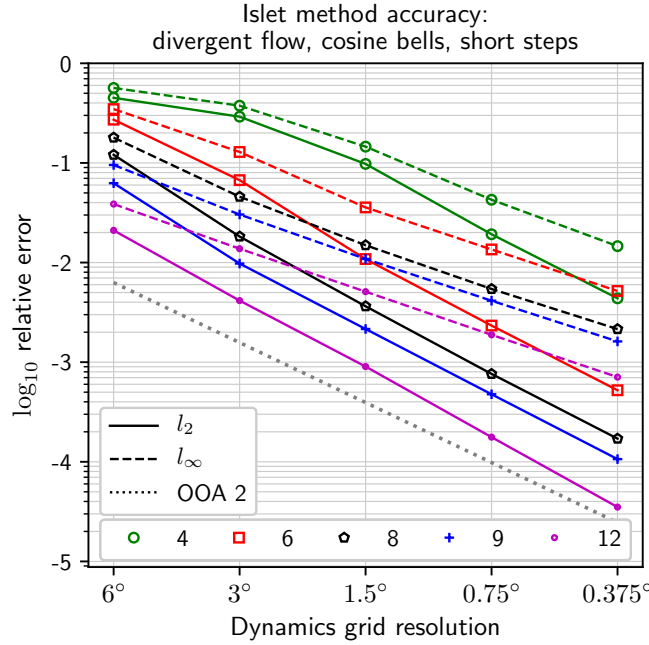


Figure 9. Accuracy diagnostic. Compare with Fig. 16 in TR14.

Fig. 7 of TS12 can be compared with the Islet method's values at 1.5° resolution and the long time step (left column of Fig. 14). The Islet method's values of l_2 , l_∞ , ϕ_{\min} , and ϕ_{\max} are at least as good as those in Fig. 7 of TS12 for $n_p^t \geq 6$.

1240 4.3 Source ~~term~~term

Now we move to validation problems that include a source term: $f \neq 0$ in Eq. (3).

4.3.1 Accuracy

The first test validates the property-preserving remaps between the physics and tracer grids. The test is constructed as follows. Two ~~tracer~~tracer mixing ratios, a source ~~tracer~~ $q_1 = s$ and a manufactured tracer $q_2 = m$, are paired. ~~$m(0)$~~ At time $t = 0$, $m(x, t)$ is set to 0 . ~~0~~ , where x is position on the sphere. A tendency Δm is applied to m on the physics grid: $\Delta m(t) = -[\cos(2\pi(t + \Delta t)/T) - \cos(2\pi t/T)]s(t)/2$. ~~$\Delta m(x, t) \equiv -[\cos(2\pi(t + \Delta t)/T) - \cos(2\pi t/T)]s(x, t)/2$~~ , so that the exact solution is $m(t) = (1 - \cos(2\pi t/T))s(t)/2$. ~~$m(x, t) = (1 - \cos(2\pi t/T))s(x, t)/2$~~ and, in particular, $m(T/2) = s(T/2)m(x, T/2) =$ To compute the tendency, the state $s(t)$ ~~$s(x, t)$~~ must be remapped to the physics grid. Thus, ~~this test depends on accuracy in the results for this test depend on~~ both grid-transfer directions. We measure the error at time $T/2$, on the dynamics grid as usual, as in Fig. 3. Figure 15 shows the results. We run this test with $n_f = n_p^t$ ~~$n_f = n_p^t$~~ (dash-dotted lines) and $n_f = 2$ ~~$n_f = 2$~~ (dashed lines). The solid lines show the error in $s(T/2)$ ~~$s(x, T/2)$~~ as a reference. The dotted line provides the OOA-2 reference. We see that when $n_f = n_p^t$ ~~$n_f = n_p^t$~~ , the errors are nearly the same as those for $s(T/2)$; ~~for $n_p^t = n_f = 4$~~ $s(x, T/2)$; ~~for $n_p^t = n_f = 4$~~ $s(x, T/2)$; ~~for $n_p^t = n_f = 4$~~ $s(x, T/2)$.

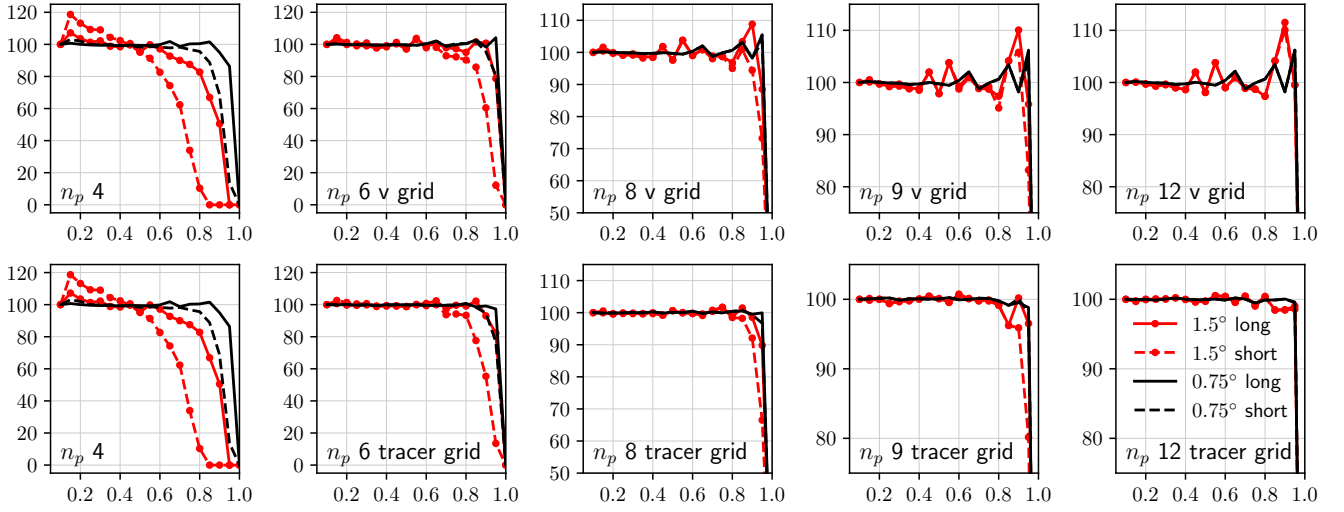


Figure 10. Filament diagnostic, following Sect. 3.3 of TS12. Compare with Fig. 5 in TR14. The top row shows the diagnostic measured on the $n_p^v=4$ $n_p^v=4$ dynamics grid; the bottom row, on the tracer grid. The legend describes the dynamics-grid resolution and the time step length. The prescribed validation problem is the nondivergent flow with cosine bells IC. Property preservation is on. The x -axis is τ , the mixing ratio threshold. The y -axis is the percent area having mixing ratio at least τ relative to that at the initial time.

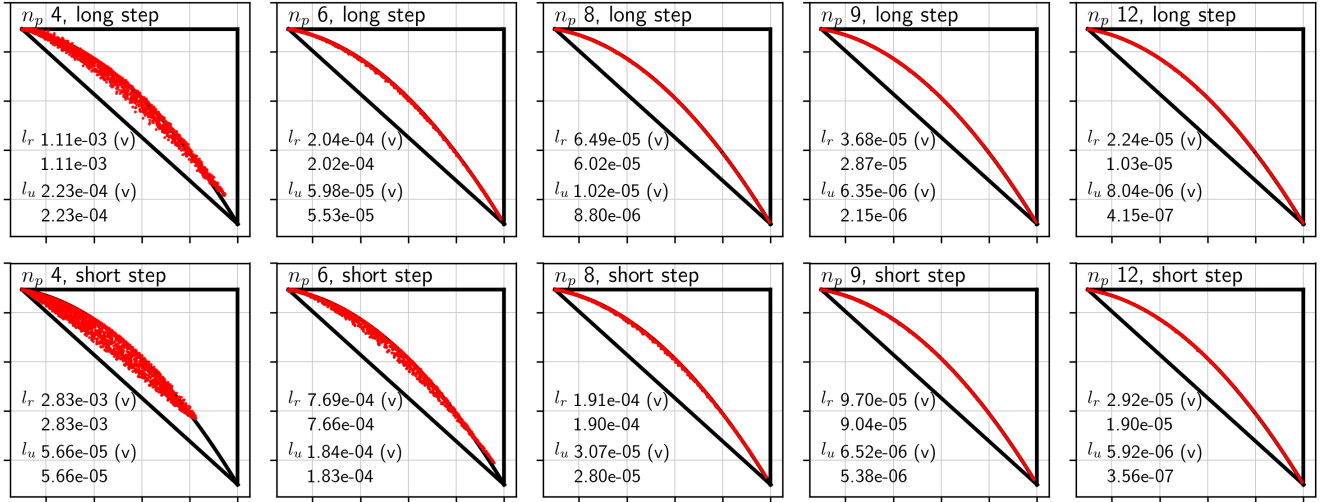


Figure 11. Mixing diagnostic, following Sect. 3.5 of TS12. Compare with Figs. 11–14 in TR14. This figure shows results for dynamics-grid resolution of 1.5° . l_o is exactly 0 in all cases because shape preservation is on, and so is not shown. See text for further details.

the curves overlap at the resolution of the plot. As one expects, when $n_T=2$ $n_T=2$, the error in $m(t)$ $m(x,t)$ is much larger than in $s(T/2)$ $s(x,T/2)$, but the OOA remains 2.

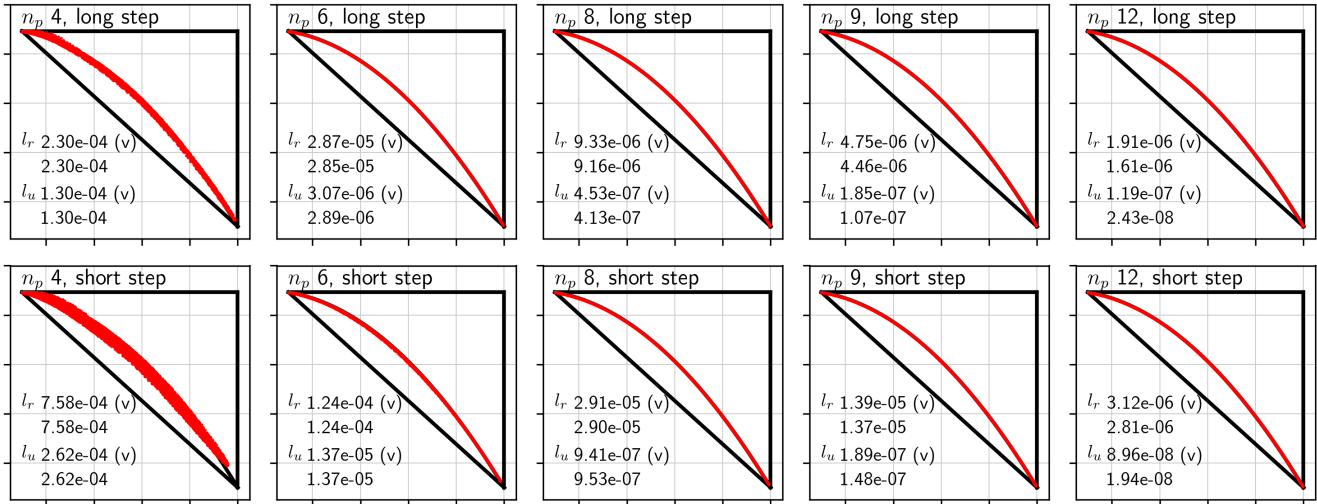


Figure 12. Same as Fig. 11 but with dynamics-grid resolution 0.75° .

1255 4.3.2 Toy chemistry diagnostic

The toy chemistry validation problem is described in Lauritzen et al. (2015), subsequently TC15. The problem consists of two tracer mixing ratios, $q_i = X_i$, $i = 1, 2$, that interact according to chemical kinetic equations that are nonlinear in one of them: $DX_i(\mathbf{y}, t)/Dt = f_i(\mathbf{y}, X_1, X_2)$, where \mathbf{y} is the spatial coordinate on the sphere. (We use \mathbf{y} here to avoid overlap with the species symbol, X , that is used in TC15.) The tracers are composed of a monatomic and a diatomic molecule, respectively, of the same atomic species. The reactions are extremely sensitive to solar insolation. The sun's position is held fixed with respect to the grid. As a result, the largest-scale spatial pattern one sees in the fields is the boundary dividing nonzero (day) and zero (night) solar insolation, the solar terminator; this boundary is particularly visible in the right image of Fig. 17, a figure that we will describe subsequently. The ICs are designed so that the sum over the atomic mixing ratio at each point in space is a constant, \bar{X}_T . The source terms have this property, too, since they model chemical reactions. Thus, in the exact solution, \bar{X}_T is maintained at every point in space and time. Let X_T , without a bar, be the corresponding measured quantity. The toy chemistry diagnostics are then $c_2(t)$, the l_2 norm of $X_T - \bar{X}_T$ at time t ; normalized by \bar{X}_T , and $c_\infty(t)$, the same but for the l_∞ norm.

As explained in the context of equation 14 in TC15, any advection operator that is *semi-linear* will produce a perfect diagnostic value of 0 when using exact arithmetic. Linear operators are semi-linear; the CEDR algorithms we use are, as well, as explained in Bradley et al. (2019); and a composition of semi-linear operators is, too. Thus, the Islet method is semi-linear, and deviation from 0 in the diagnostic values is wholly due to the effects of finite precision arithmetic.

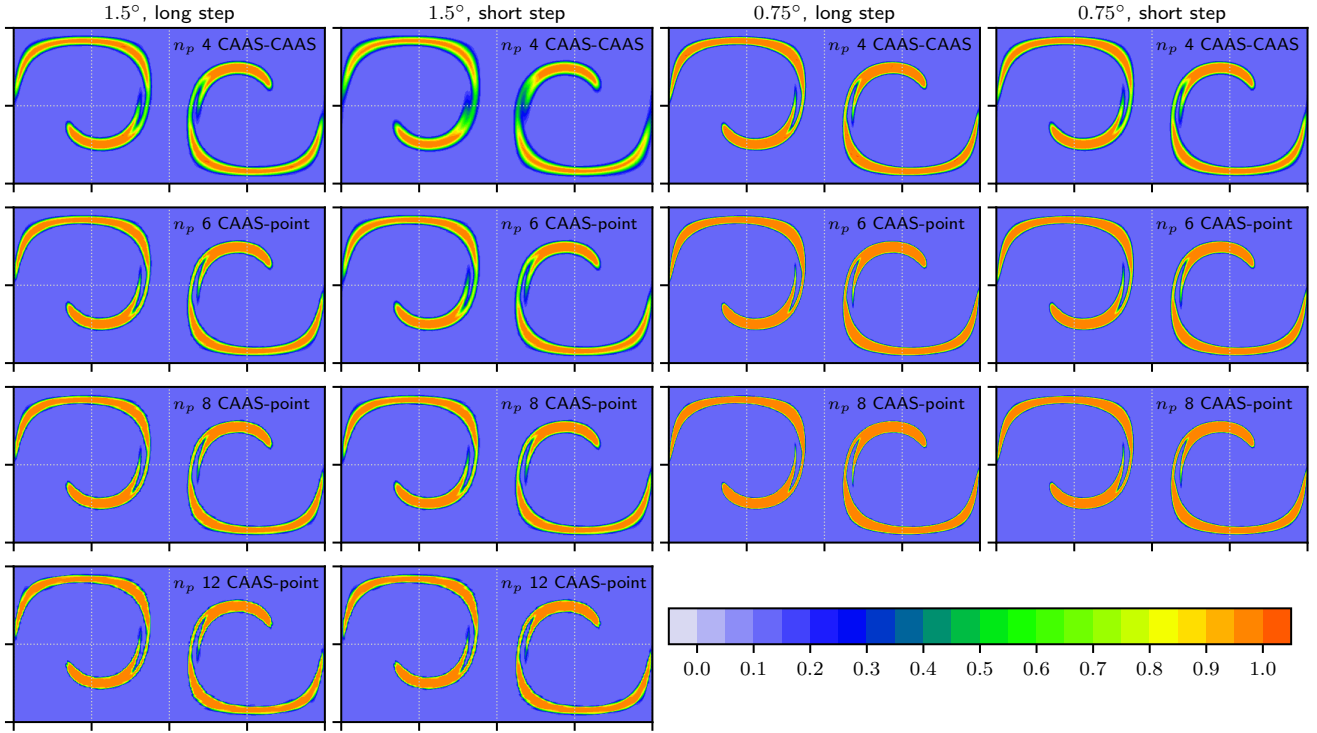


Figure 13. Images of the slotted cylinders IC advected by the nondivergent flow at the simulation’s midpoint. Each column corresponds to a spatial resolution and time step length configuration, as stated at the top of each column. Each row corresponds to a particular value of $n_p^l n_p^r$, as stated in the text at the top-right of each image. We omit $n_p^l = 12, n_p^r = 12$ results for the 0.75° resolution because they are essentially identical at the resolution of the figure to the $n_p^l = 8, n_p^r = 8$ images.

We compute the diagnostics, as usual, on the dynamics grid. Following the Islet tracer transport method described in Sect. 3.1, the source terms are computed on the physics grid using states remapped from the tracer grid, and then the computed tendencies are remapped to the tracer grid.

1275 It is already known that the Eulerian spectral element tracer transport method yields poor values for this diagnostic due to finite-precision effects of the limiter (Lauritzen et al., 2017). ~~Islet-~~The Islet method with property preservation using CAAS-CAAS does, as well. ~~In-~~Again, in exact arithmetic, each of these methods would produce perfect values. The poor diagnostic values are due to quickly accumulating machine-precision ~~truncation-round-off~~ errors that break semi-linearity in finite precision. The interaction of the chemistry source term with exact bounds in element-local limiter applications is responsible for
1280 this fast accumulation. In contrast, ~~Islet-with-property-preservation-the Islet method~~ using CAAS-point and relaxed-bound, element-local CAAS applications produces good diagnostic values ~~because the-~~. The relaxed bounds in the element-local part make unnecessary many of the mixing ratio adjustments that lead to loss of semi-linearity in finite precision. Recall that CAAS-point, applied at the end of the step, imposes the exact bounds, so at the end of a time step, shape preservation still holds to

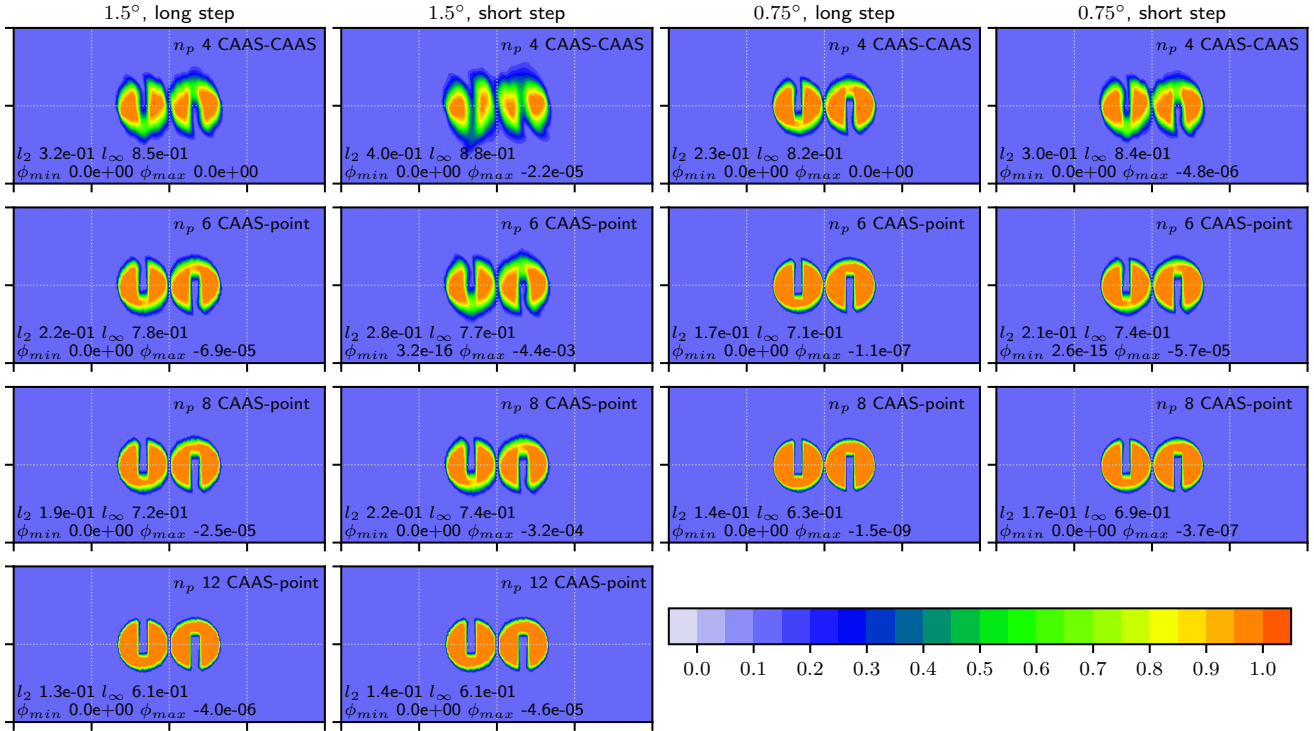


Figure 14. Same as Fig. 13 but for the simulation final point. Error measures are printed at the bottom-left of each image; see text for details.

machine precision. Clips to bounds must still occur, but adjustments to other grid points to compensate are smaller because the adjustments are spread over many more grid points.

Figure 16 shows the diagnostic values for the case of nondivergent flow, 1° dynamics-grid resolution, and a 30-minute time step, where these configuration details are prescribed in TC15. For the $n_p^t = 4$ $n_p^f = 4$ case, we use CAAS-point rather than CAAS-CAAS as previously, since we already know that CAAS-CAAS will produce poor values. The diagnostic is usually plotted over the course of one cycle (12 days) of the flow, but it is useful to view it over multiple cycles. Figure 16 shows ten cycles on the x -axis, for a total of 120 days. The y -axis is the diagnostic value. Solid lines plot $e_2 c_2(t)$; dashed, $e_\infty c_\infty(t)$. Markers are placed on the curves at the start of each cycle to help differentiate the curves. In each case, $n_f = n_p^t n_f = n_p^f$. We choose this value of n_f because the toy chemistry source term has nearly a singularity—an extremely large gradient at the terminator—the terminator is clearly seen in Fig. 17—and thus it makes sense to compute the physics tendencies at high spatial resolution. The $n_p^t = 4$ $n_p^f = 4$ case with CAAS-point is greatly improved relative to the Eulerian spectral element results shown in Fig. 7 of TC15, even after ten cycles instead of the one cycle shown in that figure. For $n_p^t > 4$ In Fig. 7 of TC15, $c_2 \approx 10^{-2}$ and $c_\infty > 10^{-1}$ at the end of one cycle, compared with approximately 10^{-7} and 10^{-5} , respectively, for $n_p^t = 4$ at the end of ten cycles. For $n_p^t > 4$, the growth in error is very small, with e_2 less than 10^{-10} $c_2 < 10^{-10}$ through ten cycles.

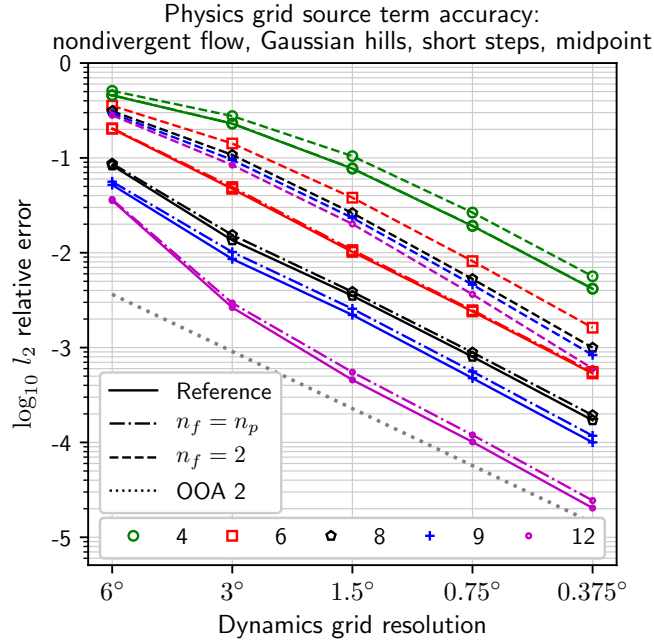


Figure 15. Validation of the remap of tendencies from physics to tracer grids and state from tracer to dynamics grids. See Sect. 4.3 for a description of the problem.

To illustrate what these diagnostics measure, Figure 17 shows latitude-longitude images of the monatomic tracer at the end of the first cycle for $n_p^t = n_f = 4$ $n_p^t = n_f = 4$ with CAAS-CAAS (left) and $n_p^t = n_f = 8$ $n_p^t = n_f = 8$ with CAAS-point (right). Note that the images in TS12 and TR14 are plotted with longitude ranging from 0 to 2π ; those in TC15, from $-\pi$ to π . In our latitude-longitude figures so far, we have chosen the convention used in TS12 and TR14, and we continue to use it in these toy-chemistry images. Thus, these images are circularly shifted horizontally by half the image width relative to those in TC15. The globally extremal tracer values are printed in the upper-right quadrant of each image. The correct maximum is 4×10^{-6} and the correct minimum is at least 0. The right image is free of noise and satisfies these bounds. The left image shows substantial noise, as we expect when using exact bounds in the local property preservation problems, and consistent with previous observations about spectral element transport. Other than noise and some filaments that grow from the noise, the two images are qualitatively similar. Figure 18 shows images in the same format, but the quantity is now $(X_T - \bar{X}_T)/\bar{X}_T$ at the end of the first cycle. The correct value is 0 everywhere. In the right image, the pointwise relative error is a little better than 10^{-11} , consistent with the l_∞ -norm diagnostic value for $n_p^t = 8$ $n_p^t = 8$ at the end of the first cycle in Fig. 16.

5 Performance results

4.1 Communication volume

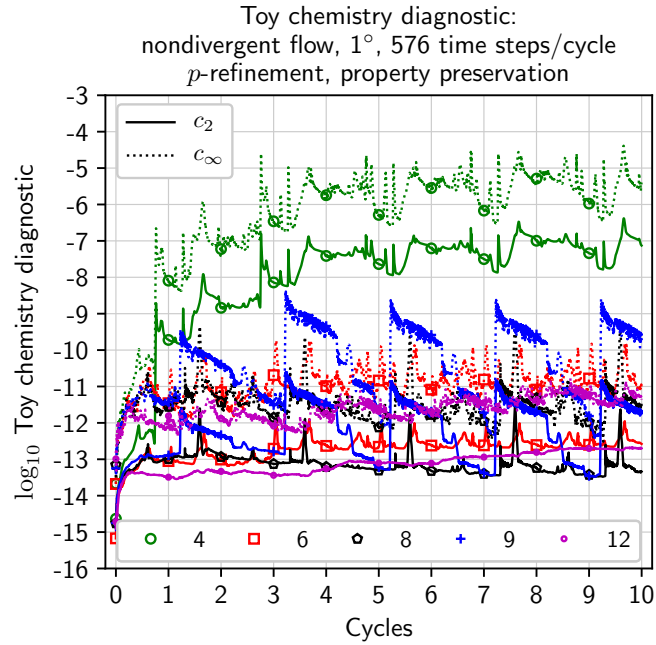


Figure 16. Toy chemistry diagnostic values as a function of time for ten cycles of the nondivergent flow. Time is on the x -axis and measured in cycles. Diagnostic values c_2 (solid lines) and c_∞ (dashed lines) are on the y -axis. Markers as listed in the bottom legend are placed at the start of each cycle to differentiate the curves.

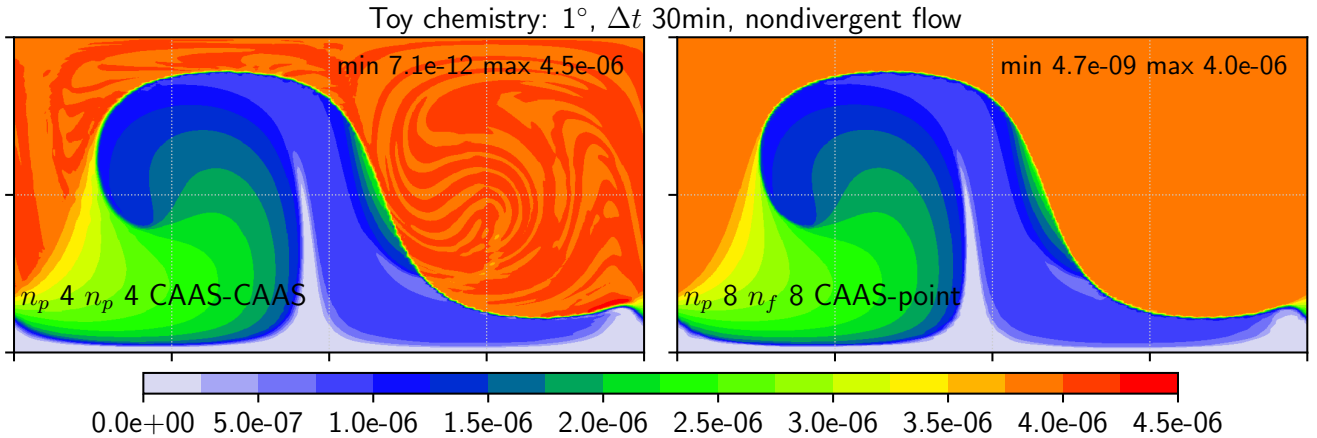


Figure 17. Images of the monatomic tracer at the end of the first cycle. Text at the lower left of each image states the configuration. Text at the upper right reports global extremal values.

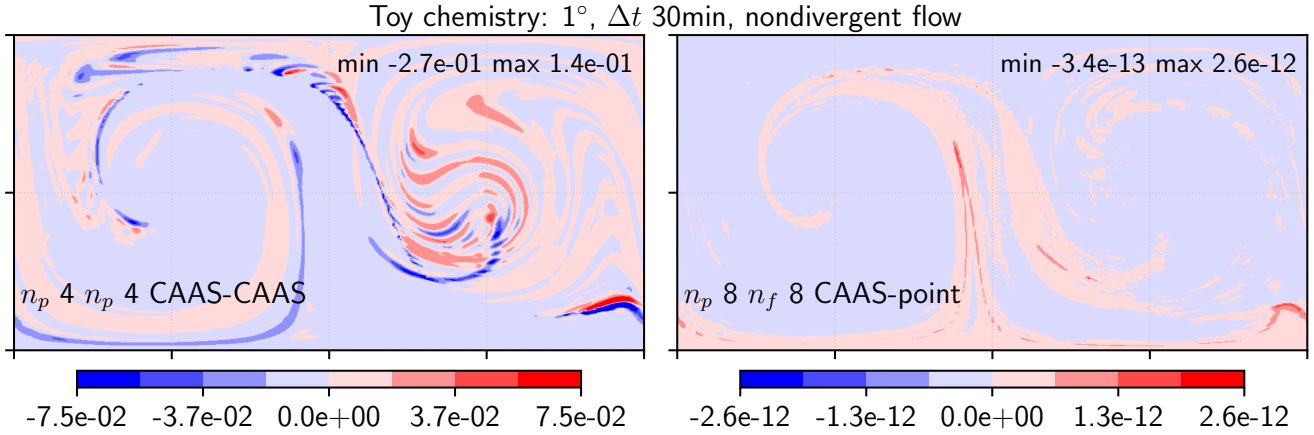


Figure 18. Same as Fig. 17, but now the images are of $(X_T - \bar{X}_T)/\bar{X}_T$.

Communication volume, in number of real scalars transmitted in q -messages per tracer per element per time step (y -axis) vs. time in days of the simulation (x -axis), in the case of one element per process, for the nondivergent flow, with long (left) and short (right) time steps. Statistic and n_p^l line patterns are stated in the legends.

1315 Following the discussion in Sect. ??, Figure ?? plots the number of real scalars transmitted in q -messages per tracer per element per time step (y -axis) vs. time in days of the simulation (x -axis), in the case of one element per process. The configurations use nondivergent flow, 1° resolution, long and short time steps, and $n_p^l \in \{4, 8, 12\}$. Statistics include the maximum (dashed line) and median (solid line) over all target elements. A dotted horizontal line provides the reference value $(n_p^l)^2$. As discussed in Sect. ??, because the volume of extrema data is almost independent of n_p^l , the measured volume

1320 decreases relative to the reference $(n_p^l)^2$ with increasing n_p^l .

4.1 GPU performance

Performance comparison of SL transport with $n_p^v = n_p^l = 4$ vs. Eulerian transport in the E3SM Atmosphere Model's dynamical core on the Summit supercomputer. The x -axis is number of NVIDIA V100 GPUs on Summit used in a run; the y -axis is dycore throughput reported in simulated years per wallclock day (SYPD). The black curves are for Eulerian transport; the red, for SL. Dashed lines are for 40 tracers; solid and the dotted black line, for 10. A number above a data point reports the y -value of that point.

E3SM Atmosphere Model's (EAM) nonhydrostatic dynamical core, HOMME (Dennis et al., 2005, 2012; Taylor et al., 2020), has been ported to C++ to run on GPU-based supercomputers (Bertagna et al., 2020), and performance results are reported in Bertagna et al. (2020) for the NGGPS benchmark (Whitaker, 2016; Michalakes et al., 2016). At that time, only the original

1330 Eulerian-tracer transport module was available. In related work, a quasiuniform 3.25km convection-permitting configuration of EAM, called SCREAM, has been developed (Caldwell et al., 2021); this work uses SL transport with $n_p^v = n_p^l = 4$. Other

than this difference, Bertagna et al. (2020) and Caldwell et al. (2021) use the same dycore configurations. We omit description of these configuration details because they are not relevant to this article; see the cited articles for details.

Our SL transport implementation in HOMME does not yet permit $n_p^l > n_p^v$, but otherwise it follows, in particular, the communication procedures described in Sect. ?? . We re-ran the benchmark in Bertagna et al. (2020) with SL transport configured as in Caldwell et al. (2021) and report results in Fig. ?? . The x -axis is number of NVIDIA V100 GPUs on the Oak Ridge Leadership Computing Facility's Summit supercomputer used in a run; Summit has just over 27,600 V100 GPUs. The y -axis is dycore throughput reported in simulated years per walleck day (SYPD). Some data points have SYPD numbers above the point for additional precision.

First, we established that we could roughly reproduce the results in Bertagna et al. (2020). The black dotted line shows the data we obtained using Eulerian transport; compare this curve with the data from Bertagna et al. (2020), plotted as the black solid line with circular markers. Our results are a little less than 10% slower, which is acceptable given different compiler and MPI versions, different job profiles running at the same time, and a slightly different code base. We then switched to using SL transport and obtained the solid red line with square markers. These simulations used ten tracers. We repeated them with 40 tracers, the number used in E3SMv1 and v2 models; the corresponding curves are dashed. Comparing the new SL data with the data in Bertagna et al. (2020), SL transport speeds up the nonhydrostatic dycore by a factor of over 1.4 when 10 tracers are transported and a factor of 2.8 when 40 tracers are transported.

5 Conclusions

We have presented a set of compact bases, the Islet bases, for very accurate interpolation semi-Lagrangian element-based tracer transport. We then described algorithms described the Islet method, a property preserving tracer transport method to support a three-grid atmosphere model, one with a shared element grid but separate subelement grids for physics parameterizations, dynamics, and tracer transport ~~using the Islet bases~~. This configuration permits the modeler to create a dynamics grid with a tolerable CFL-limited time step independent of the other two subcomponents, while physics parameterizations and tracer transport can run at resolutions potentially substantially higher. The shared element grid minimizes communication during remaps between grids, and almost all operations are local to each element, making the Islet method extremely efficient.

Future work includes the following. First, in Sect. ?? we described an infinite set of potential bases, but then in Sect. ?? we restricted our attention to a finite set of these. It is possible that there are much better bases outside of the subset we considered. Searching for these is one line of work. Section 4 presented a number of validation problems with diagnostics assessing accuracy, order of accuracy, numerical mixing, filament preservation, and nonlinear correlation preservation. The corresponding figures and diagnostic values can be compared directly with those of several other methods. The Islet method performs well in a number of detailed comparisons.

Second Possible future work includes the following. First, because all operations except the DSS are either local to the element or act on element-level scalars, both the tracer and the physics grids permit various types and subsets of spatially and temporally adaptive and tracer-dependent refinement and derefinement, possibly in combination with already existing E3SM

regionally refined models (RRM). ~~In the vertical direction, we believe there is an opportunity to combine Islet with FIVE (Yamaguchi et al., 2017), or a method like it, to transport highly resolved vertical data while still running the dynamics solver on the coarser background vertical grid.~~ Future work could develop procedures to permit spatially and temporally varying n_p^t and n_f values and incorporate these into the Islet method.

Second, this article focuses on the horizontal dimensions or, more generally, 2D Lagrangian levels in a three-dimensional discretization. However, it is necessary to increase simultaneously both the horizontal and vertical resolutions of a plume to capture and maintain its structure (Eastham and Jacob, 2017). Future work using methods similar to those in Sect. 3 should address the vertical dimension.

Third, it is possible to recover local mass conservation in an ISL method by modifying the ~~space-time~~ linear advection operator's coefficients so that the sum of ~~the~~ mass over all ~~the~~ target points associated with a source element is consistent with the source element's total mass (Kaas, 2008). This ~~optional~~ coefficient modification step ~~is~~ might be compatible with the Islet method and should be explored.

Finally, we predict that applications related to, in particular, aerosols will benefit from the Islet method. In future work, we intend to integrate the Islet method into the E3SM Atmosphere Model to investigate its impact on science applications.

Code and data availability. Code and scripts for the algorithms and figures presented in this paper are available at <https://doi.org/10.5281/zenodo.5595499> and, alternatively, <https://github.com/E3SM-Project/COMPOSE/releases/tag/v1.1.2>. In this repository, read `methods/-islet/readme.txt` for further instructions; in particular, this file points to details instructions to recreate the data in this article in the file `methods/islet/figures/figs.tex`. These data are also available at <https://doi.org/10.5281/zenodo.5595518>.

Appendix A: ~~Maximum eigenvalue magnitude~~ Islet bases

~~Let an eigenvector of \mathbf{A} defined in Eq. be \mathbf{V} , and write $\mathbf{V} \equiv (v_0^T \cdots v_{N-1}^T)^T$, where d -vector \mathbf{v}_k .~~ The GLL basis functions yield an unstable ISL method for $n_p \geq 4$. In the companion article Bradley (2022) (see also Bradley et al. (2021, Sect. 2 and 3)), we develop a set of bases on GLL nodes that yield ISL methods that meet a necessary condition for stability on the test problem of uniform periodic flow. We refer to these as *Islet bases*. We use Islet bases in this article. This appendix describes these bases for completeness. This article can be understood equally well by assuming the standard GLL bases are used; only the numerical results depend on the details of the basis functions.

A 1D reference element has domain $[-1, 1]$. Higher-dimensional basis functions are tensor products of 1D basis functions. The bases are nodal, meaning a basis function has the value 1 at one node and 0 at all others. The nodes are GLL. Let $x_G^{n_p}(i)$, $i \in \{0, \dots, n_p - 1\}$, be the reference coordinates of the GLL nodes. Let *region* $r \in \{0, \dots, n_p - 2\}$ be the segment $[x_G^{n_p}(r), x_G^{n_p}(r + 1)]$. A 1D basis function is a continuous piecewise polynomial over $[-1, 1]$; in each region, it is a polynomial. Each region r has an associated ordered list of *support nodes*, denoted $\mathcal{I}_r^{n_p}$. Each list $\mathcal{I}_r^{n_p}$ is a subset of $\{0, \dots, n_p - 1\}$. Let $|\mathcal{I}_r^{n_p}|$ be the number of elements in $\mathcal{I}_r^{n_p}$, let $n_p^{\text{submin}} \equiv \min_r |\mathcal{I}_r^{n_p}|$, and let $\mathcal{I}_r^{n_p}(j)$ be the j th element.

In region r and thus for $x \in [x_G^{n_p}(r), x_G^{n_p}(r+1)]$, basis function k is d -length block row k of \mathbf{V} , $k \in \{0, \dots, N-1\}$. We suppose $\mathbf{v}_k = \mu^k \mathbf{q}$, $\mu = e^{i2\pi j/N}$, $j \in \{0, \dots, N-1\}$, i the imaginary unit, 0 if $k \notin \mathcal{I}^{n_p}$ and otherwise has the value given by the Lagrange polynomial

$$\phi_k^{n_p}(x) \equiv \prod_{\substack{j \in \{0, \dots, |\mathcal{I}^{n_p}|-1\}, \\ \mathcal{I}^{n_p}(j) \neq k}} \frac{x - x_G^{n_p}(\mathcal{I}^{n_p}(j))}{x_G^{n_p}(k) - x_G^{n_p}(\mathcal{I}^{n_p}(j))}.$$

1400 We construct a basis for each $n_p \in \{4, \dots, 13\}$. Each basis is described by a set of support node lists, one $\mathcal{I}_r^{n_p}$ per region. For $n_p = 4$, there are additional details that we describe in Appendix A1. There are two types of bases, each corresponding to a method of describing the support node lists.

The first type of basis is the *offset nodal basis*. Each list $\mathcal{I}_r^{n_p}$ is given by an *offset*, which is the first index in the list, and its size, $|\mathcal{I}_r^{n_p}|$. For example, $n_p = 7$, offset 2, and \mathbf{q} an unknown d -vectorsize 4 correspond to support nodes $\mathcal{I}_r^7 = \{2, 3, 4, 5\}$. In
1405 addition, the basis is symmetric, meaning basis function $\phi_i^{n_p}(x) = \phi_{n_p-1-i}^{n_p}(-x)$. Thus, first, support nodes are specified for regions 0 through $\lfloor n_p/2 \rfloor - 1$, and the support nodes for the remaining regions are determined by symmetry. Second, if n_p is even, then the middle region, $r = n_p/2 - 1$, has support nodes $\mathcal{I}_r^{n_p}$ that are symmetric around reference coordinate 0.

The second type of basis is the *nodal basis*. An offset nodal basis is a nodal basis whose support nodes all permit the more compact description of offset and size. A general nodal basis has at least one support nodal list that cannot be described by just
1410 an offset and a size. Instead, the list is stated explicitly. Again, all nodal bases are symmetric.

The order of accuracy of the ISL method using an Islet GLL basis is $n_p^{\text{submin}} - 1$ if there is no property preservation step and the initial condition is $C^{n_p^{\text{submin}}-2}$.

Let $r \equiv (r' + \rho d) \bmod (Nd)$ for $\rho \in \{0, \dots, N-1\}$. We need to establish some indices and permutations. Let $\bar{r} \equiv \lfloor r/d \rfloor$, $\hat{r} \equiv r \bmod d$ and

$$1415 \mathbf{S}_r^\mu \equiv \begin{pmatrix} \mathbf{0} & \mathbf{I}^{(d-\hat{r}) \times (d-\hat{r})} \\ \mu \mathbf{I}^{\hat{r} \times \hat{r}} & \mathbf{0} \end{pmatrix}.$$

If $n_p^{\text{submin}} = n_p$, then the basis is the natural GLL basis. Thus, any statement about an Islet GLL basis also holds for the natural GLL basis.

Now consider rows $(r + (0:d-1)) \bmod (Nd)$ of the eigenvalue problem $\mathbf{A}\mathbf{V} = \lambda\mathbf{V}$, and assume $\mathbf{v}_k = \mu^k \mathbf{q}$ as described above. The left hand side can be written

$$1420 \mathbf{A} \left((r + (0:d-1)) \bmod (Nd), : \right) \mathbf{V} \\ = \mathbf{B}\mathbf{V}((\rho d + (0:d)) \bmod (Nd)) \\ = \mathbf{B}\mu^\rho \begin{pmatrix} \mathbf{I}^{d \times d} \\ \mu \mathbf{e}_0^T \end{pmatrix} \mathbf{q}.$$

~~The right hand side can be written~~

$$\underline{\lambda} \quad \underline{\mathbf{V}((r + (0:d-1)) \bmod (Nd))}$$

1425

$$\begin{aligned} &= \lambda \mu^{\bar{r}} \begin{pmatrix} \mathbf{q}(\hat{r}:d-1) \\ \mu \mathbf{q}(0:\hat{r}-1) \end{pmatrix} \\ &= \lambda \mu^{\bar{r}} \mathbf{S}_r^\mu \mathbf{q}. \end{aligned}$$

~~Together, these give the $d \times d$ eigenvalue problem~~

$$\underline{\mu^{-\lfloor r'/d \rfloor} (\mathbf{S}_r^\mu)^H \mathbf{B} \begin{pmatrix} \mathbf{I}^{d \times d} \\ \mu \mathbf{e}_0^T \end{pmatrix} \mathbf{q} = \lambda \mathbf{q},}$$

1430

~~where H is A search procedure searches for nodal bases that provide ISL methods that are stable on the test problem of uniform flow on a uniform grid. An accuracy heuristic is used to find the most accurate basis satisfying this stability condition. Table A1 lists nodal bases, most of them offset nodal bases, for $n_p \in \{5, \dots, 13\}$. In addition, a special basis is constructed for $n_p = 4$.~~

A1 $n_p = 4$ Islet basis

1435

~~For $n_p = 4$, we boost the accuracy of the basis by combining an offset nodal basis and the GLL basis. The middle region has support nodes $\mathcal{I}_1 = \{0, 1, 2, 3\}$, i.e., all four nodes. Region 0 convexly combines basis function k given by $\mathcal{I}_0^4 = \{0, 1, 2\}$ with GLL basis function k over the region $x \in [x_G^4(0), x_G^4(1)]$, with convex combination function $\alpha(x)$. $\alpha(x)$ is a quadratic polynomial having values 1, 0.306, and 0 at, respectively the left side, middle, and right side of region 0. The combination is defined so that at the left side, where $\alpha(-1) = 1$, only the GLL basis function k is used; at the right side, where $\alpha(x_G^4(1)) = 0$, only the offset nodal basis function k is used. Region 2 is constructed by symmetry. $\alpha(x)$ is determined to maximize the accuracy of the matrix conjugate transpose and we have used $\mu^{\rho-\bar{r}} = \mu^{-\lfloor r'/d \rfloor}$. The eigenvalues of \mathbf{A} are the union of those obtained by solving resulting ISL method subject to the constraint of stability on the test problem.~~

1440

A2 Conservation when interpolating

1445

~~The element-local conservation expressed in Eq. for each $\mu = e^{i2\pi j/N}$, $j = 0, \dots, N-1$ (15) holds for the following reasons. First, for any Islet basis for which $(n_p^t)_{\text{submin}} > n_p^v$, the continuum field f is the same on each grid, since the tracer grid can exactly represent polynomials of degree $n_p^v - 1$. Second, for n_p^v even, such as the standard $n_p^v = 4$, if $(n_p^t)_{\text{submin}} = n_p^v - 1$, the continuum field f is in general different on each grid, but the integral of each over an element has the same value. This is because for polynomial degree d odd and $f(x) \equiv \sum_{i=0}^d a_i x^i$, $\int_{-1}^1 f(x) dx = \int_{-1}^1 (f(-x) + f(x))/2 dx$, and $g(x) \equiv (f(-x) + f(x))/2$ has degree at least one less than $f(x)$. $g(x)$ is exactly represented on the tracer grid, and thus the integral of f , which is the same as the integral of g , is the same on the tracer grid as on the dynamics grid. These two reasons assure that $\mathcal{I}^{v \rightarrow t}$ is conservative for the Islet bases in Table A1 when $n_p^v = 4$.~~

n_p	OOA	$ \mathcal{I}_r $	Supports
4	2	see text	see text
5	2	{3, 4}	offsets {0, 0}
6	4	{5, 5, 6}	nodal subsets {0, 1, 2, 3, 4}, {0, 1, 2, 3, 5}, {0, 1, 2, 3, 4, 5}}
7	4	{5, 5, 6}	offsets {0, 0, 0}
8	5	{6, 6, 7, 6}	offsets {0, 0, 0, 1}
9	6	{7, 8, 8, 7}	nodal subsets {0, 1, 2, 3, 4, 5, 8}, {0, 1, 2, 3, 4, 5, 7, 8}, {0, 1, 2, 3, 4, 5, 6, 8}, {1, 2, 3, 4, 5, 6, 7}}
10	6	{7, 7, 7, 8, 8}	offsets {0, 0, 0, 0, 1}
11	7	{8, 9, 8, 9, 8}	offsets {0, 0, 0, 0, 1}
12	8	{9, 9, 10, 10, 9, 10}	offsets {0, 0, 0, 0, 1, 1}
13	9	{10, 10, 10, 10, 11, 10}	offsets {0, 0, 0, 0, 0, 1}

Table A1. Islet GLL nodal subset bases. Each row provides a formula for the row's n_p value. Columns are n_p , order of accuracy (OOA), the support sizes $|\mathcal{I}_r|$ for each region ordered left to middle, and the supports. For offset nodal subset bases, supports are given by offsets. For general nodal subset bases, supports are given by nodal subsets, again ordered from left region to middle. The case $n_p = 4$ is described in Appendix A1. In all cases, the support points are GLL nodes.

1450 If every value of N is of interest, then $\mu = e^{i\theta}$, $\theta \in [0, 2\pi)$. Thus, to compute the maximum eigenvalue of \mathbf{A} independent of N , we solve the problem

$$\lambda_{\max}(\mathbf{B}) \equiv \max_{\substack{\theta \in [0, 2\pi) \\ i \in \{0, \dots, d-1\}}} |\lambda_i|$$

$$\text{subject to } (\mathbf{S}_r^{e^{i\theta}})^H \mathbf{B} \begin{pmatrix} \mathbf{I}^{d \times d} \\ e^{i\theta} \mathbf{e}_0^T \end{pmatrix} \mathbf{Q} = \Lambda \mathbf{Q},$$

1455 where Λ is the diagonal matrix of eigenvalues λ_i , $i \in \{0, \dots, d-1\}$. We omit $\mu^{-\lfloor r'/d \rfloor}$ from the eigenvalue problem because it does not affect an eigenvalue's magnitude. In practice we must limit the search to a discrete, but possibly large, subset of $[0, 2\pi)$. This problem is useful because it requires solving many separate $d \times d$ eigenvalue problems instead of one or more (since we

are interested in arbitrary N) $(Nd) \times (Nd)$ problems. Thus, first, one can program a work- and parallel-efficient method to check a particular discretization for empirical satisfaction of the necessary condition; second, we avoid the superlinear growth in the cost of the eigenvalue computation with N .

1460 A3 Unstable and stable integration

Figure A1, which has the format of the figures described in Sect. 4, compares accuracy and stability between the natural and Islet bases. The divergent flow, Gaussian hills IC, property preservation, TTPR, and long time steps are used. To test the stability of the ISL linear advection operator using the Islet bases, we run the validation problem for both 1 cycle and 100 cycles. A curve's line pattern corresponds to basis type and number of cycles: solid, Islet basis for 1 cycle; dashed, Islet basis for 100 cycles (or $12 \times 100 = 1200$ days); dash-dotted, natural basis for 1 cycle. For each n_p^l , the l_2 norm of the solution using the natural basis diverges with increasing resolution within the first cycle, demonstrating that the basis leads to an unstable ISL linear advection operator. In the case of $n_p^l = 4$, we see the start of the curve's divergence, but further element-grid refinement is needed to see the curve fully diverge. In contrast, the curves for the Islet method with the Islet bases converge at order of accuracy 2.

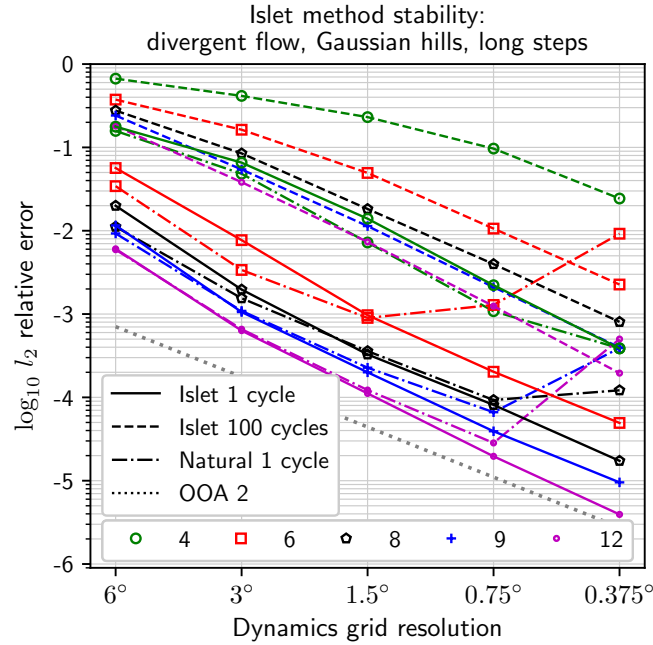


Figure A1. Stability of the Islet method with the Islet GLL bases, compared with the instability of the method with the natural GLL bases. The x -axis is the average dynamics-grid grid point spacing at the equator in degrees for the quasiuniform cubed-sphere grid. The y -axis is $\log_{10} l_2$ relative error. A curve's line pattern corresponds to the basis type and number of cycles, as listed in the top legend. A curve's marker corresponds to n_p^l , as listed in the bottom legend.

1470 **Appendix B: Discretization of the continuity equation**

Our primary interest is the advection equation and not the continuity equation. However, to run tests with divergent flow and property preservation enabled, we need to solve the continuity equation for the air density to provide the ρ argument to CAAS. This appendix describes a discretization of the continuity equation that differs in only one factor from the discretization of the advection equation.

1475 We start by following, e.g., Giraldo (1997, Sect. 3.3) and Bosler et al. (2019, Sect. 3). The Reynolds transport theorem for a Lagrangian fluid parcel $\Omega(t)$ containing a fluid having density $\rho(\mathbf{x}, t)$ is

$$\frac{d}{dt} \int_{\Omega(t)} \rho \, d\mathbf{x} = \int_{\Omega(t)} \rho_t + \nabla \cdot (\rho \mathbf{u}) \, d\mathbf{x}. \quad (\text{B1})$$

To balance clarity and concision, we include or omit the position and time arguments depending on the context. Suppose ρ satisfies the continuity equation,

1480 $\rho_t + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (\text{B2})$

In addition, consider a function $\phi(\mathbf{x}, t)$ that satisfies the advection equation:

$$\frac{D\phi}{Dt} = 0. \quad (\text{B3})$$

Then by Eqs. (B1), (B2), and (B3),

$$\begin{aligned} \frac{d}{dt} \int_{\Omega(t)} \rho \phi \, d\mathbf{x} &= \int_{\Omega(t)} (\rho \phi)_t + \nabla \cdot (\rho \phi \mathbf{u}) \, d\mathbf{x} \\ &= \int_{\Omega(t)} \phi [\rho_t + \nabla \cdot (\rho \mathbf{u})] + \rho \frac{D\phi}{Dt} \, d\mathbf{x} \\ &= 0, \end{aligned}$$

1485

which in turn implies

$$\int_{\Omega(t_1)} \rho \phi \, d\mathbf{x} = \int_{\Omega(t_0)} \rho \phi \, d\mathbf{x} \quad (\text{B4})$$

for all times t_0, t_1 .

1490 **Appendix C: Interpolants**

In this appendix, we construct expressions for an infinite set of potential bases. In this article, we restricted our search to two finite subsets of these. Now we discretize Eq. (B4) in space. First, we write an integral over a scalar f in terms of the element reference coordinate system. Let $\Omega(t)$ be a quadrilateral element, with $\Omega(t_1)$ the arrival Eulerian element. Then

$$\int_{\Omega(t)} f(\mathbf{x}, t) d\mathbf{x} = \int_{-1}^1 \int_{-1}^1 f(\mathbf{x}(\mathbf{r}, t), t) J(\mathbf{r}, t) d\mathbf{r}. \quad (\text{B1})$$

Here, $\mathbf{x}(\mathbf{r}, t) \equiv \mathbf{X}(t; \mathbf{x}(\mathbf{r}, t_1), t_1)$ is the time-dependent map from the reference element to the manifold and satisfies Eq. (6). $J(\mathbf{r}, t)$ is the absolute value of the Jacobian determinant of this map, $|\det(\partial \mathbf{x}(\mathbf{r}, t) / \partial \mathbf{r})|$.

1. Given a departure point \mathbf{x} , Second, let the element have a tensor-product grid of GLL nodes, with n_p the number of 1D nodes. We apply tensor-product GLL quadrature with this value of n_p to the right hand side of Eq. (B1). Then

$$\int_{-1}^1 \int_{-1}^1 f(\mathbf{x}(\mathbf{r}, t), t) J(\mathbf{r}, t) d\mathbf{r} \approx \sum_{i=0}^{n_p^2-1} f(\mathbf{x}(\mathbf{r}_i, t), t) J(\mathbf{r}_i, t) w_i,$$

where i iterates over nodes in the element, \mathbf{r}_i is the i th GLL node, and w_i is the interpolant is a linear operator in the values \mathbf{y} at basis points \mathbf{x} . Anticipating additional constraints, we begin by writing this interpolant in the form

$$f(\mathbf{y}; \mathbf{x}) \equiv \mathcal{L}(\mathbf{x}; \mathbf{M}^{n \times n}(\mathbf{x}) \mathbf{y}, \mathbf{x}^n).$$

2. Within each region, $M(\mathbf{x})$ is constant:-

$$\mathbf{M}(\mathbf{x}) = \sum_{r=0}^{d-1} \text{Hat}(\mathbf{x}; \mathbf{x}(r), \mathbf{x}(r+1)) \mathbf{A}_r^{n \times n},$$

where

$$\text{Hat}(\mathbf{x}; a, b) \equiv \begin{cases} 1 & \text{if } a \leq \mathbf{x} \leq b \\ 0 & \text{else.} \end{cases}$$

We call each \mathbf{A}_r a region operator with GLL weight. Let $\mathbf{x}_i(t) \equiv \mathbf{x}(\mathbf{r}_i, t)$.

3. Third, we combine the quadrature approximation with a specific form of f must recover a degree- p polynomial with specified $p = s - 1$ and $2 \leq s \leq n$. This order constraint implies

$$\mathbf{A}_r \begin{pmatrix} \mathbf{I}^{s \times s} \\ \mathbf{B}^{(n-s) \times s} \end{pmatrix} = \begin{pmatrix} \mathbf{I}^{s \times s} \\ \mathbf{B}^{(n-s) \times s} \end{pmatrix},$$

1515 $f = \rho\phi$, where $\mathbf{B}^{(n-s) \times s} = \mathcal{L}(\mathbf{x}(s:n-1); \mathbf{I}^{s \times s}, \mathbf{x}(0:s-1))$. ϕ satisfies Eq. (B3). Let $\phi(\mathbf{x}, t_1)$ be an element, GLL nodal, tensor-product n_p -basis function on an Eulerian arrival element. Because ϕ is a nodal basis function with the same nodes as the element, $\phi = 1$ at one node, say node k , and is 0 at all other nodes $i \neq k$. ϕ need not be a natural GLL basis function; for example, it can be an Islet GLL basis function. Because $D\phi/Dt = 0$, $\phi(\mathbf{x}_i(t), t) = \phi(\mathbf{x}_i(t_1), t_1)$ for any time t . Thus, the quadrature sum can be simplified:

$$\sum_{i=0}^{n_p^2-1} \rho(\mathbf{x}_i(t), t) \phi(\mathbf{x}_i(t), t) J(\mathbf{r}_i, t) w_i \\ = \rho(\mathbf{x}_k(t), t) J(\mathbf{r}_k, t) w_k.$$

4. Fourth, we apply quadrature to each side of Eq. (B4) to obtain the discretization

$$1520 \quad \rho(\mathbf{x}_k(t_1), t_1) J(\mathbf{r}_k, t_1) w_k = \rho(\mathbf{x}_k(t_0), t_0) J(\mathbf{r}_k, t_0) w_k,$$

or

$$\rho(\mathbf{x}_k(t_1), t_1) = \frac{J(\mathbf{r}_k, t_0)}{J(\mathbf{r}_k, t_1)} \rho(\mathbf{x}_k(t_0), t_0). \quad (\text{B2})$$

Eq. f must interpolate nodes $r, r+1$. This region-interpolation constraint implies

$$\mathbf{A}_r([r, r+1], :) = \begin{pmatrix} \mathbf{0}^{2 \times r} & \mathbf{I}^{2 \times 2} & \mathbf{0}^{2 \times (n-r-2)} \end{pmatrix}.$$

1525 (B2) differs from Eq. (5) in the appearance of the *density factor*, the quotient of the two Jacobian determinants. In this equation, both ρ values and the denominator of the density factor are grid point values. Note that this simple form depends on three different node sets being the same: those for the element, the GLL quadrature, and the basis function ϕ .

5. Consider the n basis functions $\phi_i(x) \equiv f(\mathbf{e}_i; x)$. These must have the symmetry $\phi_i(x) = \phi_{n-1-i}(-x)$. This symmetry constraint implies, first, \mathbf{x} has points symmetric around 0; second,

$$1530 \quad \mathbf{A}_r = \mathbf{A}_{n-1-r}(n-1:-1:0, n-1:-1:0).$$

Fifth, to complete the discretization, the discretization steps in Sect. 2.0.1 are applied to Eq. (B2) as they were to Eq. (5), but now with the additional factor present.

For a non-middle-region operator, \mathbf{A}_r starts with n^2 degrees of freedom, the order constraint removes ns , and Since Eq. (B2) is not the focus of this article, we omit detailed analysis of the discretization. However, we note some basic facts. Importantly, the Jacobian determinant of the Lagrangian element appears in the numerator rather than the denominator of the density factor; thus, the denominator is always well behaved if the Eulerian grid has reasonable quality. Second, for certain simple flows such as uniform translational flow on the plane, J is constant in time and Eq. (B2) reduces to the discretization of the region interpolation constraint removes $2(n-s)$ more since $2s$ degrees of freedom are shared with the order constraint. This leaves

$n^2 - ns - 2(n - s) = (n - 2)(n - s)$ degrees of freedom. The symmetry constraint implies that a middle-region operator \mathbf{A}_T has half of these: $(n - 2)(n - s)/2$.
 We can devise a basis \mathcal{N}_{n_p} by selecting values for node points \mathbf{x}^{n_p} and the region operators \mathbf{A}_T , subject to constraints 1–5: advection equation, Eq. (5). For general nondivergent flow, J is time dependent because of the spatial discretization error. Third, only the density factor can make the order of accuracy of the discretization Eq. (B2) differ from that of Eq. (5). Details concerning the manifold on which flow occurs and the map from the reference element to the manifold affect the order of accuracy. In the validation problems in this article, the density ρ is approximated using the Islet $n_p = 4$ basis, the manifold is the sphere, and the map from the reference element to the sphere is the isoparametric map of the finite element method for the n_p -basis. The discretization of the continuity equation for ρ then has order of accuracy 2.

Author contributions. AMB developed the algorithms, wrote the software, ran the numerical studies, and wrote the manuscript, all with contributions from PAB and OG. PAB administers the project that funded most of this work.

Competing interests. The authors declare that they have no conflict of interest.

Disclaimer. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. SAND2021-10514 O.

Acknowledgements. We thank Mark A. Taylor for valuable discussions regarding material in this article.

This work was supported by the US Department of Energy (DOE) Office of Science’s Advanced Scientific Computing Research (ASCR) and Biological and Environmental Research (BER) Programs under the Scientific Discovery through Advanced Computing (SciDAC 4) ASCR/BER Partnership Program, and by the Energy Exascale Earth System Model (E3SM) project, funded by BER. This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

References

- Ainsworth, M.: Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods, *J. Comput. Phys.*, 198, 106–130, 2004.
- 1565 Baptista, A. M.: Solution of advection-dominated transport by Eulerian-Lagrangian methods using the backwards method of characteristics, Ph.D. thesis, Massachusetts Institute of Technology, 1987.
- Bertagna, L., Guba, O., Taylor, M., Foucar, J., Larkin, J., Bradley, A., Rajamanickam, S., and Salinger, A.: A performance-portable nonhydrostatic atmospheric dycore for the Energy Exascale Earth System Model running at cloud-resolving resolutions, in: *SC20: Int. Conf. High Perform.*, pp. 1304–1317, IEEE Computer Society, 2020.
- 1570 Bochev, P., Moe, S., Peterson, K., and Ridzal, D.: A Conservative, Optimization-Based Semi-Lagrangian Spectral Element Method for Passive Tracer Transport, in: *Coupled Problems*, edited by Schrefler, B., Onate, E., and M., P., pp. 23–34, CIMNE, 2015.
- Bosler, P. A., Bradley, A. M., and Taylor, M. A.: Conservative multimoment transport along characteristics for discontinuous Galerkin methods, *SIAM J. on Sci. Comput.*, 41, B870–B902, 2019.
- Bradley, A. M.: Stabilized bases for spectral-element interpolation semi-Lagrangian transport, in preparation, 2022.
- 1575 Bradley, A. M., Bosler, P. A., Guba, O., Taylor, M. A., and Barnett, G. A.: Communication-efficient property preservation in tracer transport, *SIAM J. on Sci. Comput.*, 41, C161–C193, 2019.
- Bradley, A. M., Bosler, P. A., and Guba, O.: Islet: Interpolation semi-Lagrangian element-based transport, *Geosci. Model Dev. Discuss.*, p. [preprint], <https://doi.org/10.5194/gmd-2021-296>, 2021.
- Caldwell, P. M., Terai, C. R., Hillman, B. R., Keen, N. D., Bogenschutz, P. A., Lin, W., Beydoun, H., Taylor, M. A., Bertagna, L., Bradley, A. M., et al.: Convection-Permitting Simulations with the E3SM Global Atmosphere Model, *J. Adv. Model Earth Sy.*, p. e2021MS002544, 2021.
- 1580 Chen, Y., Simon, K., and Behrens, J.: Extending legacy climate models by adaptive mesh refinement for single-component tracer transport: a case study with ECHAM6-HAMMOZ (ECHAM6. 3-HAM2. 3-MOZ1. 0), *Geosci. Model Dev.*, 14, 2289–2316, 2021.
- Cohen, G.: *Higher-order numerical methods for transient wave equations*, Springer Science & Business Media, 2001.
- 1585 Dennis, J., Fournier, A., Spatz, W. F., St-Cyr, A., Taylor, M. A., Thomas, S. J., and Tufo, H.: High-resolution mesh convergence properties and parallel efficiency of a spectral element atmospheric dynamical core, *Int. J. High Perform C.*, 19, 225–235, 2005.
- Dennis, J., Edwards, J., Evans, K. J., Guba, O., Lauritzen, P., Mirin, A. A., St-Cyr, A., Taylor, M. A., and Worley, P. H.: CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model, *Int. J. High Perform C.*, 26, 74–89, 2012.
- Dormand, J. R. and Prince, P. J.: A family of embedded Runge-Kutta formulae, *J. of Comput. and Appl. Math.*, 6, 19–26, 1980.
- 1590 E3SM Project: Energy Exascale Earth System Model (E3SM), [Computer Software] <https://dx.doi.org/10.11578/E3SM/dc.20180418.36>, 2018.
- Eastham, S. D. and Jacob, D. J.: Limits on the ability of global Eulerian models to resolve intercontinental transport of chemical plumes, *Atmos. Chem. Phys.*, 17, 2543–2553, 2017.
- Erath, C. and Nair, R. D.: A conservative multi-tracer transport scheme for spectral-element spherical grids, *J. Comput. Phys.*, 256, 118–134, 2014.
- 1595 Field, C., Cheung, W., Dilling, L., Frumhoff, P., Greely, H., Hordequin, M., Hurrell, J., Light, A., Lin, A., MacMartin, D., et al.: *Reflecting Sunlight: Recommendations for Solar Geoengineering Research and Research Governance*, The National Academies Press, Washington, DC, <https://doi.org/10.17226/25762>, 2021.

- Fischer, P. F. and Patera, A. T.: Parallel spectral element methods for the incompressible Navier-Stokes equations, in: *Solution of Superlarge Problems in Computational Mechanics*, pp. 49–65, Springer, 1989.
- Giraldo, F. X.: Lagrange–Galerkin methods on spherical geodesic grids, *Journal of Computational Physics*, 136, 197–213, 1997.
- Giraldo, F. X.: The Lagrange–Galerkin spectral element method on unstructured quadrilateral grids, *J. Comput. Phys.*, 147, 114–146, 1998.
- Giraldo, F. X., Perot, J. B., and Fischer, P. F.: A spectral element semi-Lagrangian (SESL) method for the spherical shallow water equations, *J. Comput. Phys.*, 190, 623–650, 2003.
- Golaz, J.-C., Caldwell, P. M., Van Roekel, L. P., Petersen, M. R., Tang, Q., Wolfe, J. D., Abeshu, G., Anantharaj, V., Asay-Davis, X. S., Bader, D. C., et al.: The DOE E3SM coupled model version 1: Overview and evaluation at standard resolution, *J. Adv. Model Earth Sy.*, 11, 2089–2129, 2019.
- Golaz, J.-C. et al.: The DOE E3SM Model Version 2: Overview of the physical model, in preparation for *J. Adv. Model Earth Sy.*, 2022.
- Guba, O., Taylor, M. A., Ullrich, P. A., Overfelt, J. R., and Levy, M. N.: The spectral element method (SEM) on variable-resolution grids: Evaluating grid sensitivity and resolution-aware numerical viscosity, *Geosci. Model Dev.*, 7, 2803–2816, 2014.
- Hannah, W. M., Bradley, A. M., Guba, O., Q., T., Golaz, J.-C., and Wolfe, J.: Separating Physics and Dynamics grids for Improved Computational Efficiency in Spectral Element Earth System Models, *J. Adv. Model Earth Sy.*, 13, e2020MS002419, 2021.
- Herrington, A. R., Lauritzen, P. H., Reed, K. A., Goldhaber, S., and Eaton, B. E.: Exploring a lower resolution physics grid in CAM-SECSLAM, *J. Adv. Model Earth Sy.*, p. 2019MS001684, <https://doi.org/10.1029/2019MS001684>, 2019.
- Idelsohn, S., Storti, M., and Nigro, N.: Stability analysis of mixed finite element formulations with special mention of equal-order interpolations, *Int. J. for Numer. Meth. in Fl.*, 20, 1003–1022, 1995.
- Kaas, E.: A simple and efficient locally mass conserving semi-Lagrangian transport scheme, *Tellus A*, 60, 305–320, 2008.
- Lauritzen, P. H.: Atmospheric Transport Schemes: Desirable Properties and a Semi-Lagrangian View on Finite-Volume Discretizations, in: *Numerical Techniques for Global Atmospheric Models*, edited by Lauritzen, P. H., Jablonowski, C., Taylor, M. A., and Nair, R. D., vol. 80 of *Lecture Notes in Computational Science and Engineering*, pp. 185–250, Springer: Berlin, 2011.
- Lauritzen, P. H. and Thuburn, J.: Evaluating advection/transport schemes using interrelated tracers, scatter plots and numerical mixing diagnostics, *Q. J. Roy. Meteor. Soc.*, 138, 906–918, 2012.
- Lauritzen, P. H., Nair, R. D., and Ullrich, P. A.: A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid, *J. Comput. Phys.*, 229, 1401–1424, 2010.
- Lauritzen, P. H., Skamarock, W. C., Prather, M. J., and Taylor, M. A.: A standard test case suite for two-dimensional linear transport on the sphere, *Geosci. Model Dev.*, 5, 2012.
- Lauritzen, P. H., Ullrich, P. A., Jablonowski, C., Bosler, P. A., Calhoun, D., Conley, A. J., Enomoto, T., Dong, L., Dubey, S., Guba, O., et al.: 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, 105–145, 2014.
- Lauritzen, P. H., Conley, A. J., Lamarque, J.-F., Vitt, F., and Taylor, M. A.: The terminator “toy” chemistry test: a simple tool to assess errors in transport schemes, *Geosci. Model Dev.*, 8, 1299–1313, 2015.
- Lauritzen, P. H., Taylor, M. A., Overfelt, J., Ullrich, P. A., Nair, R. D., Goldhaber, S., and Kelly, R.: CAM-SE–CSLAM: Consistent Coupling of a Conservative Semi-Lagrangian Finite-Volume Method with Spectral Element Dynamics, *Mon. Weather Rev.*, 145, 833–855, 2017.
- Lee, D., Lowrie, R., Petersen, M., Ringler, T., and Hecht, M.: A high order characteristic discontinuous Galerkin scheme for advection on unstructured meshes, *J. Comput. Phys.*, 324, 289–302, 2016.

- Lee, H.-H., Bogenschutz, P., and Yamaguchi, T.: The Implementation of Framework for Improvement by Vertical Enhancement (FIVE) into Energy Exascale Earth System Model (E3SM), *J. Adv. Model Earth Sy.*, p. e2020MS002240, 2020.
- Lozier, D. W.: NIST digital library of mathematical functions, *Ann. Math. Artif. Intel.*, 38, 105–119, 2003.
- Michalakes, J., Govett, M., Benson, R., Black, T., Juang, H., Reinecke, A., Skamarock, B., Duda, M., Henderson, T., Madden, P., Mozdzyński, G., and Vasic, R.: AVEC Report: NGGPS Level-1 Benchmarks and Software Evaluation, Tech. rep., NOAA, <https://repository.library.noaa.gov/view/noaa/18654>, 2016.
- Natarajan, H. and Jacobs, G. B.: An explicit semi-Lagrangian, spectral method for solution of Lagrangian transport equations in Eulerian-Lagrangian formulations, *Comput. Fluids*, 207, 104 526, 2020.
- Oliveira, A. and Baptista, A. M.: A comparison of integration and interpolation Eulerian-Lagrangian methods, *Int. J. for Numer. Meth. in Fl.*, 21, 183–204, 1995.
- Semakin, A. N. and Rastigejev, Y.: Optimized wavelet-based adaptive mesh refinement algorithm for numerical modeling of three-dimensional global-scale atmospheric chemical transport, *Q. J. Roy. Meteor. Soc.*, 146, 1564–1574, 2020.
- Shampine, L. F. and Reichelt, M. W.: The Matlab ODE suite, *SIAM J. on Sci. Comput.*, 18, 1–22, 1997.
- Stensrud, D. J.: *Parameterization Schemes: Keys to Understanding Numerical Weather Prediction Models*, Cambridge University Press, 2009.
- Strikwerda, J. C.: *Finite difference schemes and partial differential equations*, SIAM, 2004.
- Tang, Q., Klein, S. A., Xie, S., Lin, W., Golaz, J.-C., Roesler, E. L., Taylor, M. A., Rasch, P. J., Bader, D. C., Berg, L. K., et al.: Regionally refined test bed in E3SM atmosphere model version 1 (EAMv1) and applications for high-resolution modeling, *Geosci. Model Dev.*, 12, 2679–2706, 2019.
- Taylor, M. A., Guba, O., Steyer, A., Ullrich, P. A., Hall, D. M., and Eldrid, C.: An energy consistent discretization of the nonhydrostatic equations in primitive variables, *J. Adv. Model Earth Sy.*, 12, 2020.
- Ullrich, P. A., Lauritzen, P. H., and Jablonowski, C.: Some considerations for high-order ‘incremental remap’-based transport schemes: edges, reconstructions, and area integration, *Int. J. for Numer. Meth. in Fl.*, 71, 1131–1151, 2013.
- Vichnevetsky, R. and Bowles, J. B.: *Fourier analysis of numerical approximations of hyperbolic equations*, SIAM, 1982.
- Whitaker, J.: HIWPP non-hydrostatic dynamical core tests: Results from idealized test cases, Tech. rep., NOAA, https://www.weather.gov/media/sti/nggps/HIWPP_idealized_tests-v8%20revised%2005212015.pdf, 2016.
- Yamaguchi, T., Feingold, G., and Larson, V. E.: Framework for improvement by vertical enhancement: A simple approach to improve representation of low and high-level clouds in large-scale models, *J. Adv. Model Earth Sy.*, 9, 627–646, 2017.