EULAG, a computational model for multiscale flows
Joseph M. Prusa a,*, Piotr K. Smolarkiewicz b, Andrzej A. Wyszogrodzki b

a Iowa State University, Ames, IA 50011, USA
b National Center for Atmospheric Research, Boulder, CO 80307, USA

Received 8 August 2007; received in revised form 28 November 2007; accepted 3 December 2007
Available online 23 December 2007

Abstract

EULAG (Eulerian/semi-Lagrangian fluid solver) is an established computational model for simulating thermo-fluid flows across a wide range of scales and physical scenarios. It is noteworthy for its nonoscillatory integration algorithms, robust elliptic solver, and generalized coordinate formulation enabling grid adaptivity technology. In this paper we highlight the key model ingredients, demonstrate its capabilities with a select subset of recent applications, and show its performance both in terms of accuracy and scalability on massively parallel processor architectures. A comprehensive list of references is provided to facilitate more detailed study.

1. Introduction

There are only a few principal categories of tools available for quantitative research in engineering and the physical sciences: laboratory experimentation, linear and weakly nonlinear theories, and numerical simulation. When used in concert, these tools can amplify each other’s contribution and accelerate the cognition; cf.[17,67]. Decades of progress with computer hardware and software have elevated the position of numerical calculations in the classical paradigm of theory and experimentation. Nevertheless, existing research and commercial software typically aim at narrowly focused though accurate applications, or more broadly formulated applications suitable for a wide range of practical problems but of reduced accuracy. EULAG [13] provides a multi-scale computational model that maintains uniformly second-order accuracy while being adaptable to a broad range of applications.

Historically, EULAG originated as a computational model to test advanced numerical methods for integrating atmospheric/oceanic fluid equations. As the model evolved with the progress of the methods, it turned out to be useful as a research tool in a much broader range of fluid dynamics problems. At present, EULAG clones are used in a number of university departments and research centers nationally and internationally. EULAG has a proven record of successful applications in the areas of turbulence (see Fig. 1 for illustration), urban flows (Fig. 2), gravity wave dynamics, flows past complex/moving boundaries (Fig. 3), micrometeorology, cloud microphysics and dynamics [1], global atmospheric [50], and basic fluid dynamics of incompressible fluids. Derivatives of EULAG also have been applied to simulations of visco-elastic waves in the human brain [6], oceanic flows [64], and stellar convection (Fig. 4). Present developments include extensions to gas dynamics and solar MHD (magneto-hydrodynamics). The model scales well on parallel computing architectures; recent simulations have demonstrated performance approaching and exceeding one teraflop. EULAG is also capable of running on a number of computer architectures, down to the scale of laptops. This makes the model attractive for adaptation in a wide range of academic/industrial environments.

The name EULAG alludes to the capability to solve the fluid equations in either an Eulerian (flux form) [47] or a Lagrangian (advective form) [46] mode. In either case the underlying transport operators are formulated for arbitrary (sufficiently continuous) curvilinear frameworks.

* Corresponding author. Address: Teraflux Corp., Boca Raton, FL 33486, USA. Tel.: +1 561 417 8571.
E-mail address: jprusa@bellsouth.net (J.M. Prusa).

0045-7930/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2007.12.001
The viscosity values are indicated along the curves. Sign preservation can be assured [55]. Adequately limited time step; for arbitrary flows, the weaker condition of the preceding time step, given solenoidal advecting flow and an dissipative term [41], while being second-order-accurate in space and time as well as sign preserving. The distinctness of MPDATA’s methodology from other high-resolution schemes is emphasized in [28,30]. Therein, the authors derived an MPDATA-based framework for rationalizing ILES (implicit large eddy simulation)—an ability of selected nonoscillatory methods to produce LES (large eddy simulations) without invoking explicit SGS (subgrid-scale) models; see [20] for a comprehensive exposition.

Recent series of studies [51,26,8,9,27] have quantified MPDATA’s implicit turbulence modeling capability (see Fig. 5 for example), suitable for high Reynolds number flows. These simulations have ranged from canonical decaying turbulence in a triply-periodic box, through development of convective planetary boundary layers and breaking of internal gravity waves in the Earth’s atmosphere, to turbulent solar convection [52,25,35,36,12]. Having an option for ILES is significant because it obviates the need to evaluate the viscous stress—a formidable task in generalized curvilinear frameworks [54]. Nevertheless, EULAG also retains the standard options for LES, cf. [33], and DNS (direct numerical simulation), cf. [67], whereby the generalized coordinate approach has been extended and coded to necessary higher-order differential expressions. Each parameter regime (i.e., ILES, LES, or DNS) is important in a multi-scale model.

The default analytic formulation of EULAG assumes the nonhydrostatic anelastic equations of motion, with options available for compressible/incompressible Boussinesq, incompressible Euler/Navier–Stokes [50], and fully compressible equations for high-speed flows [57]. Current work focuses on extensions to Durran’s pseudo-incompressible system [10,58] and compressible equations for low and moderate Mach number flows. The Durran equations conveniently separate the baroclinicity per se from the compressibility effects, thus enabling inquiries into the role of baroclinicity in fluid states close to bifurcation [42,43,58]. The subsonic compressible systems appear important for studies of environmental acoustics and near field solutions in, e.g., public aviation [68]. The role of compressibility versus baroclinicity is of particular interest in weather and climate prediction.

The governing PDEs of EULAG are formulated and solved in generalized time-dependent curvilinear coordinates [35,38,65,54,40]. This enables grid adaptivity (hereafter GA for brevity) via continuous mappings from a physical space $S_p$, where the problem is posed, into a transformed space $S_l$ where the problem is solved. Formally, the mapping is assumed sufficiently continuous that $S_l$ is homeomorphic to $S_p$. A tensor formalism is used in EULAG to develop the analytical formulation; and it is also utilized to enforce several geometric identities that are ubiquitous in tensor manipulations (see [40] for details). We have found that attention to such identities is essential to maintaining accuracy with generalized coordinate models. From a computer-science perspective, continuous mappings are perfectly suited for modern massively-parallel distributed-memory machines, as they add little communication overhead compared to uniformly discretized Cartesian (reference) equations. From a physical viewpoint GA, together with the nonhydrostatics underlying EULAG’s formulation, admits multi-scale simulations within a single model domain.

In the following section we present salient details of the analytical and numerical aspects of the model. Section 3 reports on advanced details entailing implicit advection integrals, coordinate mappings, and the tensor identities underlying the model. In Section 4, we address the model accuracy, and provide results showing the scalability of EULAG using several massively parallel processor machines. The paper concludes with some general comments about the model and implications for further development in Section 5.

---

1 Numerical solutions remain bounded by the surrounding local values of the preceding time step, given solenoidal advecting flow and an adequately limited time step; for arbitrary flows, the weaker condition of sign preservation can be assured [55].

2 We do not introduce singular surfaces and/or points so as to change the discrete topology of $S_l$ from that of $S_p$.
2. Model formulation

For clarity, we present the model equations here using only the dynamical variables, without complications such as moisture or chemistry, and leave the nature of physical forces other than buoyancy, pressure, and Coriolis explicitly unspecified. Furthermore, we use a concise symbolic operator-form description of the governing equations; for the complete tensorial expositions refer to \[38,39,65,54,40\] and references therein. The following thus demonstrates only a small portion of EULAG’s details and capabilities, but the methodology is representative of the entire model.

2.1. Analytics

The physical problem to be solved by EULAG is posed in a classical Lorentz space \( \mathbb{S}^p \) termed the physical space \( \mathbb{S}_p \). A defining characteristic of this space is that it has zero Riemannian curvature and the mapping into transformed space \( \mathbb{S}^t \) preserves this characteristic. This constraint does not preclude model applications in submanifolds of \( \mathbb{S}_p \) that have nonzero curvature, such as curved surfaces. To more easily reveal the physics and improve the formulation of boundary conditions that are characteristic of particular classes of flows in a variety of domains, the governing equations may be formulated utilizing any stationary, orthogonal system of coordinates to describe \( \mathbb{S}_p \).\(^3\) The general form of the mapping from connected subdomain \( \mathcal{D}_p \subseteq \mathbb{S}_p \) to connected subdomain \( \mathcal{D}_t \subseteq \mathbb{S}_t \) is

\[
\forall (\bar{t}, \bar{x}) \in \mathcal{D}_t, \quad (\bar{t}, \bar{x}) = (t, \mathcal{F}(t, x)),
\]

where \((t, x) \in \mathcal{D}_p\). Here \( \mathcal{D}_p \) is the computational domain in physical space, and \( \mathcal{D}_t \) is its image in transformed space. \( \mathcal{F} \) is a sufficiently continuous function that takes 4-tuples of \( \mathcal{D}_p \) into the spatial coordinates \( x \) of \( \mathcal{D}_t \) such that the pair of functions \((t, \mathcal{F})\) are an almost global diffeomorphism. By this we mean the mapping is diffeomorphic in the interiors of a finite number of disjoint subdomains \( \mathcal{D}_p^m \), \( m = 1, 2, \ldots, q \) that cover \( \mathcal{D}_p \). Discontinuity is allowed along a finite number of boundaries between the subdomains by EULAG’s nonoscillatory numerics, to be described in the following subsection. In particular, it is possible to have sudden changes in grid resolution, see \[38,40\] for examples. The map \( \mathcal{F} \) enables grid adaptivity simply by defining the computational grid along the transformed coordinates.

\(^3\) For example, rotating stars can be modeled using spherical coordinates whereas flows around cylinders can be modeled using polar cylindrical coordinates. Presently, the metric terms for rectangular, polar cylindrical and spherical coordinates are in the model code. Care must be given to the particulars of the boundary conditions so that they maintain the topological equivalence of \( \mathbb{S}_p \) and \( \mathbb{S}_t \).
Thus the image $\mathcal{D}_t$ has a regular form while $\mathcal{D}_p$ may be of highly irregular form (Fig. 6).\(^4\)

To solve the problem in the regular domain $\mathcal{D}_t$, it is necessary to formulate it in the transformed space $\mathcal{S}_t$. Conceptually, this is done most easily by formulating the physical problem in $\mathcal{S}_p$ and then transforming that using a generalized tensor approach (or via the chain rule on a case by case basis) into a formulation appropriate for the transformed space. Given the mapping (1), this transformation process results in the following compact forms for the generalized anelastic equations of Lipps and Hemler \[^24\]:\(^5\)

---

\(^4\) By regular (vs. irregular) form we mean that the domain boundaries follow coordinate lines in the simplest ways and do not vary in time.

\(^5\) Mathematically, the anelastic equations may be thought as a generalization of the classical incompressible Boussinesq approximation to vertically stratified basic states.
Because of the transformation (1), physical and (tensor) algebraic aspects are now intertwined with each other. Insofar as the physics is concerned: \( \mathbf{v} \) denotes the physical velocity vector (most easily defined in \( \mathbf{S}_p \) as \( d\mathbf{x}/dt \)); \( \theta, \rho, \) and \( \pi \) denote potential temperature,\(^6\) density, and a density-normalized pressure; \( \mathbf{g} \) is the acceleration of gravity, \( \mathbf{f} \) is the Coriolis acceleration (due to global rotation of the domain); \( \mathbf{D} \) and \( \mathbf{H} \) symbolize viscous dissipation of momentum and diffusion of heat, respectively. Geometry of \( \mathbf{S}_p \) enters through the term \( \mathbf{M} \) that symbolizes the inertial forces of coordinate-dependent metric accelerations [viz. Christoffel terms that account for the twisting, bending, and stretching of the (orthogonal) coordinates \( \mathbf{x} \) of \( \mathbf{S}_p \)]. Because (3) is formulated for physical velocities \( \mathbf{v} \), the mapping (1) adds no metric accelerations to \( \mathbf{M} \). Primes denote deviations from the Coriolis-balanced hydrostatic ambient (alias, environmental) state \( \mathbf{v}_e, \theta_e; \) and the subscript 0 refers to the basic state, i.e., a horizontally homogeneous, constant-stability hydrostatic reference state (cf. Section 2b in [5]).

Geometry further enters the governing equations as follows: in the mass continuity Eq. (2), \( \rho^* \equiv \rho_0 \bar{G} \) with \( \bar{G} \) denoting the Jacobian of the mapping from \( \mathbf{S}_p \) to \( \mathbf{S}_e \); whereas in the momentum Eq. (3), \( \bar{G} \) symbolizes a renormalized Jacobian matrix of the transformation coefficients \( \sim (\delta \mathbf{x}/\delta \mathbf{x}) \); \( \nabla^e \equiv \partial/\partial \mathbf{x}_e \), and the total derivative is given by \( d/d\tau = \partial/\partial t + \mathbf{v}^* \cdot \nabla \), where \( \mathbf{v}^* \equiv d\mathbf{x}/d\tau \equiv \dot{\mathbf{x}} \) is the contravariant velocity in \( \mathbf{S}_e \). Appearing in the continuity (2) and potential temperature (4) equations is a weighted solenoidal velocity,

\[
\mathbf{v}^e = \mathbf{v}^* - \frac{\partial \mathbf{x}}{\partial t}.
\]

Given the definition (5) as well as the generalized form of the anelastic continuity equation (see [38]), the form (2) readily follows from the assumptions that (i) \( \rho_e = \rho_e(\mathbf{x}) \), and (ii) the coordinate system in the physical space is time independent [37,40].

Use of the solenoidal velocity facilitates the solution procedure because it preserves the incompressible character of numerical equations. While numerous relationships can be derived that express any velocity (solenoidal, contravariant, or physical) in terms of any other, in either the transformed or physical coordinate system [38], a particularly useful transformation

\[
\mathbf{v}^e = \mathbf{G}^v \mathbf{v}
\]

relates the solenoidal and physical velocities directly. For further details of the metric and transformation tensors as well as the formulation of viscous and dissipative terms in the governing equations, the interested reader is referred to [39,54] and the references therein.

### 2.2. Numerics

Each prognostic equation that forms the anelastic system (3) and (4) can be written in two equivalent forms, either as a Lagrangian evolution equation

\[
\frac{d\psi}{d\tau} = R
\]

or Eulerian conservation law

\[
\frac{\partial \rho \psi}{\partial t} + \nabla \cdot (\rho \mathbf{v} \psi) = \rho R.
\]

Here \( \psi \) symbolizes components of \( \mathbf{v} \) or \( \theta^* \), and \( R \) denotes the associated rhs.

We approximate either (8) or (7) to second-order accuracy in space and time using an NFT (nonoscillatory forward-in-time) approach\(^7\) – see [52] for a review. A default NFT algorithm employed in EULAG can be formally written as

\[
\psi^{n+1}_i = LE_i(\psi^n) + 0.5\Delta t\psi^{n+1}_i \equiv \hat{\psi}_i + 0.5\Delta t\tilde{R}^{n+1}_i,
\]

where \( \psi^{n+1}_i \) is the solution sought at the grid point \( (\bar{r}^{n+1}, \bar{x}_i) \), \( \hat{\psi}_i \equiv \psi^n + 0.5\Delta tR^n \), and \( LE \) denotes a two-time-level either advective semi-Lagrangian [46] or flux-form Eulerian [47].

\(^6\) Note that for an ideal fluid, the potential temperature is isomorphic with the specific entropy, as then \( s = e_p \ln \theta; \) in the compressible clone of EULAG, either total or internal energy equations may be solved in lieu of (4).

\(^7\) "Nonoscillatory forward-in-time" labels a class of second-order-accurate two-time-level algorithms built on nonlinear advection techniques that suppress/reduce/control numerical oscillations characteristic of higher-order linear schemes; and it is meant to distinguish these algorithms from classical centered-in-time-and-space linear methods.
NFT transport operator (viz. advection scheme such as MPDATA).\(^8\)

For inviscid adiabatic dynamics, Eq. (9) represents a system implicit with respect to all dependent variables in (3) and (4), because all principal forcing terms are assumed to be unknown at \(n + 1\).\(^9\) For the physical velocity vector \(\mathbf{v}\), it can be written compactly as

\[
\mathbf{v}_i = \hat{\mathbf{v}}_i - 0.5\Delta \tau (\hat{\mathbf{G}}(\nabla \mathbf{p}'))_i + 0.5\Delta \tau \mathbf{R}_i(\mathbf{v}, \hat{\theta}'_i),
\tag{10}
\]

where

\[
\mathbf{R}_i(\mathbf{v}, \hat{\theta}'_i) \equiv -\frac{1}{\theta_o} (\hat{\theta}'_i - 0.5\Delta \tau ((\hat{\mathbf{G}}^T \mathbf{v}) \cdot \nabla \theta_o)_i) - (\mathbf{f} \times (\mathbf{v} - \mathbf{v}_c))_i,
\tag{11}
\]

accounts for the implicit representation of the buoyancy via (4), and the superscript \(n + 1\) has been dropped as there is no ambiguity. Separating (11) into the explicit and implicit part

\[
\mathbf{R}_i(\mathbf{v}, \hat{\theta}'_i) = \mathbf{R}_i(\mathbf{v}_c, \hat{\theta}'_i) + \frac{\Delta \tau}{\theta_o} ((\hat{\mathbf{G}}^T \mathbf{v}) \cdot \nabla \theta_o)_i - (\mathbf{f} \times \mathbf{v}_c)_i,
\tag{12}
\]

and denoting the sum of the second and third term on the rhs as \(\mathbf{\tilde{R}}_i(\mathbf{v}, \hat{\theta}_i)\), allows one to write (10) as

\[
\mathbf{v}_i = \hat{\mathbf{v}}_i - 0.5\Delta \tau (\hat{\mathbf{G}}(\nabla \mathbf{p}'))_i + 0.5\Delta \tau \mathbf{\tilde{R}}_i(\mathbf{v}, \theta_o),
\tag{13}
\]

where \(\hat{\mathbf{v}}\) is a shorthand for all explicit terms on the rhs of (10). On grids co-located with respect to all prognostic variables, (13) can be inverted algebraically to construct expressions for the solenoidal velocity components, via (6), that are subsequently substituted into (2) to produce an elliptic equation for pressure

\[
\left\{ \hat{\Delta} \rho^* \nabla \cdot \rho^* \mathbf{G}^\top (\mathbf{\hat{v}} - (1 - 0.5\Delta \tau \mathbf{R}^{-1}) \mathbf{G}(\nabla \mathbf{p}')) \right\}_i = 0,
\tag{14}
\]

where \(\hat{\mathbf{v}}\equiv (1 - 0.5\Delta \tau \mathbf{R}^{-1}) \mathbf{G}(\nabla \mathbf{p}');\) cf. [38] for the complete exposition. Boundary conditions imposed on \(\hat{\mathbf{v}} \cdot \mathbf{n}\), subject to the integrability condition \(\int_{\partial \Omega} \rho^* \hat{\mathbf{v}} \cdot \mathbf{n} \, d\sigma = 0\), imply the appropriate boundary conditions for \(\mathbf{p}^*\) [38,65]. The resulting boundary value problem is solved using a preconditioned generalized conjugate residual GCR algorithm [48,53] – a nonsymmetric Krylov subspace solver akin to the popular generalized minimum residual GMRES scheme [11,44]. Given the updated pressure, and hence the updated solenoidal velocity, the updated physical and contravariant velocity components are constructed from the solenoidal velocities using transformations (6) and (5), respectively.

3. Advanced features

3.1. Implicit \(\theta\) integrals

Adverting \(\hat{\theta}'\) in the entropy Eq. (4), and treating the convective derivative of the ambient state as an implicit forcing on the rhs, has significant benefits for computational accuracy and stability that greatly exceed the associated intellectual and technical difficulty of the ultimate model algorithms (10) and (11) and the resulting elliptic problem.
14. In fact, these two features are among the key ingredients of EULAG’s multi-scale capability, admitting simulations from micro to planetary scales within a single numerical framework. In addition to stable integrations of the terms responsible for gravity-wave propagation in anisotropic inhomogeneous media, these features allow conservation of entropy perturbations with accuracy to round-off error (see Section 3a in [50] for a discussion), and prevent dilution of the ambient stratification due to implicit viscosity of nonoscillatory advection. The latter is particularly important for long-time integrations of turbulent estuarine/oceanic flows [64]. Further benefits include improved accuracy of impermeability conditions imposed along irregular boundaries as well as the improved energetics of the model [66].

A specific perturbational form of PDEs solved in EULAG depends on the admitted class of ambient states. Consider for illustration that the standard governing system (2)-(4) derives from the generic anelastic form, in which the pressure and temperature perturbations in the momentum equation are taken with respect to the static vertical profiles of the generalized Boussinesq expansion, simply by postulating there exists an inertial ambient state determined by the balance of pressure, buoyancy, Coriolis and metric forces

\[ 0 = -\mathbf{G}(\nabla(p_\eta - p_0)) - \mathbf{g}\frac{\theta_\eta - \theta_0}{\theta_\eta} - \mathbf{f} \times \mathbf{v}_\eta + \mathbf{M}, \]  

(15)

together with implied compatibility conditions (e.g., for a sheared ambient flow, \( \theta_\eta \) must change accordingly in the horizontal; cf. [50] for discussion). Subtracting (15) from the generic form results in (3), while the perturbational formulation of the entropy Eq. (4) follows readily the generic form \( \delta \eta / \delta t = \mathcal{F} \), given \( \theta_\eta = \theta_\eta(x) \).

In general, ambient states can support any subset of the solution to the full problem. There are substantial mathematical and computational benefits of using carefully selected ambient states. In particular, the use of ambient states reduces the stiffness of the problem at hand, by subtracting known balances a priori from the full problem. Even the relatively simple inertial state assumed in (15) conveniently circumvents the need to specify the ambient pressure – so initializing the model with \( \mathbf{v}(t = 0) = \mathbf{v}_\eta - \nabla \phi \) (where \( \phi \) is a potential determined from the mass continuity and boundary conditions) assures well posed initial conditions for a broad class of engineering and environmental flows; cf. [61] and references therein for a discussion. Furthermore, a reduced stiffness is tantamount to a reduced condition number (of the problem at hand), thus eventually accelerating convergence of the elliptic solver in the model.

3.2. Vertical and horizontal mappings

The combination of the tensor formulation [38,54,40] and NFT numerics underlying EULAG admits a class of almost global diffeomorphic mappings given by (1) with \( \mathcal{F}(t, \mathbf{x}) = (\tilde{x}(t,x,y), \tilde{y}(t,x,y), \tilde{z}(t,x,y,z)) \). (16)

The current implementation of the vertical mapping in EULAG involves only a subset of (16); i.e., the similarity transformation [65]

\[ \tilde{z}(t,x,y,z) = \tilde{z}(\zeta), \quad \zeta = H_0(z - z_1)/(H - z_1), \]  

(17)

where \( z_1(t,x,y) \) is the lower boundary position, \( H_0 \) is a representative domain depth, and \( H(t,x,y) \) is the upper boundary of the domain. This similarity transformation is both elegant and computationally efficient. However, (17) does not readily allow surfaces of constant \( \tilde{z} \) to vary arbitrarily in \((t,x,y,z)\) subject only to the continuity assumptions in (16).

In contrast, the horizontal mappings utilized in the code may be any of those allowed by the general form (16). Inspection of this general form reveals that the horizontal mapping can be any time dependent, spatially 2D transformation, subject to the required continuity constraints. The horizontal mapping does not depend upon the vertical coordinate. Historically, this feature arose from the geophysical setting in which EULAG was first developed. It keeps vertical columns vertical, thus following the basic hydrostatic structure of geophysical flows. It has the added benefit of simplifying metric terms and the model code (see Footnote 10); as well as keeping the CPU overhead for GA (grid adaptivity) negligible [38,40]. In the applications tested so far, the 2D nature of the horizontal mapping has not been a serious limitation. Both horizontal and vertical grid mappings may be computed numerically [38,33,65] (Fig. 3) or specified analytically [38,54,40] (Fig. 6). In addition to any number of globally smooth mappings, of special interest is that the GA machinery in EULAG may employ sudden step function changes in resolution, as in nested grids, without any apparent ill effects [38,40].

To date, our experiences are that beneficial GA mappings and targeting algorithms are highly application dependent and may need to contain stationary elements as well as dynamic ones [40]. No single, simple GA mechanism appears to work well all the time. For example, grid adaptation is relatively straightforward in more elementary applications such as in flows past “irregular” boundaries [63,15,62,16,35,2,54]; free surface flows [34,22,65,33]; or in tracer transport [7,14,2,21]. In these examples focusing the GA on boundary shape or targeting various combinations of tracer magnitude, gradient and/or second derivatives of the tracer field have all produced good results. It is not at all clear how to adapt in more complicated applications such as climate prediction. In climate the targeting choices for GA are far more varied. Major geographical features such as mountain ranges, valleys,
3.3. The tensor formalism

Grid adaptivity in the model is enabled by the mapping (1), (16) and (17). It enables the resolution of an extended range of scales in a single computational domain as well as more accurate treatment of boundary conditions. A signature characteristic of EULAG is that a rigorous tensor formalism underlies the application of this mapping. This formalism offers additional benefits beyond those of GA, per se. It also provides guidance on numerical developments that improve accuracy, such as in choices of the variables to work with and the forms of the governing equations they must satisfy (e.g., for velocity see [38], for vorticity see [39,54]).

Invariance refers to the property that geometric or physical objects do not change character as the point of view of an observer moves from one system of coordinates to another [31]. For example, an observer on the Earth finds that wind magnitude and direction are independent of coordinates – they are invariants. Mass conservation provides another example of invariance; the mass of a closed system must remain fixed. More generally, conservation of many properties is central to the development of computational models. But with the addition of GA the computation of requisite metric terms can make otherwise conservative numerical methods nonconservative. Thomas and Lombard [62] first developed a way to maintain conservation in curvilinear coordinates by utilizing a geometric conservation law (GCL) that maintains the conservation of space. In EULAG, the GCL has been upgraded into the multi-component tensor identity [37,38,40]:

\[ \frac{G}{G} \frac{\partial}{\partial \mathcal{F}} \left( \frac{\partial}{\partial \mathcal{F}} \right) = 0, \quad (18) \]

where \( \mathcal{F} = (i,x) \) and \( \mathcal{F} = (i,x) \). It is a purely geometrical constraint that is formally independent of physics and/or numerical methods. The tensor GCL is trivially satisfied only when uniform grids are utilized in \( D_p \). In generalized coordinates, we employ a variant of (18) to relate the derivatives of the Jacobian matrix to those of the Jacobian, that is, specific metric terms needed by EULAG are computed using derivatives of \( (G/G) \) [65,40]. This is distinctly different from the approach advanced in [62] wherein a similar variant of the GCL is employed as an evolution equation to compute the Jacobian.

Another tensor identity that is ubiquitous in tensor manipulations involves the Kronecker delta \( \delta^m_{\eta} \) [39,54,40], which gives the elements of the product \( \left( \frac{\partial}{\partial \mathcal{F}} \right) \left( \frac{\partial}{\partial \mathcal{F}} \right) \). In principle, \( \delta^m_{\eta} = 0, 1 \) depending upon whether \( i \neq m \) or \( i = m \), respectively, and generates a system of differential identities that relate the elements of the Jacobi and inverse Jacobi matrices of the transformation. Given the mapping (16), there are 20 nontrivial metric coefficients in \( \partial \mathcal{F}/\partial x \) and \( \partial \mathcal{F}/\partial x \), \( \delta^m_{\eta} \) provides 10 non-trivial differential identities relating these metric coefficients and the GCL (18) provides an additional four. Together, the tensor GCL and \( \delta^m_{\eta} \) differential identities are equivalent to the transformation laws for a contracted form of Christoffel symbols of the second kind [40].

These tensor identities lie at the core of EULAG. They provide fundamental guidance for the general property of tensor invariance – as the use of the Kronecker delta and the transformation law for Christoffel symbols are ubiquitous in standard tensor manipulations. The most significant idea here (unproven) is that by satisfying a core set of tensor identities, a large number of invariants will be satisfied automatically to round-off error, provided the numerics satisfy those invariant properties in Cartesian coordinates. For illustration, consider that deriving the generalized vorticity equation from the momentum equation generates “extra” terms that should vanish identically according to specific tensor identities. Consequently, violating those identities in the numerical model will result in spurious vorticity production and associated artifacts such as fictitious boundary layer separations and drag in potential flows [65,54].

4. Model performance

4.1. Numerical accuracy

EULAG’s ability to simulate a broad range of fluid problems raises questions regarding suitable accuracy standards, valid throughout the range of diverse applications. From the perspective of formal numerical analysis the convergence rate should be the only guiding principle. Indeed, for sufficiently smooth problems the model solution must converge as implied by the underlying numerics; e.g., for demonstration of the second-order convergence rate see [60], where an MPDATA-based NFT (nonoscillatory forward-in-time) approach has been employed for integrating compressible Euler equations on unstructured meshes. However, for the class of high-Reynolds number flows emphasized in this paper, the formal convergence is a moot issue, as the resolutions affordable in practice are still far from the DNS limit, whereupon the only convergence one may demand from either LES or ILES (implicit large-eddy-simulation) is for the statistics and morphological structure.
of the turbulence; e.g., for a demonstration of the LES convergence of EULAG’s energy spectra see [26].

Fig. 1 substantiates the issue. It shows remarkable agreement of the NFT and the pseudo-spectral calculations, the latter method traditionally prized for its accuracy and considered a superior tool for studying turbulent flows. For DNS ($v > 0$) all solutions agree uniformly; see [19,52,26] for discussions and physical insights. Without viscous dissipation, in the infinite Reynolds number limit, unbounded enstrophy growth is predicted (dotted line; cf. Chapter VII.7 in [23]), with a finite enstrophy blowup time of $t \sim 0.55$ s. With rapid enstrophy growth, the spectral calculations become computationally (nonlinearly) unstable and must be terminated after $\sim 0.35$ s [19]. Up to this time, the NFT, spectral, and theoretical results agree closely, attesting to the accuracy of the NFT approach. As the EULAG calculations proceed in the ILES mode, a difference emerges between the Eulerian (flux-form) and semi-Lagrangian (advective) calculations suggesting, seemingly, a larger dissipation in the latter. It turns out that the related idea of the effective viscosity for numerical simulations of turbulent flows is involved [52,8], and that dissipative properties of the Eulerian and semi-Lagrangian schemes are rather problem dependent and distinct [49,52]. In particular, the finite volume NFT schemes appear more suitable for simulating high-Reynolds-number flows [3], arguably because their flux-form design is consistent with finite-volume representation of the fluid equations [28]. The Kolmogoroff function displayed in Fig. 5 atests to the quality of the Eulerian ILES calculations, showing that overall the prediction for the inertial range spectrum is of the same quality as observed in LES performed with standard eddy viscosity models as well as in high resolution DNS [8].

Fig. 7 illustrates the effects of grid resolution on turbulent flow morphology with ILES simulations of the breaking of deep internal gravity waves in the Earth’s atmosphere [35]. The middle panel of Fig. 7, with a standard resolution of 625 m, depicts an isolated region of wavebreaking centered at an altitude of $\sim 90$ km and $\sim 50$ km horizontal. The waves are being forced by a traveling disturbance along the lower boundary that is a proxy for a narrow squall line. The waves travel upward and horizontally in both directions; and increase in amplitude with time and altitude. When wave amplitude approximately matches the vertical wavelength, they break. For the standard resolution, breaking begins suddenly in a very localized region at 145 min. The region of wavebreaking grows in time to encompass most of the domain above $\sim 60$ km altitude. The leading edges of the wavebreaking region (below and to the right) constitute a breaking front that consists of regions of vigorous wave overturning, yet recognizable wave structure. Behind the propagating breaking front lies the turbulence field. This interior region is characterized by much finer structures devoid of any coherent waves. The top and bottom panels show the effects of reduced and enhanced grid resolution, respectively. The higher resolution result is similar to the standard one, albeit the wavebreaking region is somewhat larger, and highly localized grid sized structures (micro details) do not match in regions where the gravity waves have broken and vigorous mixing has occurred. The reduced resolution result shows only the nearly monochromatic, linear wavefield that precedes breaking. It is clear that the standard resolution matches the higher resolution case much more closely than the lower resolution result. Thus the figure demonstrates convergence of the morphological structure of the turbulence. This interpretation is further corroborated by detailed spectral and velocity derivative skewness analyses – useful in gauging the inhomogeneity and anisotropy of the turbulence; for which high resolution results are given in [37,36].

4.2. Parallel performance

Here, we describe the performance of EULAG on massively parallel processor architectures. Distribution of the overall computational load to multiple processors is...
accomplished using a two-dimensional horizontal decomposition of the grid. Locally, each processor is responsible for a subset of the total horizontal grid and all of the associated vertical nodes. Each subset is explicitly dimensioned to also contain an extension overlapping part of the subset belonging to neighboring processors. These extra boundary points are referred to as “ghost”, “guard” or “halo” cells, and are used to minimize communications needed when horizontal finite difference operations are performed. When necessary, halo cell information is updated with fresh data distributed between neighboring processors. The process of data exchange needs to be efficiently balanced to keep each processor’s work rate at the same level. The number of halo cells depends on the local stencils used in the model algorithm; and for the semi-Lagrangian option, also on the maximum Courant number expected for the problem at hand [37].

To examine code scalability in highly anisotropic global applications, a number of experiments with HS (Held–Suarez) idealized climates [50] and solar MHD (magnetohydrodynamics) [4] were performed. The HS tests were conducted on two IBM POWER PC systems – Bluevista and Seaborg. Both machines use processors (PEs) that perform four floating-point operations per cycle. The MHD tests where conducted on three Linux clusters – Altix, Sherbrooke, and a CRAY XD1. Details of each machine’s hardware and peak performance are listed in Table 1.

The scalability of the HS tests on Bluevista are depicted in Figs. 8 and 9. In Fig. 8 the bold dashed line represents ideal scalability, whereby the wall clock time scales like 1/NPE where NPE is the number of processors used. Here the wall clock times have been divided by the CPU time using the one processor result. The results show excellent scalability up to NPE ~ 16 PE’s for the (64 x 32) grid; 64 PE’s for the (128 x 64) grid; and 256 PE’s for the (256 x 128) grid. The corresponding speedups in wall clock times at these processor numbers are 20×; 90×; and 205× for the three grid sizes, respectively. With larger numbers of processors, one is tempted to think that this weakening of the scalability is due to an increased ratio in the amount of information exchanged between processors versus the amount of local computations. This seems even more likely given that EULAG uses a semi-implicit iterative elliptic solver which requires global type parallel operations (sums). While the iterative solver can be expensive [50], its performance has been found to be relatively insensitive to NPE and appears not to be the primary reason for the weaker scalability with larger numbers of processors.

Instead, we find the scaling performance on Bluevista is very sensitive to the particular grid decomposition used for a given number of processors (NPX x NPY = NPE). Fig. 9 shows results from all of the HS simulations, using the (256 x 128) grid, on Bluevista in the form of total CPU time (wallclock time × NPE) vs. number of grid points (mp = M/NPY) in the y-direction. With the exceptions of small numbers of processors (NPE = 1, 2, 4); each NPE value is associated with a family of four or five simulations that differ according to the processor geometry. Together Figs. 8 and 9 show that, approximately, as long as NPE ≤ M/(N PE) and np, mp ≥ 16, then the results scale well. This suggests that scalability is limited strongly by cache performance. For all the NPE cases except one, it was possible to choose processor geometry such that both np, mp ≥ 16, and all those results showed excellent scalability. The only family of simulations for which this was not possible was NPE = 256 (mp = 8); and that run did not scale quite as well. Nevertheless, it still ran at 100 GFlops, or 5.1% of the peak speed. With NPE = 64, however, the best scaling configuration ran at 40 GFlops or 8.2% of the peak speed. With careful attention to optimization, we have been able to reach peak speeds of 10.3% with NPE = 64 on Bluevista.

Table 1

<table>
<thead>
<tr>
<th>Computer</th>
<th>NPE&lt;sub&gt;max&lt;/sub&gt;</th>
<th>PE type</th>
<th>PE Speed</th>
<th>PE Flops</th>
<th>Total Flops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bluevista&lt;sup&gt;a&lt;/sup&gt;</td>
<td>624</td>
<td>POWER5</td>
<td>1.9 GHz</td>
<td>7.6 G</td>
<td>4.7 T</td>
</tr>
<tr>
<td>Seaborg&lt;sup&gt;b&lt;/sup&gt;</td>
<td>6080</td>
<td>POWER3+</td>
<td>0.38 GHz</td>
<td>1.5 G</td>
<td>9.1 T</td>
</tr>
<tr>
<td>Sherbrooke&lt;sup&gt;c&lt;/sup&gt;</td>
<td>1152</td>
<td>Xeon+</td>
<td>3.6 GHz</td>
<td>7.2 G</td>
<td>9.1 T</td>
</tr>
<tr>
<td>CRAY XD1&lt;sup&gt;d&lt;/sup&gt;</td>
<td>84</td>
<td>AMD</td>
<td>2.8 GHz</td>
<td>5.6 G</td>
<td>0.47 T</td>
</tr>
<tr>
<td>Opteron</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Altix 4700&lt;sup&gt;d&lt;/sup&gt;</td>
<td>384</td>
<td>Dual-Core</td>
<td>1.6 GHz</td>
<td>12.7 G</td>
<td>4.9 T</td>
</tr>
<tr>
<td>Itanium2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NPE<sub>max</sub> is the total number of processors available in the machine; PE Type specifies the processor type; PE Speed is the processor clock speed; PE Flops gives the peak floating point operations per second (GFlops) per processor – for the Altix, PE Flops is the combined value from both cores, Total flops = NPE<sub>max</sub> × PE flops gives the peak computational speed of the entire machine (TFlops).

<sup>a</sup> National Center for Atmospheric Research, Boulder, CO.
<sup>b</sup> National Energy Research Scientific Computing Center, Berkeley, CA.
<sup>c</sup> Université de Sherbrooke, Québec, Canada.
<sup>d</sup> Université de Montréal, Québec, Canada.

14 This is highly application dependent [53]; for the HS test problem it consumes ~80% of the total computing time.
Fig. 8. Held–Suarez runs on IBM POWER5 machine Bluevista, using three different horizontal grid sizes \((N \times M)\). Each test case used the same number of vertical levels \((L = 41)\) and employed variable horizontal grid increments (e.g., utilized GA); all other parameters were identical including timestep.

Fig. 9. Effect of domain decomposition on scaling performance. Held–Suarez simulations with \((256 \times 128)\) horizontal grid on Bluevista.
The strong scaling results for Bluevista can immediately be inferred from the data given in Fig. 8 by multiplying the relative times by NPE. This exercise indicates nearly ideal performance within the NPE limit given previously. Although not shown, the weak scaling behavior on Bluevista is very similar to that shown in the left panel of Fig. 10. For the range of data depicted in Fig. 8, weak scaling results yield horizontal lines, typically to within ±5%.

We conclude this section by showing weak and strong scaling results from the IBM POWER3+ machine Seaborg in Fig. 11. Generally EULAG’s performance on Seaborg was similar to that found on Bluevista, only 5–10× slower (clock cycle considerations alone account for a factor of 5). Scaling performance on Seaborg was qualitatively similar to that shown in Figs. 8 and 9, but the falloff with larger values of NPE was somewhat faster. Typically np, mp ≥ 32 were required for good scaling behavior. Following the format for the Linux machines, the left and right panels of Fig. 11 show weak and strong scaling results, respectively. The weak scaling results show two sets of data – with the top set using only half as many PEs for each grid size as the bottom data set. They show a nearly linear increase (i.e., constant positive slope) in wallclock time with log(NPE); in contrast to the Linux and Bluevista results which show more ideal scaling (i.e., zero slope).

Both sets of data indicate that the largest grids required approximately twice the wallclock time of the smallest grids. Compared to the other machines, this change in scal-

---

Fig. 10. MHD tests; idealized solar magneto-convection on a 256 × 256 × 200 Cartesian grid. Weak and strong scaling on Linux clusters Sherbrooke (Mp), XD1, and Altix. Note results based upon Portland compiler in addition to Intel compiler on Sherbrooke and XD1.

Fig. 11. Weak and strong scaling on IBM POWER3+ machine Seaborg. Results are based upon HS experiments using horizontal grid sizes ranging from (128 × 64) to (2048 × 1024). The range in the number of processors employed is 4 ≤ NPE ≤ 2048.
ing behavior may be due to differences in machine hardware, especially the interconnection between processor nodes. Strong scaling results shown in the right panel of Fig. 11 correspond closely to those for Bluevista, and show better performance than the corresponding Linux machines. We estimate that the largest problem with \( N \times M = 2048 \times 1024 \) gridpoints and NPE = 2048 ran at \( \sim 150 \) GFlops, or 5% of the peak speed.

5. Remarks

EULAG is a computational model for simulating thermo-fluid flows over a wide range of scales and physical processes. It is noteworthy for its NFT (nonoscillatory forward-in-time) numerics formulated for arbitrary curvilinear frameworks, robust elliptic Krylov solver, GA (grid adaptivity) technology, and scalability on massively parallel processor machines. Published applications of EULAG range from millimeter scales in sub-meter cloud microphysics studies [1], up to \( 10^4 \text{–} 10^5 \text{ km in} \), respectively, planetary atmospheric [50,58] and solar convection studies [12]. The concomitant physical processes are diverse. The model has been used for canonical laboratory studies of turbulence and wave-wave interactions [52,66,67]; to provide a standard of comparison in immersed boundary problems (e.g. urban flows [45,59]; Fig. 2); and is capable of resolving topographically forced waves when simulating global atmospheric flows [40].

This level of flexibility stems from several sources: the NFT numerics, the implicit treatment of the convective derivative of the ambient state along with the advection of perturbation potential temperature, a parallel coding structure that scales well on massively parallel processor machines, and the use of GA technology to better resolve finer scales in selected regions of the computational domain. These last two characteristics in particular allow EULAG to simulate flow problems employing a considerably larger range of scales than would otherwise be possible in a single computational domain.

The tensor formulation guides the numerics and helps to provide identities important for numerical invariances (such as conservation properties). Conversely, numerical developments have shown that some forms of the tensor equations are considerably more useful than others in building an efficacious model. The “cleanest” tensor expression of a given principle may not be the most numerically useful form. In summary, we find that good design in a computational model requires a close, synergistic interaction between the analytical and numerical aspects.

Finally, we note that while the numerical algorithms of EULAG were developed employing structured grids, they are in essence independent of spatial discretization. The potential of EULAG’s NFT integration methods for unstructured-mesh approximations have been realized only recently. Bacon et al. [2] pioneered with an implementation of MPDATA advection in the multi-scale environmental model OMEGA for operational forecast of weather and pollutant dispersion. Independently, Margolin and Shashkov [29] drew inspiration from the MPDATA approach to develop a second-order, sign-preserving conservative interpolation for remapping two-dimensional arbitrary Lagrangian–Eulerian (ALE) grids. Aiming at a broad range of applications involving complex, multiply-connected domains and inhomogeneous flows, Smolarkiewicz and Szmelter [55] developed a general, compact edge-based fully unstructured-mesh formulation of MPDATA and demonstrated its potential for adaptive NFT simulations of high-speed aerodynamics flows [60].

Acknowledgements

This work was supported in part through the US Department of Energy (DOE) Climate Change Prediction Program. Computer support was provided by the US National Energy Research Scientific Computing Center (NERSC), sponsored by the DOE; the US National Center for Atmospheric Research (NCAR), sponsored by the US National Science Foundation; and by the Réseau Québécois de Calcul de haute Performance (RQCHP), Canada. We also express our appreciation and thanks to Jacques Richer, Michél Béland, and Mihai Ghizaru for making available the MHD scaling results; and to two anonymous reviewers that helped us improve the presentation.

References


