Arch Computat Methods Eng (2016) 23:673–722
DOI 10.1007/s11831-015-9152-1

ORIGINAL PAPER

A Review of Element-Based Galerkin Methods for Numerical Weather Prediction: Finite Elements, Spectral Elements, and Discontinuous Galerkin

Simone Marras[1] · James F. Kelly[2] · Margarida Moragues[3] · Andreas Müller[1] · Michal A. Kopera[1] · Mariano Vázquez[3,4] · Francis X. Giraldo[1] · Guillaume Houzeaux[3] · Oriol Jorba[5]

Received: 1 April 2015 / Accepted: 21 April 2015 / Published online: 19 May 2015

Abstract Numerical weather prediction (NWP) is in a period of transition. As resolutions increase, global models are moving towards fully nonhydrostatic dynamical cores, with the local and global models using the same governing equations; therefore we have reached a point where it will be necessary to use a single model for both applications. The new dynamical cores at the heart of these unified models are designed to scale efficiently on clusters with hundreds of thousands or even millions of CPU cores and GPUs. Operational and research NWP codes currently use a wide range of numerical methods: finite differences, spectral transform, finite volumes and, increasingly, finite/spectral elements and discontinuous Galerkin, which constitute element-based Galerkin (EBG) methods. Due to their important role in this transition, will EBGs be the dominant power behind NWP in the next 10 years, or will they just be one of many methods to choose from? One decade after the review of numerical methods for atmospheric modeling by Steppeler et al. (Meteorol Atmos Phys 82:287–301, 2003), this review discusses EBG methods as a viable numerical approach for the next-generation NWP models. One well-known weakness of EBG methods is the generation of unphysical oscillations in advection-dominated flows; special attention is hence devoted to dissipation-based stabilization methods. Since EBGs are geometrically flexible and allow both conforming and non-conforming meshes, as well as grid adaptivity, this review is concluded with a short overview of how mesh generation and dynamic mesh refinement are becoming as important for atmospheric modeling as they have been for engineering applications for many years.

✉ Simone Marras
smarras1@nps.edu

[1] Department of Applied Mathematics, Naval Postgraduate School, 833 Dyer Rd., SP249A, 93943 Monterey, CA, USA

[2] Exa Corporation, Burlington, MA, USA

[3] Barcelona Supercomputing Center BSC-CNS, CASE, Barcelona, Spain

[4] IIIA - CSIC, Bellaterra, Spain

[5] Barcelona Supercomputing Center BSC-CNS, Earth Sciences, Barcelona, Spain

1 Introduction

Numerical weather prediction (NWP), which began with the work of Richardson during World War I [250], remains one of the most challenging problems in the computational sciences. The two main challenges to producing an accurate forecast are (1) mathematically modeling atmospheric phenomena over a wide range of physical and temporal scales (e.g., turbulence, radiation, cloud formation), and (2) harnessing the available computational resources to evaluate these models in an accurate and efficient manner. While the goal of the first challenge is probably static (that is, a comprehensive mathematical description of the atmosphere at a given time), the second challenge represents a moving target. Computational resources not only expand; they change in character. Richardson's original idea of a "forecasting factory" consisting of thousands of human computers assembled in an amphitheater was never realized; the first NWP codes were implemented on mainframe computers. Mainframes gave way to minicomputers and later vector supercomputers such as the Cray 1, 2, X-MP, and Y-MP. By the mid-1990s, vector supercomputers were replaced by massively parallel distributed systems. Now, in 2015, we are seeing the proliferation of many-core architectures (e.g. GPUs) and hybrid distributed/shared memory

architectures (e.g. clusters of many-core processors, heterogeneous clusters). Moreover, as models increase their accuracy by resolving more phenomena (e.g. resolving non-hydrostatic effects, incorporating more complex moisture parameterizations), their appetite for high performance computing (HPC) resources grow.

The modeling challenge and computational challenge meet in the choice of the numerical method used to discretize the underlying continuum model(s), which are generally expressed as systems of both partial and ordinary differential equations. The numerical model, as this figurative middle-man, must both (1) accurately represent the continuum model, and (2) efficiently utilize the hardware used to implement the numerical method. Hence, the numerical method mediates these two grand challenges by adapting to the hardware; moreover, since NWP models may take on the order of 100 man-years to develop, test, and deploy, the designers of the numerical method should target their model to *future* HPC resources. Just as biological organisms must constantly adapt to their physical environment, numerical methods must adapt to their computational environment, competing for available resources. A natural question arises: which numerical methods will survive and flourish, and which will stagnate, decline, and perhaps go extinct?

This question was partially addressed in the review of the numerical methods for non-hydrostatic atmospheric modeling reported by Steppeler et al. [282]. Based on some of the questions posed in [282], we concentrate on a class of numerical methods that may emerge victorious in next generation atmospheric (and climate) models: element-based Galerkin methods (EBGs). Among other questions, Steppeler and co-workers asked whether the numerical error caused by terrain-following coordinates could be avoided by means of z-coordinate based methods [281, 282]; element-based Galerkin methods are a natural choice to fulfill this recommendation. Furthermore, they questioned the ability of low order methods to resolve certain phenomena at high resolution without affecting accuracy: "Experience from current models suggests that approximations of overall third order will be adequate." It is shown in this review how things have indeed evolved towards the high order approach that Steppeler et al. were discussing 10 years ago and how those schemes that in 2003 had not been used in operational mode (because considered "advanced" [281]), are currently the driving force behind the next generation NWP models.

As discussed above, element-based Galerkin schemes today are tied to their relationship with the evolution of computer hardware. We will see this in the sections that follow, after giving a short overview of the current trends in HPC and how atmospheric models are developing around this paradigm.

1.1 Trends in High Performance Computing

Twenty-five years ago (1990), state-of-the-art HPC were the Cray supercomputers (e.g. Cray Y-MP). These machines had a small number (2–8) of expensive custom vector processors, which perform a single instruction on multiple data (SIMD); all the processors fetched data from a bank of *shared* memory. This trend changed in the 1990s as commodity processors and memory became relatively inexpensive; suddenly, large clusters of commodity processors that utilized *distributed* memory architectures became available. Unlike the vector machines, distributed memory systems require communication between independent processes. At the present time (2015) another shift is occurring as many-core architectures, with a relatively small amount of shared memory, are being coupled with massively parallel systems. These distributed memory systems eclipsed the older vectorized machines by the late 1990s, and vectorized machines are no longer used in HPC.

Today, HPC is in the Petascale era, with core counts exceeding $\mathcal{O}(10^6)$ [226] while exascale technologies are rapidly approaching. For instance, the largest cluster as of November 2014 (Top500[1]) is Tianhe-2 with 3.12 million cores and a maximum LINPACK [80] performance of 33.8 PetaFLOPS. The next largest machine is Titan, a Cray XK7 with 560,640 cores and a maximum LINPACK performance of 17.59 PetaFLOPS. To take full advantage of the performance of these architectures, the need for specific characteristics in new models drove scientists from different fields to go back to the design board and start from scratch in the construction of their numerical algorithms [118]. This is required by the need for very specific features that the numerical method must have to reach very high levels of scalability on the new machines. The next section reports on most operational and research atmospheric models developed until today with special emphasis on how atmospheric modelers are moving towards numerical methods that have proved more scalable on current and future computers.

1.2 Existing Atmospheric Models and NWP Systems

Table 1 shows a non-exhaustive list of atmospheric models developed until today. Most of the listed codes are based on the finite difference method. Except for ENDGame (UK Met Office), the *Nonhydrostatic Multiscale Model* core of the NCEP NAM, and *EULerian LAGrangian* (EULAG), all FD-based codes are limited area models (LAM). Spectral transform and finite volumes represent the second major trend. Codes based on the spectral transform are common

[1] www.top500.org.

Table 1 Non-exhaustive compilation of NWP systems

Model	Country	Institution	NH/HS	Type	Equations	Space	Time
ARPEGE [68]	France	Meteo France	NH/HS	LAM/GCM	HPE	ST + FD (z)	SI
ALADIN-NH [192]	France	Meteo France	NH/HS	LAM	CEE	ST + FD (z)	SISL
ETA [165]	USA	NCEP	NH/HS	LAM	HPE	FD	FB-EX
MC2 [21]	Canada	Res. Ctr. NWP	NH	LAM	CEE	FD	SISL
COAMPS [135]	USA	NRL	NH	LAM	CEE	FD	SpEx
GEM [64]	Canada	CMC & MRB	HS + NH	LAM/GCM	HPE	FEM	SISL
HIRLAM [256]	France	Meteo France	NH	LAM	HPE	FD	SISL
GFS	USA	NOAA	HS	GCM	HPE	ST + FD (z)	SI
GME [206]	Germany	DWD	HS	GCM	HPE	FV	SI
COSMO/LM [77, 281]	Germany et al.	DWD	NH	LAM	CEE	FD	SpEx
IFS [305]	UK	ECMWF	HS	GCM	CEE	ST + FEM (z)	SISL
ICON [107, 311]	Germany	MPIfM/DWD	HS/NH	GCM	CEE/HPE	FV	SI
CAM EUL [230]	USA	NCAR	HS	GCM	HPE	ST + FD (z)	SI
CAM FV [230]	USA	NCAR	HS	GCM	HPE	FV	Explicit
CAM SE [74]	USA	NCAR/SNL	HS	GCM	HPE	SE + FD (z)	EX
NAVGEM [136]	USA	NRL	HS	GCM	HPE	ST + FD (z)	SISL
ENDGame [322]	UK	Met Office	NH/HS	LAM/GCM	CEE	FD	SISL
KIAPS-GM[a]	Korea	KIAPS	HS	GCM	HPE	SEM + FD (z)	EX
NEPTUNE [117, 176]	USA	NRL	NH	LAM/GCM	CEE	SE	IMEX/SpEx
HIRAM [330]	USA	GFDL	NH	GCM	CEE	FV	SISL
ECHAM6 [283]	Germany	MPIfM	HS	GCM	HPE	ST + FD	SI
SLAV[b]	Russia	RAS	HS/(NH)	GCM	HPE	FD	SISL
JMA [261]	Japan	JMA	NH	LAM/GCM	CEE	FD	HEVI
VCAM/CCAM [218]	Australia	CSIRO	HS	GCM	HPE	FV/FD	SISL
GRAPES [325]	China	CMA	NH	LAM/GCM	CEE	FD	SLSI
NAM [166]	USA	NCEP	NH/HS	LAM/GCM	HPE	FD/FV	SI

The acronyms used in this table, some of which have not been defined before, are the following: *FD* finite differences, *FV* finite volumes, *FE* finite elements, *SE* spectral elements, *DG* discontinuous Galerkin, *ST* spectral transform, *NH* non-hydrostatic, *HS* hydrostatic, *HPE* primitive equations, *CEE* compressible Euler equations; *SISL* semi-implicit + semi-Lagrangian, *EX* explicit, *IMEX* implicit–explicit or semi-implicit, *SpEx* split-explicit, *FB-EX* forward–backward explicit, *LF* leap-frog, *NFT* non-oscillatory forward in time, *FDGC* FD on generalized coordinates, *LES* large eddy simulation, *HEVI* horizontally explicit–vertically implicit

[a] www.kiaps.org

[b] www.meteoinfo.ru

for General Circulation Models (GCM) only. High-order element-based methods (spectral element method, SEM, and discontinuous Galerkin, DG) follow, while the finite element method (FEM), only used by a handful of models, is the least common of all. For reasons that will become clearer in later sections, the temporal integration schemes that are mostly used are the split-explicit and the semi-implicit methods (Table 2).

1.3 Traditional Approaches: Finite Difference (FD) and Spectral Transform (ST) Methods

As noticeable from the tables above, most operational NWP codes in use are based on either the finite difference (FD) method, or, in the case of global models, the spectral transform (ST) method. It is difficult to find models using these methods that scale optimally on massively parallel computers (ST methods due to their all-to-all communication requirements and FD due to non-compact stencils especially at high-order). This is also true of non-compact (high-order) finite volume methods. In order to understand the strengths and weaknesses of these traditional approaches and how EBGs address some of their shortcomings, we briefly review the FD and ST methods in this subsection.

Limited area models (LAMs) consider atmospheric flows over a subsection of the earth's surface. Examples include mesoscale models, which typically span hundreds of kilometers in the horizontal, and cloud resolving models

Table 2 Non-exhaustive compilation of atmospheric research models

Model	Country	Institution	NH/HS	Type	Equations	Space	Time
TASS [243]	USA	NASA	NH	LAM, LES	CEE	FD/FV	SI
RAMS [238]	USA	Col. State U.	NH/HS	LAM	HPE	FD	LF
MM5 [83]	USA	NCAR	NH	LAM	HPE	FD	LF
ARPS [323]	USA	U. Oklah.	NH	LAM	CEE	FD	SI
OMEGA [8]	USA	Centr. Atmo. Phys.	NH	GCM	CEE	FV	SI
OLAM [310]	USA	U. of Miami	NH	GCM	CEE	FV	SpEx
NSEAM [119]	USA	NRL	HS	GCM	HPE	SE	SISL
PUMA [99]	Germany	U. of Hamburg	HS	GCM	HPE	ST	SI
HOMME [296]	USA	NCAR	HS	GCM	HPE	SE	EX + other
WRF-ARW [271]	USA	NCAR	NH	LAM	CEE	FD	SpEx
AROME[a]	Europe	Consortium	NH	LAM	ALADIN	ST	SI
EULAG [244]	USA	NCAR	NH	LAM/GCM	CEE/Incompr.	FDGC	NFT
NICAM [262]	Japan	JAMSTEC	NH	LAM/GCM	CEE	FV	SpEx
FIM [195]	USA	NOAA	HS	GCM	HPE	FV	Expl
NIM [196]	USA	NOAA	NH	GCM	CEE	FV	SpEx
UZIM [3]	USA	Co. State U.	NH	GCM	Anel.	FD	SI
DALES [134]	Netherlands	R. Nether. Meteor. I.	NH	LAM	CEE LES	FD	IMEX
CM1[b]	USA	NCAR	NH	LAM	CEE	FD	IMEX
ExnerFOAM [313]	UK	Reading U.	NH	GCM	CEE	FV	IMEX
DUNE [34]	Germany	Freiburg U.	NH	LAM	CEE	DG	EX
MPAS [273]	USA	NCAR/LANL	NH	LAM/GCM	CEE	FV	SpEx
MCore [304]	USA	U. Mich.	NH	GCM	CEE	FV	IMEX
NUMA [117, 176]	USA	NPS	NH	LAM/GCM	CEE	SE/DG	IMEX
Alya [212, 213]	Spain	BSC-CNS	NH	LAM	CEE/Incomp./Bouss.	FE	EX/Impl.
DYNAMICO[c]	France	IPSL	HS	GCM	HPE	FD	
Gung-Ho [301]	UK	Met Office	NH/HS	LAM/GCM	CEE	FE	SI/HEVI
ASAM [162]	Germany	TROPOS	NH	LAM	CEE	FD/FV	Impl.
GEOS [255]	USA	NASA	HS	GCM	HPE	FD	SISL

The acronyms, some of which were not previously introduced, are the following: *NH* non-hydrostatic, *HS* hydrostatic, *HPE* primitive equations, *CEE* compressible Euler equations, *SISL* semi-implicit + semi-Lagrangian, *EX* explicit, *IMEX* implicit–explicit or semi-implicit, *SpEx* split-explicit, *FB-EX* forward–backward explicit, *LF* leap-frog, *NFT* non-oscillatory forward in time; *FDGC* FD on generalized coordinates, *LES* large eddy simulation, *HEVI* horizontally explicit–vertically implicit

[a] www.cnrm.meteo.fr/arome/

[b] www2.mmm.ucar.edu/people/bryan/cm1/

[c] www.lmd.polytechnique.fr/dubos/DYNAMICO/

(CRMs), which span approximately up to tens of kilometers in the horizontal. See an example of a simulated single cloud in Fig. 1.

The finite difference method (FD) is the method of choice for LAMs for several reasons. First, it is simple to implement on a Cartesian grid, especially if the curvature of the earth is neglected. Unlike EBG methods, or the finite volume method, grid generation is trivial and very few ancillary data structures are needed. Second, it is very efficient on a single processor, or on a small number of processors within a shared memory architecture (e.g. vector machines). Third, constructing both upwinded and higher order discretizations is relatively straightforward, although increasing the order of accuracy may hurt its scalability due to the larger halo required.

Global models (or general circulation models, GCMs) solve the governing equations on the whole planet, which is usually approximated as a sphere. The reader is referred to the 2007 paper by Williamson [321] for a review of GCMs. Many operational GCMs utilize ST, where spherical harmonics are used to represent both diagnostic and prognostic variables on the sphere. Spherical harmonics are the natural basis functions to solve PDEs on a sphere since they are the eigenfunctions of the negative Laplacian.

Fig. 1 Large Eddy Simulation of the evolution of a single cloud with the *Nonhydrostatic Unified Model of the Atmosphere* (NUMA). From [214]. The Maya® computer graphics software was used for the photo-realistic rendering of the simulation (for more details see http://anmr.de/cloudwithmaya)

Hence, great accuracy is achieved with a minimal number of grid points on the sphere. In order to advance the dynamical equations in time using ST, it is necessary to transform between physical and spectral space; this spectral transform is evaluated using a combination of Fourier and Legendre transforms. We perform an elementary complexity analysis of the ST method to illustrate a fundamental bottleneck as the resolution of NWP models increases.

Letting n be the number of grid points, Fourier transforms are evaluated along the longitudinal (zonal) direction with an FFT with a cost $O(n \log n)$; along the latitudinal (meridional) direction, a Legendre transform is required with a cost of $O(n^3)$. Although fast Legendre methods exist, they are not widely used in NWP since they have high cross-over points. Therefore the cost of the ST method is $O(n \log n + n^3)$, which scales adversely as n increases (e.g. horizontal resolution is increased). For a grid spacing greater than 10 km, the hydrostatic, rather than non-hydrostatic equations are the governing equations in GCMs (we will touch more on the equation sets in Sect. 2). These equations are solved via a vertical mode decomposition [119] which results in a constant-coefficient Helmholtz operator. Since spherical harmonics are exact solutions to this Helmholtz operator, no matrix inversion is required. Furthermore, ST have a very small dispersion error. ST models were developed during the era of smaller, shared-memory machines which did not require communicating data across processors. As the architectures transitioned from shared to distributed memory, the communication overhead became more important; the all-to-all communication required by both the FFT and Legendre transform poses a barrier to scalability (not all distributed-memory hardware can do this operation effectively). For instance, the ST-based model NOGAPS, used by the U. S. Navy, could not scale beyond 150 processes at typical resolutions [119]. Hence, ST methods, while both highly accurate and efficient at small processor counts, cannot compete in the era of hundreds of thousands (or millions) of processors.

To overcome the limitations of FD and ST in the current era of massively parallel computers, EBG methods are becoming the new trend in atmospheric modeling for the same reason they have always been popular in other fields of computational mechanics. This alternative is justified by the proven high parallel efficiency of local methods [73, 176, 226, 320]. The efficiency of EBGs on large to very large machines is facilitated by their small parallel communication footprint. To understand this small footprint, consider Fig. 2, where the grids needed by a (a) finite element and by a (b) finite difference method are compared. In Fig. 2a, the grid consists of nine finite elements Ω_h^{el}. With EBG the solution is sought on an element-wise basis and each element communicates information to the others only through its shared boundaries (nodes in the case of CG; faces in the case of DG). When the finite element grid is partitioned into smaller portions of the global domain, the only information that needs to be exchanged among the subdomains of the partition is that on the boundary that each subdomain shares with its neighbors. In contrast, in Fig. 2b the grid is a classical structured, rectangular finite difference grid that here is plotted to be a direct analogue (in terms of node count) of the finite element mesh. Because a finite difference stencil is such that differentiation on each node in the domain requires information from a set of adjacent nodes that varies with the order of differentiation, when the domain is partitioned, some nodes will belong to two overlapping subdomains. Because of this, additional communication is necessary. In

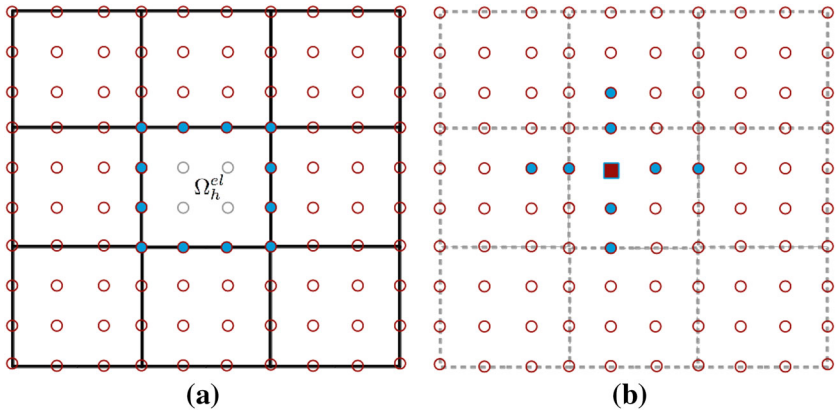

Fig. 2 Examples of the adjacency pattern for a finite element Ω_h^{el} (**a**), and for a node that belongs to a finite difference grid (**b**). In (**a**) and (**b**) information is exchanged, respectively, element- and node-wise. In (**a**), the only nodes that allow information to be shared between elements are the shared nodes on the boundary of neighboring elements (*blue dots* on the boundary of Ω_h^{el}). In (**b**), the cross made of *blue circular* nodes and a central red node is the stencil of a 4th-order differentiation performed on the central node. How these plots relate to parallelization is described in the text

the case of element-based schemes communication is naturally low by construction. The details of EBG and which models are based on them are reviewed in Sect. 3.

1.4 A Roadmap for Element-Based Galerkin Methods and This Review

Historically, finite element, spectral transform, and discontinuous Galerkin methods have been developed in relative isolation. In the past several decades, especially with the advent of spectral elements, common threads were identified. The two most important ideas are: (1) decomposing a continuous domain Ω into a finite number N_e of non-overlapping elements Ω_e and (2) expanding the state variables in N basis functions ψ_i within each sub-domain (or element) N_e. In the first operation, we express the geometry in an *element-wise* fashion; in the second operation, we perform a *Galerkin* expansion of the state variables. Hence, the moniker Element-based Galerkin method. As discussed, EBG methods are classified as either continuous (CG) or discontinuous (DG). Each of these methods may be characterized by the number of elements N_e (or equivalently, the element diameter h) and the order of the basis function p. Resolution may be increased by increasing either N_e or p independently, allowing a wide range of combinations. In the limit of $N_e = 1$, the spectral transform (ST) can be seen as an EBG method; however, being ST a degenerate EBG, in the rest of the paper it won't be considered among the EBG methods. This $h-p$ parameter space is mapped in Fig. 3. In the left panel (CG), three numerical methods are displayed: finite elements, spectral elements, and the spectral transform method. Since continuity is required between elements, the lowest possible order p is one. The finite element method (FEM) is the special case when $p = 1, 2, 3$ basis functions are employed, while the spectral transform method is recovered if a single element is used with a very large order $p \gg 1$. In the right panel, we see three numerical methods: finite volumes, DG, and the spectral transform method. Since continuity is not required between elements, we may use constant-valued basis functions, which is equivalent to cell-averaging; hence, we recover the classical finite volume (FV) method if $p = 0$. For $1 \leq p < \infty$ and $N_e > 1$ we have DG, while for large p and $N_e = 1$, we again recover the ST method. Gabersek et. al. [103] systematically mapped out the $h-p$ space for SEM. They concluded that polynomial order p between 5 and 10 with an effective resolution of $\bar{\Delta x} = h/p$ between 0.5 and 2.0 km is optimal for mesoscale simulations in terms of both accuracy and efficiency. To our knowledge, the $h-p$ space for global non-hydrostatic simulations has not been explored yet.

1.5 Scalability of EBG Methods

In the following we report on some recent scalability results of EBG on different systems and for different numerical configurations. For a more theoretical discussion on Galerkin scalability, we refer to [142, 176].

1.5.1 Scalability for (Horizontally) Explicit Time Integration

In global atmospheric simulations the vertical resolution is usually much finer than the horizontal. This leads to a much smaller time scale for vertical processes than for horizontal motion. For this reason, it is often more efficient to solve the fast processes in the vertical direction

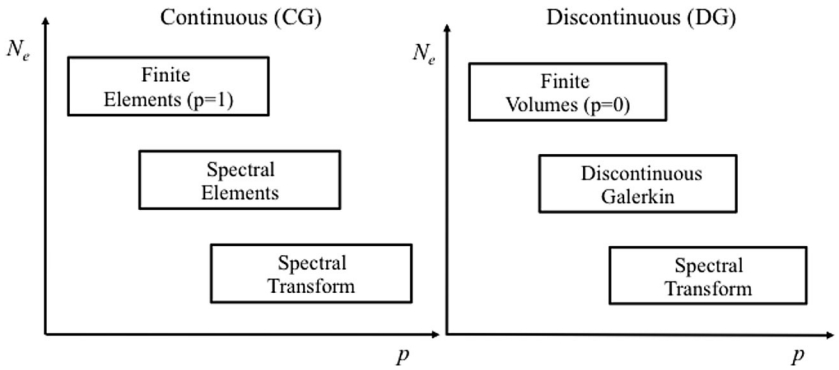

Fig. 3 EBGs are divided into two classes: continuous Galerkin (CG) methods, whose solutions are continuous with bounded weak derivative (H^1), and discontinuous Galerkin (DG) methods, whose solutions are square integrable (L^2), but not necessarily continuous. The resolution of both CG and DG methods may be characterized by the polynomial order p of their basis functions and the number of elements N_e utilized, or, equivalently, by the diameter $h \propto 1/N_e$ of each element. CG: If low order basis functions $(p = 1, 2, 3)$ are utilized with a large number of elements, we recover the classical Finite Element Method. For $p \geq 3$ and a smaller number of elements used, we have the spectral element method (SEM). As p is increased, N_e may be decreased. In the extreme case of a single element (on the sphere) and $p \gg 1$, the ST method is recovered. If we are considering problems in Cartesian geometry, this extreme case is generally termed "spectral" or "pseudo-spectral". DG: Since DG admits discontinuous solutions, a constant basis function $p = 0$ is admissible, yielding the classical finite volume (FV) method. As p is increased and N_e decreases, we enter the arena of DG methods. As with CG, if a single element is utilized, the ST method is recovered. In this extreme case, the solution becomes continuous

implicitly while using explicit time integration in the horizontal direction, or horizontally-explicit, vertically-implicit (HEVI). If a 2D domain decomposition strategy is adopted where all the elements in a vertical column are maintained on a single processor, HEVI does not incur any additional communication. A recent result for this strategy with the *Nonhydrostatic Unified Model of the Atmosphere* (NUMA) [117, 176] is shown in Fig. 4. This figure shows that NUMA achieves weak scaling up to 777,600 cores and strong scaling to about 40,000 cores; moreover, the last blue data point in this figure indicates that NUMA scales in this fashion to the limit of one horizontal element per core.

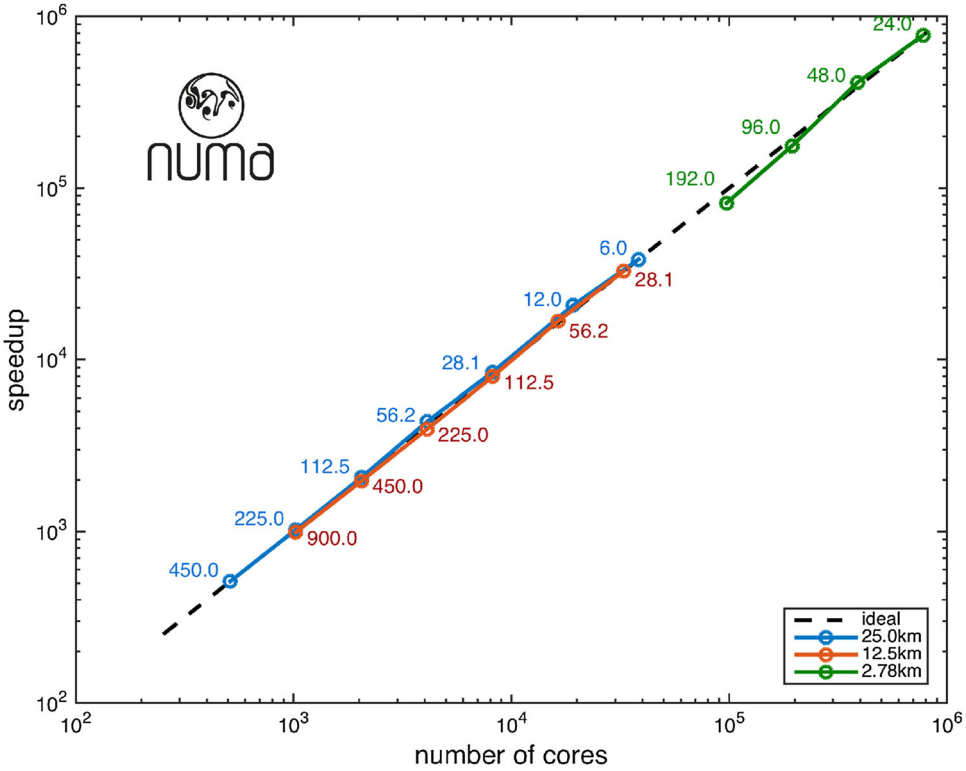

Fig. 4 Scalability study with the atmospheric model NUMA for the baroclinic instability test case [160] for three different horizontal resolutions of 25.0, 12.5 and 2.78 km (given in the legend). This scalability study was performed on the Blue Gene Mira of the Argonne National Lab. The number next to each data point shows the average number of elements per core. These simulations use a cubed sphere mesh generated by the function library p4est [42]. All simulations use six elements in the vertical direction with HEVI time integration and a fifth-order CG method

One important factor that contributed to the excellent speedup shown in Fig. 4 is that the amount of work on each core needs significantly more runtime than the time spent in communicating the data among neighboring cores. This becomes more difficult when fully explicit time integration is used and when the spatial discretization order is reduced (Fig. 5).

1.5.2 Scalability for Fully Implicit Time Integration

Scalability studies with the model Alya [142, 308] and fully implicit time integration on different machines are shown in Fig. 6.

Alya is an unstructured finite element code. The mesh partitioning therefore relies on the element graph, whose complexity depends on the geometry considered. Libraries such as ZOLTAN [28], SCOTCH [49], or METIS [174], which are based on graph partitioning algorithms, may be used to decompose an EBG mesh. Just like NUMA, Alya does not require halos and the information exchange between neighbors is carried out on the interface nodes, that is, the nodes shared by different subdomains. From the parallelization point of view, the load balance and the communication scheduling for these two codes depend on the quality of the partition.

Several iterative solvers are available, and the selection depends on the physical problem considered. The incompressible Navier–Stokes equations require the solution of the momentum equation and the pressure equation [140]. For the first algebraic system, the GMRES method with a simple diagonal preconditioning is efficient in most of the cases, and few iterations are required to obtain convergence. For the pressure equation, a deflated conjugate gradient method [203] is used together with linelet preconditioning [277], which is very efficient in the presence of boundary layers. The four scalabilities presented in Fig. 6 were obtained for the Navier–Stokes equations. The last one represents the combustion in a kiln, which consists in solving the low Mach Navier–Stokes equations together with a temperature equation and chemical reactions.

1.6 Plan of the Paper

The rest of the review is organized as follows: in Sect. 2 we give an overview of the different equation sets used in the dynamical cores of atmospheric models. Element-based Galerkin methods within the context of NWP are introduced in Sect. 3. Since EBG methods may produces unphysical extrema (especially high-order EBGs), stabilization/filtering is often required: a review of some stabilization methods follows in Sect. 4. Section 5 explores accurate grid generation within high resolution simulations (e.g. well resolved topography), along with static and dynamic grid adaptivity. A summary is reported in Sect. 6.

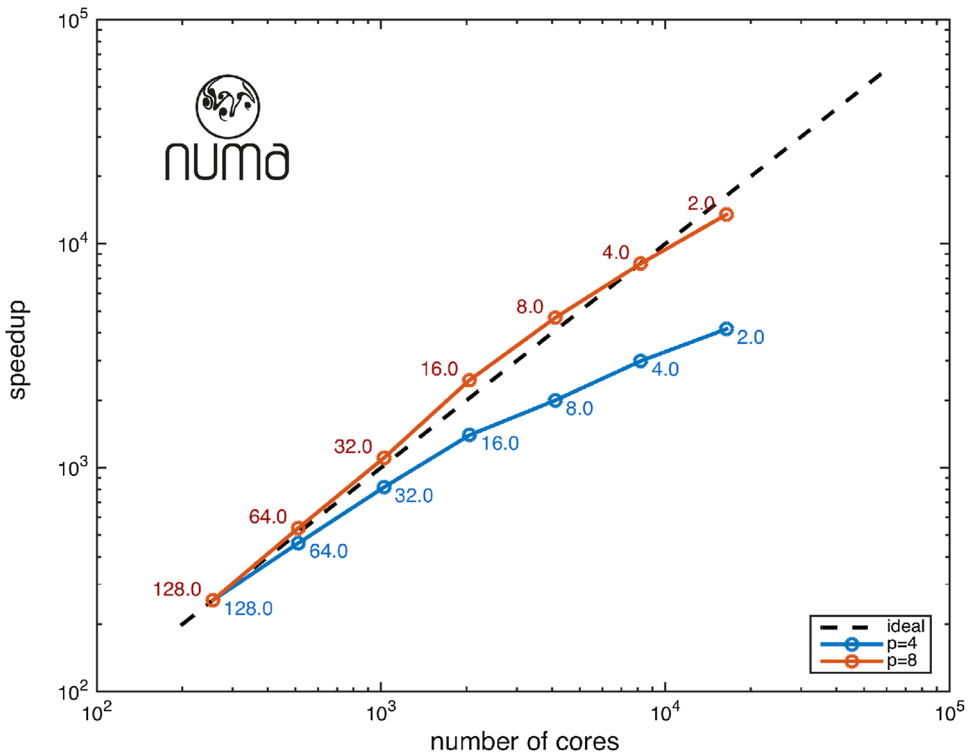

Fig. 5 Scalability study with NUMA using a 1D semi-implicit (HEVI) simulation of a 3D rising thermal bubble in a 1 km^3 cubed domain for polynomial degrees 4 and 8 (see legend), using 32^3 elements. The average number of elements per core is given by the numbers next to each data point. This scalability study was performed on the Blue Gene Vesta of the Argonne National Lab

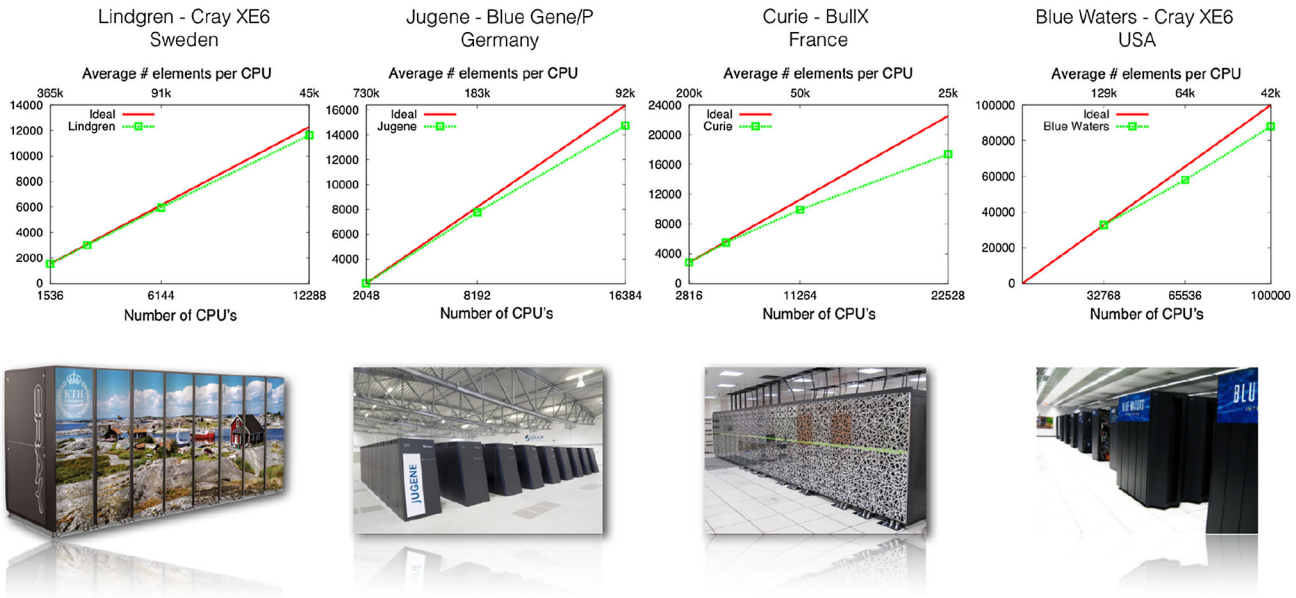

Fig. 6 Scalability study for a fully implicit simulation using Alya [142, 308] on four different HPC architectures

2 Equation Sets for Atmospheric Modeling

For typical atmospheric scales (1 m to 1000+ km), the earth's atmosphere can be treated as a continuum governed by the compressible Navier–Stokes equations with body forces to model the effects of gravity and the Earth's rotation (i.e. Coriolis force). Although the gravitational force varies with both altitude and latitude, these minor perturbations are generally neglected. In this section, we neglect the effects of moisture, solar radiation, and heat flux from the ground and consider the *dry* dynamics of the atmosphere. Let Ω be a three-dimensional domain in a rotating reference system $\mathbf{x}$ and let $t \geq 0$ be time. The state of dry, stratified air can be described by density, ρ, pressure, p, absolute temperature, T, and velocity field, $\mathbf{u}$,

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p - \nabla \cdot \boldsymbol{\sigma} = -2\rho(\boldsymbol{\omega} \times \mathbf{u}) - \rho \mathbf{g}, \tag{1a}$$

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

$$\frac{\partial E}{\partial t} + \nabla \cdot ((E + p)\mathbf{u}) - \nabla \cdot \left(\frac{\mu c_p}{\mathrm{Pr}} \nabla T + \mathbf{u} \cdot \boldsymbol{\sigma} \right) = 0, \tag{1c}$$

where $\boldsymbol{\omega}$ is the rotational velocity of the Earth, $\boldsymbol{\sigma}$ is the viscous stress tensor, $\mathbf{g}$ is the sum of true gravity and the centrifugal force, and the total energy, E, is given by

$$E = \rho c_v T + \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} + \rho g r, \tag{2}$$

where r is the radial distance from a fixed reference point at the center of the earth. Equation (2) consists of three components: internal energy, kinetic energy, and gravitational potential energy. For a Newtonian fluid with dynamic viscosity μ, the viscous stress tensor is given by

$$\boldsymbol{\sigma} = \mu \left[(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}}) - \frac{2}{3} (\nabla \cdot \mathbf{u}) \mathbf{I} \right], \tag{3}$$

where $2/3$ is a constant derived from the Stokes hypothesis and T is the vector transpose [232]. The system (1) of five conservation laws in six unknowns is closed by the equation of state (ideal gas law) for pressure:

$$p = \frac{R}{c_v} \left(E - \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} - \rho g r \right). \tag{4}$$

We note that Eq. (3) does not incorporate any effects of turbulent dissipation. Since the Kolmogorov length scale of a typical atmospheric problem is on the order of 0.1 mm, direct numerical simulation (DNS) of atmospheric motion is not possible with the current computational resources. To properly account for unresolved turbulent motion (e.g. turbulent dissipation), a sub grid scale model or turbulence closure scheme should be included. To simplify the treatment of the most commonly used sets of equations and of the numerical methods discussed below, we will neglect viscosity and restrict our analysis to the Euler equations ($\mu = 0$) and various approximations utilized in atmospheric modeling; however, we will revisit viscous effects in the context of stabilization in Sect. 4.

Atmospheric models can be broadly classified into three groups: (1) non-hydrostatic models based on the compressible Euler equations, (2) hydrostatic models, which assume a vertical momentum balance between gravity and

the vertical pressure gradient but include the vertical stratification of the atmosphere, and (3) sound-proof models. We also mention the shallow water model, which neglects all vertical motion by assuming each column of air moves as a rigid body, as shallow water models are often developed to test the horizontal propagation of features by numerical methods before they are applied to the solution of the equations for a full atmosphere.

The set of governing equations constitutes the *dynamical core* of the model. In the following sections, we broadly survey the equation sets commonly used in existing operational and research atmospheric models, beginning with non-hydrostatic models and ending with shallow water models. For a discussion of the interplay between the choice of equation set and numerical challenges encountered, consult [299]. For an analysis of the differences between non-hydrostatic, hydrostatic, shallow atmosphere (note, not to be confused with the shallow water model) and deep atmosphere approximations, see [317].

2.1 Non-hydrostatic Models

The fully compressible Euler and Navier–Stokes equations model all the scales and motions of the atmosphere. In NWP the equations expressed in the form of (1) are very often algebraically manipulated via the introduction of derived physical variables to help the physical interpretation of the atmosphere. For example, let us introduce potential temperature, θ, which is the temperature that an air parcel would have if it were expanded or compressed adiabatically to a standard pressure $p_0 = 1000\,\text{hPa}$ [138]. Potential temperature is related to p and T via the expression $\theta = T/\pi$, where

$$\pi = (p/p_0)^{R/c_p} \tag{5}$$

is a normalized pressure (known as Exner pressure) with respect to a reference pressure p_0. Given θ, the following conservation laws for $(\rho, \mathbf{u}, \theta)^{\mathrm{T}}$ are obtained by transforming Eq. (1c):

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = -\rho \mathbf{g} - 2\rho(\boldsymbol{\omega} \times \mathbf{u}), \tag{6a}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{6b}$$

$$\frac{\partial \rho \theta}{\partial t} + \nabla \cdot (\rho \theta \mathbf{u}) = 0. \tag{6c}$$

The equation of state for pressure (4)

$$p = p_0 \left(\frac{\rho R \theta}{p_0} \right)^{c_p/c_v} \tag{7}$$

completes the system. Numerical methods for the solution of this system can be easily constructed to conserve mass and momentum. It is, however, much more difficult to formulate numerical schemes that also conserve energy. However, for an adiabatic and reversible system, entropy is conserved. Entropy s may be related to potential temperature θ via the relation

$$s = c_p \ln \theta + \text{constant},$$

thereby justifying the use of θ rather than E.

The ARW-WRF model [271] is based on this set, and so are the finite volume model described in [2], the Met Office ENDGame [322], and the German LM model [62]. The Nonhydrostatic Unified Model of the Atmosphere (NUMA) [117, 176] developed at the Naval Postgraduate School is designed around two different sets, including (6). NUMA is the underlying dynamical core of the next generation NWP model of the U.S. Navy, NEPTUNE.

Constructing the divergence of flux in Eq. (6) requires some additional computational overhead; this overhead may be reduced by converting Eqs. (6) to their advection form:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho} \nabla p = -\mathbf{g} - 2\boldsymbol{\omega} \times \mathbf{u}, \tag{8a}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0, \tag{8b}$$

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = 0, \tag{8c}$$

again, completed by an equation of state given by Eq. (7). Numerical approximations to this set of equations can be constructed to conserve mass, although conservation of momentum and energy is more difficult to obtain. NUMA is designed to be able to handle this set as well, although the flux form (6) is the required formulation when NUMA is executed in the discontinuous Galerkin mode.

By combining the definition of the Exner pressure [Eq. (5)] with the continuity equation in Eqs. (8), we obtain:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + c_p \theta \nabla \pi = -\mathbf{g} - 2\boldsymbol{\omega} \times \mathbf{u}, \tag{9a}$$

$$\frac{\partial \pi}{\partial t} + \mathbf{u} \cdot \nabla \pi - \frac{R}{c_v} \pi \nabla \cdot \mathbf{u} = 0, \tag{9b}$$

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = 0, \tag{9c}$$

where $(\pi, \mathbf{u}, \theta)^{\mathscr{T}}$ is the vector of the solution variables [71, 85]. The practitioners who use this set justify it by saying that it is self-contained because there is no need for a state equation. For as much as it is evident that no equation of state is directly necessary, we still need to point out that the

algebraic computation of p from an equation of state similar to (4) is here simply substituted by the diagnosis of p and ρ from π or θ; an operation that is still necessary when it comes to the analysis of the forecast. This still contributes to the net operation count to Eq. (4). Equations (9) do not conserve mass, momentum, and energy; yet, they are widely used in operational NWP models such as MM5 developed at Penn State and NCAR [83], NMM based on the work by Janjic [166] at NCEP, COAMPS [135] from the U.S. Naval Research Laboratory (NRL), and HIRLAM [256, 257] by a consortium of European numerical weather services.

2.1.1 *Sound Waves: Anelastic Models and Implicit Time Integration*

All of the non-hydrostatic equation sets described in the previous section are compressible; therefore, they all contain sound waves which propagate at a very high speed (approximately 300 m/s) relative to meteorologically relevant phenomena. If these equations are discretized explicitly, a small time-step must be utilized in order to satisfy the stability criterion based on the Courant–Friedrichs–Lewy (CFL) condition [67], thereby increasing the computational cost of the model. Since the vertical grid-spacing is typically much smaller than the horizontal grid spacing, the vertically propagating sound waves are the most problematic aspect in these equation sets. To bypass the small time-step requirement of the models that support sound-waves, yet preserve the remaining dynamics, the *anelastic* model was introduced in 1953 by Batchelor [15] and later analyzed in [10, 202, 233], where the continuity equation in Eqs. (6) and (8) is replaced by

$$\nabla \cdot (\overline{\rho}(z)\mathbf{u}) = 0. \tag{10}$$

In (10), density ρ is only a function of height. An improved soundproof approximation is the pseudo-incompressible model proposed by Durran [84, 86], where the time dependence of density is accounted for, although density is a function of a time-invariant reference state pressure and time-dependent potential temperature. All these models are able to filter sound from the original compressible Euler/Navier–Stokes equations, but still account for the most important waves (e.g. Rossby) in the solution of the atmospheric motion. The interested reader may consult the review [179] for more on the validity of these approximations. A step towards the blending of soundproof and compressible Euler equations was recently investigated in [20].

The soundproof approximation of the governing equations is one option to the necessary filtering of sound waves. The fully compressible, non-hydrostatic equations can, on the other hand, be approximated in time via a semi-implicit scheme as done in [71, 291, 292]. Because the fast waves are treated implicitly in a semi-implicit approximation, the time step is only limited by the non-linear advective part of the equation; hence, the time-step is limited by the *advective* CFL condition $\Delta t \leq C\Delta x/||\mathbf{u}||$, which is far less stringent than the CFL condition $\Delta t \leq C\Delta x/(||\mathbf{u}|| + c_s)$, where C is a constant of order one and c_s is the speed of sound.

Semi-implicit methods are closely related to implicit–explicit (IMEX) methods [189]. Semi-implicit is, for the most cases, tied to the combination explicit leap-frog + implicit Crank–Nicholson, whereas IMEX can be viewed as a generalization that allows for different time-differencing schemes, as first proposed in 2009 by Restelli and Giraldo [249] to solve the fully compressible Navier–Stokes of nonhydrostatic stratified flows approximated in space by DG. Moreover, the IMEX + DG by Restelli and Giraldo is a general method applicable to different Mach regimes for viscous and inviscid flows. In 2004, Dolejši and Feistauer [76] coupled DG with an implicit–explicit time discretization scheme to solve the Euler equations of fully compressible flows. In that paper, the numerical flux term was first discretized in a fully implicit manner; then, the implicit numerical flux was linearized via a Taylor expansion, resulting in a linear system of equations which is solved via a sparse iterative solver, as opposed to a more expensive non-linear solver (e.g. Newton–Krylov) as required by a fully implicit discretization. More recent work on IMEX methods includes [87], which utilizes Adams and backward difference methods, and [314], which takes a horizontally-explicit vertically-implicit (HEVI) approach.

In 2013, an IMEX version of the Nonhydrostatic Unified Model of the Atmosphere NUMA was introduced in [117]. Both a 3D IMEX scheme which discretized the horizontal and vertical (linear) operators implicitly, and a 1D IMEX which only discretized the vertical operators implicitly (HEVI), were derived and compared using both second-order backward difference formulas and higher-order (up to order 4) implicit Runge–Kutta methods.

As mentioned earlier, 3D-IMEX methods require the solution of a linear system of equations. This linear solve may be poorly conditioned (especially for large Courant numbers) and hence computationally expensive. An alternative method that does not require a linear solve is the split-explicit method [293]. The split-explicit approach relies on sub-time stepping to treat the terms that represent sound and gravity waves within one larger explicit time-step for the remaining terms. This method is common in atmospheric simulations, in spite of its low accuracy [318, 319] and potential instabilities.

2.1.2 Nearly-Hydrostatic Flows

Dynamics in the atmosphere are characterized by small variations of thermodynamic quantities with respect to some background state [178, 207]:

$$\rho(\mathbf{x},t) = \rho'(\mathbf{x},t) + \bar{\rho}(z) \tag{11a}$$

$$p(\mathbf{x},t) = p'(\mathbf{x},t) + \bar{p}(z) \tag{11b}$$

$$\Theta(\mathbf{x},t) = \Theta'(\mathbf{x},t) + \bar{\Theta}(z) \tag{11c}$$

where the primed and barred quantities represent, respectively, the perturbation and the background state of ρ, p, and Θ. In Eq. (11c), $\Theta = \rho\theta$. In typical atmospheric simulations, $\rho' \ll \bar{\rho}$, $p' \ll \bar{p}$ and $\Theta' \ll \bar{\Theta}$. If the vertical acceleration is zero, the vertical component of the momentum equation reduces to the hydrostatic balance given by the following equation:

$$\frac{\partial p}{\partial z} = -g\rho. \tag{12}$$

Given these considerations and the analysis of nearly-hydrostatic flows for well-balanced methods [30], the system (6) is transformed in terms of perturbation variables where the Coriolis term is neglected. Substituting Eq. (11) into Eq. (6) and applying Eq. (12) to the z-component yields

$$\frac{\partial \rho\mathbf{u}}{\partial t} + \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u}) + \nabla p' = -\rho'\mathbf{g}, \tag{13a}$$

$$\frac{\partial \rho'}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0, \tag{13b}$$

$$\frac{\partial (\rho\theta)'}{\partial t} + \nabla\cdot(\rho\theta\mathbf{u}) = 0. \tag{13c}$$

Throughout this review, the primes will be mostly omitted to simplify notation.

2.2 Hydrostatic Versus Non-hydrostatic Models

Atmospheric models can be distinguished as hydrostatic and non-hydrostatic. If we assume the vertical acceleration to be negligible, the vertical momentum equation of the hydrostatic system reduces to the diagnostic equilibrium equation (12). At every time-step, this time-independent equation is solved instead of the full equation for vertical momentum. Sound waves are eliminated in the vertical direction [85] but not in the horizontal direction. Because the size of the domain in the horizontal direction is typically much larger than the vertical depth of the atmosphere and the grid size along x and y may be orders of magnitude larger than the grid spacing along z, a much larger time-step may be utilized.

The hydrostatic approximation has been a central approximation of NWP for the past four decades and is used in the hydrostatic primitive equations (HPE) discussed in the next section. This approximation is valid for horizontal grid spacing larger than 10 km [165, 299]. The hydrostatic approximation is still appropriate to simulate synoptic scale phenomena where the vertical acceleration can be neglected, but is no longer considered in any mesoscale simulation. With the availability of more powerful computers, the non-hydrostatic formulations described above are standard for mesoscale NWP. The reader should refer to [21, 29, 106, 118, 124, 135, 166, 167, 271, 323] for more on the evolution of non-hydrostatic models.

2.2.1 Hydrostatic Primitive Equations

The hydrostatic primitive equations (HPE) govern the dynamics in synoptic scale (e.g. global-scale) meteorology and are valid for horizontal resolution coarser than 10 km. The HPE are expressed in so-called σ coordinates which allow the boundary condition on the ground to be easily applied, even in the presence of complex orography. The HPE are derived by first expressing the compressible Euler equations in terms of pressure, velocity, and potential temperature. A hydrostatic balance is applied in the vertical direction, which removes vertical acceleration from the momentum equation. Since the HPE rarely appear outside of atmospheric and climate studies, we present a brief derivation from the compressible Euler equations. A more comprehensive treatment is found in [138].

We first apply a Coriolis term to the right hand side of the momentum equation [Eq. (8a)]. Decomposing the velocity $\mathbf{u}$ into a horizontal $\mathbf{u}_H$ and vertical w components, the horizontal momentum balance is given by

$$\frac{D\mathbf{u}_H}{Dt} = -\frac{1}{\rho}\nabla_H p - f\mathbf{k}\times\mathbf{u}_H \tag{14}$$

where $f = 2\omega\sin\alpha$ is the Coriolis constant at the latitude α for angular rotation ω. Next, we transform Eq. (14) into *isobaric* coordinates (x, y, p), where pressure is the vertical component; this is a useful intermediate step on the path to σ coordinates. Introducing a velocity potential $\Phi(x, y, p, t)$, it can be shown that $\nabla_P\Phi = \nabla p/\rho$, where the gradient is taken with respect to isobaric coordinates, yielding

$$\frac{D_P\mathbf{u}_H}{D_P t} = -\nabla_P\Phi - f\mathbf{k}\times\mathbf{u}_H, \tag{15}$$

where the total derivative in Eq. (15) is defined as

$$\frac{D_P}{D_P t} = \frac{\partial}{\partial t} + \mathbf{u}_H\cdot\nabla_H + \hat{\omega}\frac{\partial}{\partial p} \tag{16}$$

and $\hat{\omega} = Dp/Dt$. Next, we transform Eq. (15) into σ coordinates via $\sigma = p/p_s$, where $p_s = p_s(x, y, t)$ is the surface pressure. Note that in this coordinate system, the boundary

condition on the ground is always $\sigma = 1$. Application of the chain rule to the gradient of the velocity potential yields

$$\nabla_P \Phi = \nabla_\sigma \Phi - \sigma \nabla \ln p_s \frac{\partial \Phi}{\partial \sigma} \tag{17}$$

while the total derivative is given by

$$\frac{D_\sigma}{D_\sigma t} = \frac{\partial}{\partial t} + \mathbf{u}_H \cdot \nabla_H + \dot{\sigma} \frac{\partial}{\partial \sigma} \tag{18}$$

Combining Eqs. (17) and (18) in Eq. (15) yields

$$\frac{\partial \mathbf{u}_H}{\partial t} + \mathbf{u}_H \cdot \nabla_H \mathbf{u}_H + \dot{\sigma} \frac{\partial \mathbf{u}_H}{\partial \sigma} = \nabla_\sigma \Phi - \sigma \nabla \ln p_s \frac{\partial \Phi}{\partial \sigma} - f \mathbf{k} \times \mathbf{u}_H \tag{19}$$

In a similar manner, an equation of continuity for the surface pressure p_s is derived from (6a)

$$\frac{\partial p_s}{\partial t} + \nabla_H \cdot (p_s \mathbf{u}_H) + p_s \frac{\partial \sigma}{\partial \sigma} = 0 \tag{20}$$

along with a transport equation for potential temperature θ from (6c)

$$\frac{\partial \theta}{\partial t} + \nabla \cdot \mathbf{u}_H \cdot \nabla_H \theta + \dot{\sigma} \frac{\partial \theta}{\partial \sigma} = 0 \tag{21}$$

In each σ level, we solve for the *prognostic variables* $\mathbf{q} = (p_s, \mathbf{u}_H, \theta)^{\mathrm{T}}$, while the diagnostic variables are the vertical velocity $\dot{\sigma}$, pressure p, and geopotential ϕ. Because these equations are in exact hydrostatic balance, there are no vertically propagating acoustic or gravity waves. By computing the eigenvalues of the HPE, it is shown that the fastest moving waves are horizontally propagating gravity waves [114]. Hence, even with an explicit time integrator, a much larger time-step may be used with the HPE than with the compressible Euler equations. For this reason, the HPE form the basis of most global atmospheric and climate models.

2.3 Shallow Water Equations (SWE)

The hydrostatic primitive equations require a solution at N model levels (independent σ or pressure levels). This requires significant computational effort. The HPE may be simplified even further to remove all vertical dependence. One approach is to expand each prognostic variable in Eq. (20) in a 1D Fourier series with height σ as the argument and only retaining the zeroth-term in this series, commonly called the *barotropic mode*. Another approach is to start with the full compressible Euler equations and apply both the hydrostatic approximation given by Eq. (12) and the *shallow water approximation* where the deviation of the geopotential height Φ from a given reference geopotential Φ_0 is small. From an ocean modeling point of view, this assumption is equivalent to assuming the water depth is small compared to the wavelength of the waves of interest (gravity waves and Coriolis induced Rossby waves). In flux form, the SWE of a viscous atmosphere of depth h on a rotating sphere of radius r are:

$$\frac{\partial \Phi \mathbf{u}}{\partial t} + \nabla \cdot (\Phi \mathbf{u} \otimes \mathbf{u}) = -\Phi \nabla \Phi - f(\mathbf{x} \times \Phi \mathbf{u}) + \mu \mathbf{x} + \nu \nabla^2 (\Phi \mathbf{u}), \tag{22a}$$

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\Phi \mathbf{u}) = 0. \tag{22b}$$

Equation (22) may be expressed in Cartesian coordinates instead of spherical coordinates by applying a fictitious force $\mu \mathbf{x}$, where μ is the Lagrange multiplier; this approach, which facilitates an arbitrary spherical grid, was first proposed by Coté [63] for the semi-Lagrangian solution of the problem and later used in [110, 112] for the solution of the full nonlinear equations. The numerical solution of SWE on spherical geometries is reported by many authors such as [242] (FEM), [112, 158, 204, 294] (SEM), [116, 227, 303] (DG), [211] (unified CG/DG on different unstructured grids with static and dynamic adaptivity), [191, 196, 309, 316] (FV), [278] (comparison between SEM and FV), [245] (comparison using different numerical methods).

2.4 Transport in the Atmosphere

In a typical atmospheric model, there are multiple forms of water (e.g. vapor, rain, ice); in a climate model, there are also hundreds of chemical species. These quantities are transported and diffused by atmospheric dynamics and are classified as *tracers*. In turn, these tracers actively feedback to dynamics (e.g. latent heat release). To model these tracers, the governing equations of a dry flow must be coupled to a set of transport–diffusion equations for such tracers. For simplicity, we describe how tracers are treated in atmospheric models by looking at the transport of three water quantities only; however, this approach applies to an arbitrary number of tracers.

Let us define the mixing ratios of water vapor, cloud water, and rain as $q_v = \rho_v/\rho, q_c = \rho_c/\rho$ and $q_r = \rho_r/\rho$, where $\rho_{v,c,r}$ are the densities of water vapor, cloud, and rain. Let us also choose one of the nonhydrostatic equation sets described previously and write the coupled system of equations that model a moist atmosphere; we consider system (8) and write the following:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho} \nabla p = -\mathbf{g}(1 + \varepsilon q_v - q_c - q_r) - 2\boldsymbol{\omega} \times \mathbf{u} + S_{turb}, \tag{23a}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{23b}$$

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = \nabla \cdot (\kappa_\theta \nabla \theta) + S_\theta(\rho, \theta, q_v, q_c, q_r), \tag{23c}$$

$$\frac{\partial q_i}{\partial t} + \mathbf{u} \cdot \nabla q_i = \nabla \cdot \left(k_{q_i} \nabla q_i\right) + S_{q_i}(\rho, \theta, q_v, q_c, q_r), \quad \text{for } i = v, c, r, \tag{23d}$$

where $\varepsilon = R/R_v$ is the ratio of the gas constants of dry air, R, and of water vapor, R_v. Moist air contributes to the buoyancy of the flow, so that the right hand side of the momentum equation must be corrected by the total buoyancy $\mathbf{B} = -g\mathbf{g}(1 + \varepsilon q_v - q_c - q_r)$. The diffusion coefficients k_{θ_i} and k_{q_i} are typically modeled via an algebraic turbulence closure via

$$k_\theta = \nu/\mathrm{Pr}_0 + \nu_t/\mathrm{Pr}_t \tag{24a}$$

$$k_{q_i} = \nu/Sc_0 + \nu_t/Sc_t \tag{24b}$$

while the closure term S_{turb} depends on the turbulence model employed. In Eq. (24), ν is molecular viscosity, ν_t is eddy viscosity, Sc_0 is the molecular Schmidt number and Sc_t is the turbulent Schmidt number. Typical values are $Sc_0 = Sc_t = 0.7$. The microphysical processes that involve phase change in the water content are modeled by the source/sink terms, S_{θ,q_i}, in the equations. For example, in the case of water vapor, S_i is driven by evaporation and condensation. These terms can be modeled and computed by some properly designed microphysics scheme, such as the Kessler [177] scheme for warm clouds (no ice involved).

The appropriate numerical discretization of Eq. (23d) is still an active topic of research, especially since moisture possesses large gradients that can cause instabilities. In addition, since the mass fractions q_i are *a priori* non-negative, the numerical discretization should be monotonic or, at the very least, positivity-preserving. If, for example, our system produced negative moisture, the physical parameterization would have to resolve this issue in some way (e.g. clipping the negative values); in addition, the resulting incorrect feedback may pollute the overall solution and cause *artificial rain* to be produced by the model. The words of John P. Boyd are an amusing conclusion to this paragraph: "[...] Clever adaptive algorithms that work for smooth, straight shocks disintegrate into computational anarchy when flayed by gravity waves, assaulted by moist convective instability, battered by highly temperature-sensitive photochemistry, and coupled to the vastly different time and space scales of the ocean[...]" (SIAM News, Multiscale Numerical Algorithms for Weather Forecasting and Climate Modeling: Challenges and Controversies. Nov 2008, Vol. 41 issue 9). Monotonic solutions are certainly more difficult to achieve with high order numerical methods. The problem is particularly challenging when the transport equation is solved by high order methods such as spectral elements or DG. High order methods produce Gibbs oscillations near sharp gradients; these oscillations are unphysical and are exacerbated by increasing the order. Hence, limiters [187] or adaptive filtering is necessary. We will address this problem in Sect. 4, along with some issues involved with unstable Galerkin solutions.

2.4.1 Cloud Microphysics: Kessler Parameterization

Cloud microphysics include all thermo-physical processes at the scales of the particles that form the cloud. Examples are the phase change of water quantities or the agglomeration of particles into larger ones. Most physical processes typical of storm dynamics (e.g., precipitation, freezing, deposition, or sublimation) have physics across a large range of spatial and temporal scales that makes direct numerical simulation unfeasible (see [88], Ch. 10). For this reason, parameterization is commonly used within numerical models. Microphysical parameterization relies on the physical knowledge of certain processes without the need for fully resolving all the microscale processes that are involved. The clear limitation is that certain phenomena cannot be represented with high accuracy if they lie outside of the conditions required by the parameterization. Differentiation A simple representation of cloud microphysics was designed by Kessler and reported in his monograph [177].

Kessler's is a *bulk* model, meaning that water species are categorized only with respect to the particles' type. In other words, if we speak about cloud water, we would model it through one equation that represents the transport of cloud water concentration with water droplets of one single size. Bulk models are contrasted by *explicit* models, where, within each category (e.g., cloud, rain) the size of the water particles is considered as well. Explicit models are certainly more physically accurate, but they are more costly due to the greater number of quantities that must be accounted for. For more information on the topic refer to Houze's book [139] and to more recent literature (e.g. [39]).

Kessler's is a simple scheme based on the main assumption that ice is not contemplated (warm rain). The main limitation of the warm condition is that only moist convection at the tropics or at mid-latitudes in the warm season can be represented. The three forms of water that are considered are: (1) water vapor; (2) cloud water (liquid water whose size is so small that its terminal fall speed is negligible); and (3) precipitating water that only includes rain (namely, drops whose diameter is >0.5 mm). Drizzle is excluded (rain of drop diameter between 0.2 and 0.5 mm).

The main processes resolved by a warm cloud microphysics scheme are briefly described below. These processes dictate how the source terms of the previous equations are defined and how they affect the dynamics of the simulation. The reader is referred to, e.g., [139] and references therein for a more thorough analysis.

Given the approximated Teten's formula [27] for the saturation vapor pressure,

$$e^* = 611.2 \exp\left(\frac{17.67T}{T+243.5}\right),$$

the saturation mixing ratio is given by

$$q_{vs} = \frac{\varepsilon e^*}{p - e^*}. \tag{25}$$

From [181], the source terms in (23) are

$$S_\theta = -\frac{L_v}{c_p T}(\dot{q}_{vs} + E_r), \tag{26a}$$

$$S_{q_v} = \dot{q}_{vs} + E_r, \tag{26b}$$

$$S_{q_c} = -\dot{q}_{vs} - A_r - C_r, \tag{26c}$$

$$S_{q_r} = \frac{1}{\rho}\frac{\partial}{\partial z}(\rho V_r q_r) - E_r + A_r + C_r, \tag{26d}$$

where c_{pl} and c_{pv} are the heat coefficients at constant pressure of liquid water and water vapor, respectively, $L_v = L_{v0} - (c_{pl} - c_{pv})(T - T_0)$ is the latent heat of vaporization with reference value $L_{v0} = 2.5e + 6\,J\,kg^{-1}$, T_0 is a reference temperature, V_r is the terminal fall speed of raindrops (taken positive in the downward direction), and $\dot{q_{vs}}$ is the rate of condensation or evaporation (the dot symbol indicates differentiation with respect to time). A_r, C_r, and E_r are the rates of autoconversion, collection, and evaporation of rain. They are computed using the formulas:

$$A_r = \texttt{MAX}(0, k_1(q_c - a_T)), \tag{27a}$$

$$C_r = k_2 \rho^{0.375}\, q_c\, q_r^{0.875}, \tag{27b}$$

$$E_r = -\frac{1}{\rho}\frac{(q_v/q_{vs} - 1)k(\rho\, q_r)^{0.525}}{5.4 \times 10^5 + 2.55 \times 10^6 (p\, q_{vs})}, \tag{27c}$$

where $k_1 = 0.001\,s^{-1}$, $k_2 = 2.2\,s^{-1}$, $a_T = 0.001\,kg^{-1}$ are Kessler's parameters and k is the ventilation factor that is a function of the terminal fall speed. Equation (27a) was derived by Kessler considering that a cloud is converted into rainwater whenever q_c exceeds a threshold a_T. *Autoconversion* is the rate at which the rain water content increases at the expense of cloud water due to the coalescence of smaller particles. Yet, this process is not fully understood. Nor is it fully understood how *collection* occurs. As the name suggests, collection can be explained as cloud water particles being collected by the falling larger rainwater droplets that go through the cloud layers during their fall. *Evaporation* occurs when the sensible heat flux from the environment into the water droplet is balanced by the latent heat of evaporation of the water particle. As in [276], the cloud droplets move at the same speed of the flow because they are considered having negligible terminal velocity.

The values of the constants in (27) are, to a certain extent, arbitrary [139]; however, by the observations, it is of common agreement that k_1, k_2 and a_T are non-linear terms with respect to q_c itself. They are also a function of temperature and of the distribution of the condensation nuclei. As it is pointed out in Emanuel's book [88], the lack of understanding of the underlying physics is such that different results are being obtained by different and more sophisticated schemes. However, this topic is beyond the scope of the present article. Nevertheless, it is important to emphasize that microphysical parameterization has a major role in forecasting clouds and precipitation, but is still an active field of investigation (see the 2008 paper by Morrison and Grabowski [223]).

2.4.2 Method of Solution Via Saturation Adjustment

Regardless of the type of space approximation, phase changes are classically treated via the *saturation adjustment technique* explained in detail in the appendix of [276]. Saturation adjustment—or fractional step method—is not the only option; however, due to its simplicity, it is convenient to describe it here to give a sense of how phase change is accounted for in these models.

The *saturation adjustment technique* consists of solving the problem in two steps. First, the prognostic equations are solved by neglecting all the terms that involve phase changes (all the S-terms are set to zero). This means that the dynamics and transport equations are advanced forward to an intermediate time-step n^* so that the intermediate values of the prognostic variables, $(\rho, p, \theta_\rho, q_{vs}, q_v, q_c, q_r)^*$, are obtained. These values are plugged into the `Kessler` module to compute the S-quantities defined above. Once the computation of S has completed, thermodynamic variables are updated and returned to the *Euler/transport* solver as the initial values for the next time step $n + 1$.

3 Element-Based Galerkin Methods: Finite Elements, Spectral Elements, and Nodal Discontinuous Galerkin

As discussed in the EBG roadmap, the finite element (FEM), spectral element (SEM), and discontinuous Galerkin (DG) methods, are specific types of Galerkin

approximation techniques. In this section, we introduce the ideas behind Galerkin schemes in general and then distinguish between FEM, SEM, and DG in particular. We then trace the history of EBG methods in NWP and climate modeling over the past 20 years.

3.1 Element-Based Continuous Galerkin Methods

The birth of Galerkin methods dates back to Boris Grigoryevic Galerkin and his work on the numerical solution of the equations of the elastic equilibrium of rods and plates [105], and to the original ideas of Walter Ritz [253] 6 years earlier. Popularized by Courant in the early 1940s for the study of vibration and equilibrium [66] and extensively developed only in the late 1950s and 1960s by structural dynamicists in the aircraft industry [4], finite element methods in particular are among the most common numerical methods based on a Galerkin approach and that are used today in a wide range of applications. They are used in industry and for research purposes in, e.g., structural analysis [324], fluid dynamics [333], and electromagnetism [14]. Galerkin based methods are a robust tool for the solution of any differential problem [79] and are accepted by scientists and engineers in theoretical studies and applications for a series of reasons such as the ease in modeling complex geometries, the flexible and general purpose programming format that they imply, and the intrinsic treatment of differential-type boundary conditions. In the following, we will describe the idea behind the method of weighted residuals, of which the Galerkin finite element, spectral element, and discontinuous Galerkin methods represent special cases. For a simple but quasi-rigorous analysis of the method we use a problem of real engineering interest and that is a fundamental problem in numerical weather prediction: the advection–diffusion equation. The reader is referred to the books by Fletcher [95] or by Karniadakis and Sherwin [173] as a reference for the more mathematical aspects of Galerkin methods, and to the lecture notes by Giraldo [115] for a unified treatment of high-order continuous and discontinuous Galerkin methods.

Let us take a general differential problem

$$\mathscr{L}(q) = S, \tag{28}$$

where $\mathscr{L}$ is the combination of both linear and non-linear differential operators in space $\mathbf{x}$ and time t, and S is a source function. Let d indicate the space dimension and let $\Omega \subset \mathbb{R}^d$ be the domain with the boundary $\partial\Omega$ where (28) is defined within the time interval $(0, t_f)$, and $t_f \in \mathbb{R}^+$. For the problem to be well-posed, suitable boundary and initial conditions must be added to (28). Unless otherwise stated, given a known function g, Dirichlet boundary conditions $q(\mathbf{x}) = g$ for $\mathbf{x} \in \partial\Omega$ will be applied to the problems described throughout this section.

As previously stated, Galerkin methods are a particular case of the method of weighted residuals. The idea behind this method is the numerical representation of the solution variable q by a finite dimensional approximation q^h obtained by the expansion

$$q^h(\mathbf{x}) = \sum_{k=0}^{N} \psi_k(\mathbf{x})\, \hat{q}_k, \tag{29}$$

where N is the number of nodes p_k of a possible partition of the domain Ω. On its discrete counterpart, Ω^h, a set of $k = 0, \ldots, N$ known analytic test functions ψ_k are defined (The two terms *test* and *basis* will be used interchangeably. The unknown coefficients $\hat{q}_k$ correspond to the physical values of q at node p_k. The finite difference method is conceptually different in that what is approximated in the differential problem are the differential operators and not the solution variable. Substitution of (29) into (28) is such that $\mathscr{L}(q^h) - S \neq 0$. The method is called *method of weighted residuals* because a linear system of algebraic equations in the unknowns $\hat{q}$ is built by imposing that

$$\int_\Omega w\, R\, d\Omega = 0, \tag{30}$$

where $R = \mathscr{L} - S$ is the (non-zero) residual of (28) and w is the weight function that has certain properties. Different methods arise from the selection of different w. The Bubnov–Galerkin method is found when $w = \psi_k$. We can then write the following:

$$\int_\Omega \psi\, [\mathscr{L} - S]\, d\Omega = 0. \tag{31}$$

This is the weak form of the original equation to be solved.

Remark 3.1 So far, no distinction between the finite and spectral element methods has been made. The difference stems from the definition of the interpolation points used to construct ψ_k.

3.1.1 Suitable Function Spaces

The choice of basis and test functions depend on the operator $\mathscr{L}$ under consideration. In the specific cases of the advection–diffusion equation and the Navier–Stokes equations of compressible flows, the highest order of the derivatives is 2, and the choice of the basis functions and the space to which they belong must depend on this regularity condition.

We show that the weak solutions to a linear elliptic operator must resides in the Sobolev space $H^1(\Omega)$. Consider an operator $\mathscr{L}$ defined on a global domain Ω with

boundary Γ acting on a state vector q; specifically, consider the elliptic operator

$$\mathscr{L}(q) = \nabla \cdot (\nu \nabla q) \tag{32}$$

where $\nu > 0$ is the kinematic viscosity. We consider the boundary-value problem

$$\mathscr{L}(q) = 0 \tag{33}$$

with a Dirichlet boundary condition $q(\mathbf{x}) = q_0(\mathbf{x})$ for all $\mathbf{x} \in \Gamma$ and $q_0(\mathbf{x}) \in C^1(\Gamma)$. Consider a test function $\psi \in L^2(\Omega)$ and also assume $q \in L^2(\Omega)$. The following calculation demonstrates that $q, \psi \in H^1 \subset L^2$.

Integrating Eq. (32) by parts yields

$$\int_\Omega \nu \nabla \psi \cdot \nabla q \, d\Omega = \int_\Gamma \psi \nabla \cdot (\nu \nabla q) \, d\Gamma. \tag{34}$$

Since $q_0 \in C^1$ and $\psi \in L^2$, the right hand side of Eq. (34) is bounded, implying that the left hand side is bounded as well. We then write that:

$$\begin{aligned} \left| \int_\Omega \nu \nabla \psi \cdot \nabla q \, d\Omega \right| &\leq \int_\Omega |\nu \nabla \psi \cdot \nabla q| \, d\Omega \\ &\leq \nu \|\nabla \psi\|_{L^2} \|\nabla q\|_{L^2} \end{aligned} \tag{35}$$

where the second line follows from the Cauchy–Schwartz inequality. Hence, both the norms $\|\nabla \psi\|_{L^2}$ and $\|\nabla q\|_{L^2}$ are bounded, implying that $\nabla \psi$ and ∇q are square-integrable over the global domain Ω. In other words, $\psi, q \in H^1(\Omega)$.

With regards to CG methods, this elementary calculation illustrates two key points:

1. The space of test functions must be a subset of H^1.
2. Since $H^1 \subset C^0$, the state vector q is necessarily continuous.

We hence define the space W of test functions ψ and trial solutions q as a subset of H^1 such that

$$W \doteq \{\psi, q \in H^1(\Omega) \quad \text{s.t.} \quad \psi = 0 \quad \text{and} \quad q = g \text{ on } \partial\Omega\}. \tag{36}$$

3.2 Finite and Spectral Elements: Discretization and Basis Functions

To discretize the problem in a finite and spectral element sense, the domain tΩ is first decomposed into a finite element partition $\mathscr{P}^h = \{K^i\}_{i=1,\ldots,n_{el}}$ of n_{el} conforming elements K^i such that

$$\Omega = \bigcup_{i=1}^{n_{el}} K^i, \quad \text{and} \quad \bigcap_{i=1}^{n_{el}} K^i = 0, \tag{37}$$

where every element K^i is the image of the reference element $\boldsymbol{I}$ by a non-singular bijective mapping $\mathbf{x} = \mathscr{H}^i(\boldsymbol{\xi})$ from physical space $\mathbf{x}$ to computational space $\boldsymbol{\xi}$. $\mathbf{J} = d\mathbf{x}/d\boldsymbol{\xi}$ is the transformation Jacobian matrix. A two-dimensional example of the map is represented in Fig. 7.

The need for mapping is purely practical and forms the foundations of the finite element computation. For details see [157].

3.2.1 Basis Functions: Finite Elements

Lagrange basis functions are a common choice in finite elements since they interpolate a continuous function exactly at the nodes x_l. These functions, defined by h_k from now on, have the property of being piecewise continuous and are such that

$$h_k(x_l) = \delta_{kl} \quad k, l = 0, \ldots, N,$$

where δ_{kl} is the Kronecker delta.

For linear, quadratic, and cubic finite elements, the roots of the basis function along the reference element $\boldsymbol{I}$ are the N + 1 equi-spaced nodes within the element. Using the definition of the Lagrange polynomials

$$h_k(\xi) = \prod_{l=0, l\neq k}^{N} \frac{\xi - \xi_l}{\xi_k - \xi_l}, \tag{38}$$

in Fig. 8 we plot h_k along a reference element up to 2nd-order. A 4th-order finite element and corresponding basis function are plotted in Fig. 9 (left).

3.2.2 Basis Functions: Spectral Elements

Unlike the case of high-order finite elements, the polynomials used with spectral elements are associated with zeros that are not equi-spaced. A convenient set is represented by the Legendre–Gauss–Lobatto (LGL) points. LGL nodes ξ_i are the roots of

$$(1 - \xi^2) P'_N(\xi) = 0, \tag{39}$$

where $P_N(\xi)$ are the Nth-order Legendre polynomial whose construction by recursive formulas can be found in [173].

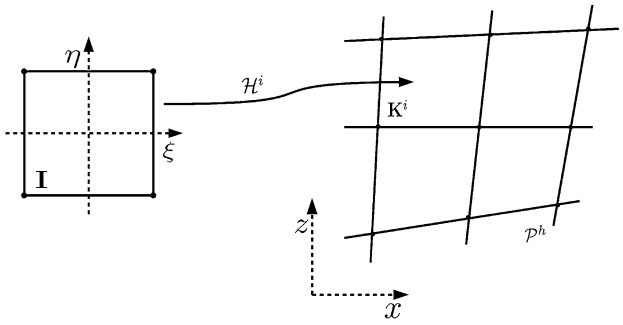

Fig. 7 Mapping from reference, (ξ, η), to physical space, (x, z). $\forall K^i \in \mathscr{P}^h : \quad K^i = \mathscr{H}^i(I)$

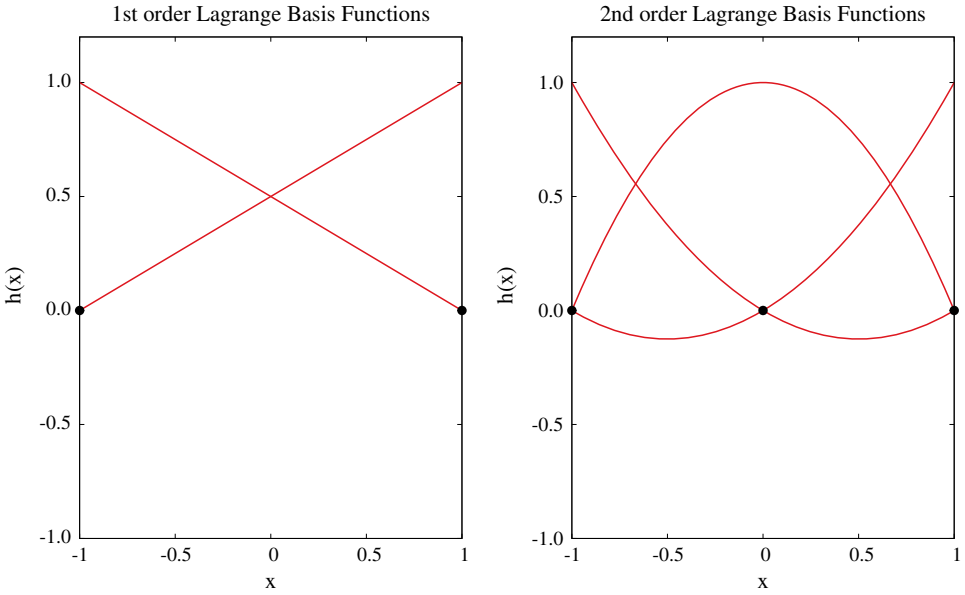

Fig. 8 Lagrange polynomials of order 1 (*left*) and 2 (*right*) along the 1D reference element $\boldsymbol{I} = [-1, 1]$. Clearly, they are equivalent for FE and SE

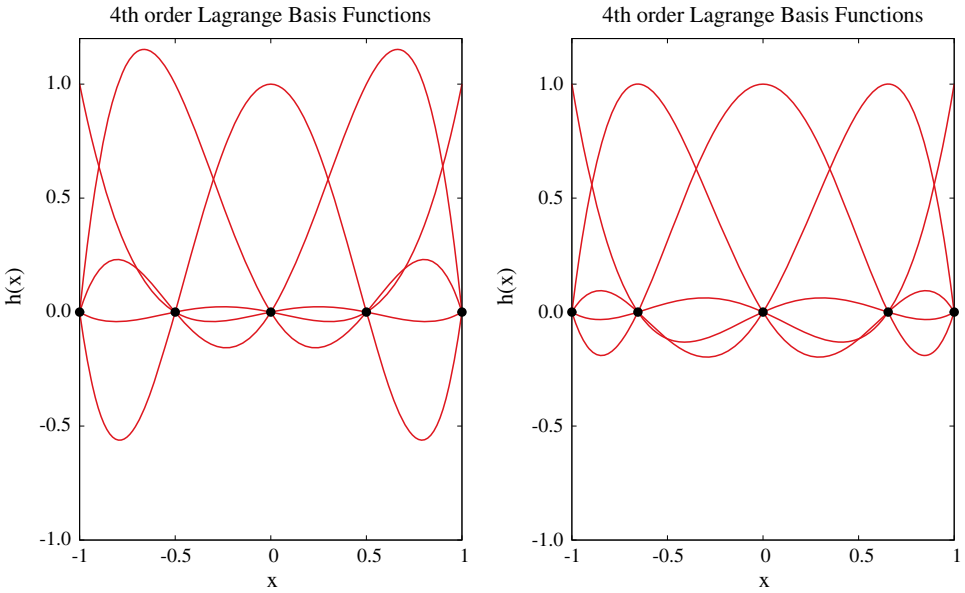

Fig. 9 Basis functions of order 4 along the 1D reference element $\boldsymbol{I} = [-1, 1]$. *Left* the nodes within the element are equi-spaced as for classical high-order FE. *Right* Lagrange–Legendre polynomials of order 4 whose roots are the non-equi-spaced Legendre–Gauss–Lobatto (LGL) quadrature points. Nodal SE and DG may employ LGL or LG quadrature. However, to obtain a diagonal mass matrix then LGL is the only choice for SE, while LG can still be used for DG

The polynomials that are used have the same δ-property of the Lagrange polynomials defined above. Their analytic expression is given by

$$h_k(\xi) = \frac{(\xi^2 - 1)P'_N(\xi)}{N(N+1)(\xi - \xi_k)P_N(\xi)}, \quad k = 0, \ldots, N, \qquad (40)$$

where P' indicates differentiation with respect to $\mathbf{x}$.[2] The 4th-order k-polynomials along $\boldsymbol{I} = [-1, 1]$ are plotted on the right panel of Fig. 9. The comparative plot (finite element on the left and spectral element on the right) is used to show that, if high-order is required, equi-spaced nodes produce unsatisfactory types of basis functions in the proximity of the edge points of the element. In other words, we lose control on the maximum and minimum values of h_k at the extrema of the element. When this happens, interpolation of any function is likely to suffer from such a condition. To show how this feature translates into the interpolation of a known analytic function, we use the following example from [115]. We define the *Witch of Agnesi* of unitary height as

$$z(x) = \frac{1}{1.0 + 50\,x^2},$$

[2] Alternatively, the basis functions can be constructed using Eq. (38).

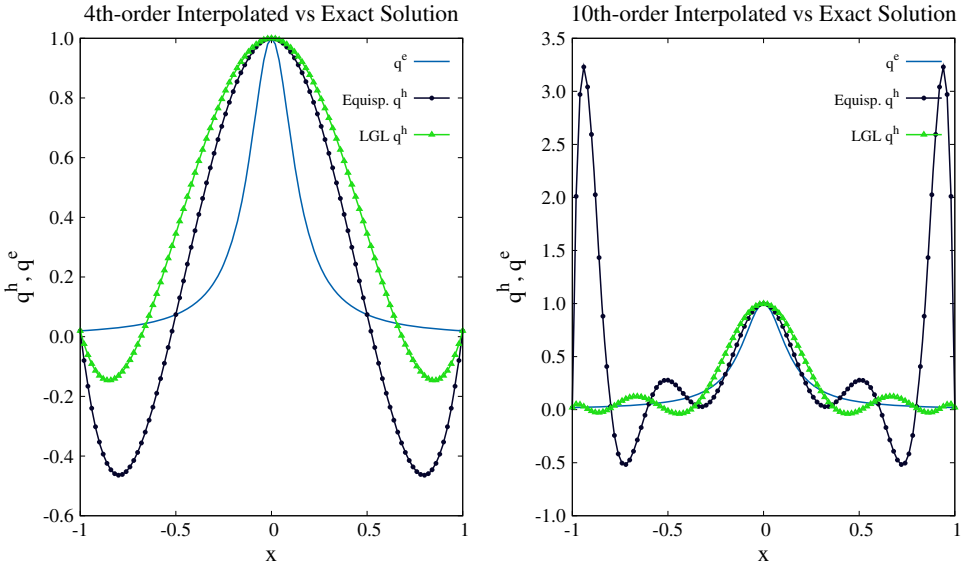

Fig. 10 Interpolation of a known function (Witch of Agnesi) using high-order interpolating functions with equi-spaced and LGL points. *Left* 4th-order interpolation. *Right* 10th-order interpolation

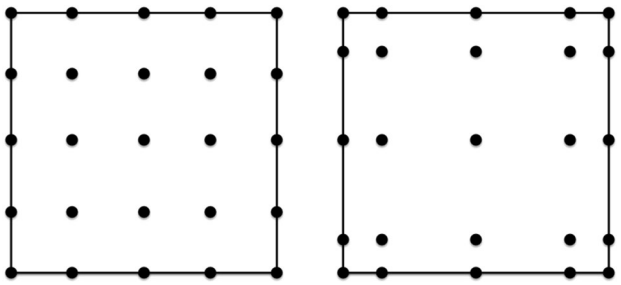

Fig. 11 Nodes disposition for a two-dimensional 4th-order finite element (*left*), and spectral element (*right*)

where $z(x)$ is smooth and continuous, and interpolate it using the basis functions $\psi(x) = h_k(x)$ defined above. The test is performed by 4th-order interpolation. Equi-spaced and non equi-spaced points are used along the unitary domain. Figure 10 shows that the more the polynomial order is increased, the better the result is when LGL nodes are employed. This is tied to the definition of the Lagrange polynomials and their interpolation strength given by the Lebesgue constant. The reader is referred to [115] for more details on this issue. Roughly speaking, this analysis serves as a practical way of showing one reason for the use of LGL points in high-order simulations rather than high-order elements with evenly distributed nodes. Figure 11 is a schematic representation of two 4th-order elements in two dimensions.

3.3 Discontinuous Galerkin

The discontinuous Galerkin method allows the numerical solution and, therefore, the basis functions to be discontinuous at the interface between neighboring elements. For this reason, the basis function is no longer required to live in H^1 but, rather, in L^2. Assume n_i is the number of elements that share grid point i. Then we will have n_i different values of the solution at that grid point; one coming from the computation on the left element and one on the right element, where left and right are defined with respect to the shared edge (or face in 3D). The basis functions for element Ω_e vanish everywhere outside the element. Hence Eq. (31) becomes a set of n_{el} independent equations for each element Ω_e:

$$\int_{\Omega_e} \psi \left[\mathscr{L}(q) - S(q)\right] d\Omega_e = 0. \tag{41}$$

The equations for the different elements are coupled by means of the fluxes between neighboring elements. For this purpose we write our equations in flux form:

$$\mathscr{L}(q) = \frac{\partial q}{\partial t} + \nabla \cdot \mathbf{F}(q) \tag{42}$$

where $\mathbf{F}$ is the flux tensor. Equation (41) becomes

$$\int_{\Omega_e} \left(\frac{\partial q}{\partial t} + \nabla \cdot \mathbf{F} - S\right) \psi(\mathbf{x})\, \mathrm{d}\Omega_\mathrm{e} = 0. \tag{43}$$

with $\mathbf{F} = \mathbf{F}(q)$ and $S = S(q)$. Integration by parts leads to

$$\int_{\Omega_e} \left(\frac{\partial q}{\partial t} - \mathbf{F} \cdot \nabla - S\right) \psi(\mathbf{x}) \mathrm{d}\Omega_\mathrm{e} = -\int_{\Gamma_\mathrm{e}} \psi(\mathbf{x}) \mathbf{n} \cdot \mathbf{F}\, \mathrm{d}\Gamma_\mathrm{e}, \tag{44}$$

where Γ_e is the boundary of Ω_e and $\mathbf{n}$ is the outward pointing unit normal vector on Γ_e. To compute the numerical solution, q^h, we replace the flux in the boundary integral by a so called numerical flux $\mathbf{F}^{h*}$:

$$\int_{\Omega_e}\left(\frac{\partial q^h}{\partial t}-\mathbf{F}^h\cdot\nabla-S^h\right)\psi(\mathbf{x})\mathrm{d}\Omega_\mathrm{e}=-\int_{\Gamma_\mathrm{e}}\psi(\mathbf{x})\mathbf{n}\cdot\mathbf{F}^{\mathrm{h}*}\mathrm{d}\Gamma_\mathrm{e}, \tag{45}$$

with $\mathbf{F}^h=\mathbf{F}\left(q^h\right)$ and $S^h=S\left(q^h\right)$. The numerical flux $\mathbf{F}^{h*}$ describes the flux through the discontinuous interface between neighboring elements in the same way of the finite volume (FV) method; therefore, we can choose any of the fluxes that are used with FV. For an introduction to different choices of fluxes we refer to [302]. Unlike FV, DG is relatively insensitive to the choice of the numerical flux due to the high order ($p\geq 3$) basis functions that are used within the element. Therefore a common choice for the numerical flux is the simple Rusanov flux:

$$\mathbf{F}^{h*}=\frac{1}{2}\left[\mathbf{F}\left(q_L^h\right)+\mathbf{F}\left(q_R^h\right)-\lambda\,\mathbf{n}\left(q_R^h-q_L^h\right)\right], \tag{46}$$

where $\lambda=||\boldsymbol{u}||_2+c$ is the maximum wave speed, $||\boldsymbol{u}||_2=\sqrt{u^2+v^2+w^2}$, and c is the speed of sound. The subscript L denotes the index of the element Ω_e whereas the subscript R denotes the index of the neighboring element. There are recent approaches to incorporate fluxes that are not perpendicular to the interface between elements [327].

The integration by parts of Eq. (45) leads to

$$\int_{\Omega_e}\left(\frac{\partial q^h}{\partial t}+\nabla\cdot\mathbf{F}^h-S^h\right)\psi(\mathbf{x})\mathrm{d}\Omega_\mathrm{e}=\int_{\Gamma_e}\psi(\mathbf{x})\mathbf{n}\cdot\left(\mathbf{F}^h-\mathbf{F}^{h*}\right)\mathrm{d}\Gamma_\mathrm{e}. \tag{47}$$

Using the expansion (29) of the numerical solution gives us

$$\frac{\partial\hat{q}_k}{\partial t}=-\int_{\Omega_e}\hat{\psi}_k(\mathbf{x})\left(\nabla\cdot\mathbf{F}^h-S^h\right)\mathrm{d}\Omega_\mathrm{e}+\int_{\Gamma_e}\hat{\psi}_k(\mathbf{x})\mathbf{n}\cdot\left(\mathbf{F}^h-\mathbf{F}^{h*}\right)\mathrm{d}\Gamma_\mathrm{e}, \tag{48}$$

with the definition $\hat{\psi}_i(\mathbf{x})=\sum_{k=1}^{M^h}M_{ik}^{-1}\psi_k(\mathbf{x})$ where $M_{ik}=\int_{\Omega_e}\psi_i(\mathbf{x})\psi_k(\mathbf{x})\mathrm{d}\Omega_\mathrm{e}$ are the components of the mass matrix M.

Second order derivatives in the differential operator $\mathscr{L}$ can be discretized with DG by transforming the problem into a coupled set of equations containing only first order derivatives as done in [54]. This approach is called the local DG (LDG) method. Other choices for discretizing second order operators are given in [266].

3.4 EBG Methods in Atmospheric and Climate Modeling

3.4.1 Continuous Galerkin

The use of continuous Galerkin methods in atmospheric simulations began five decades ago with the work on finite elements by Holmstrom [137] and Simons [269] in the 1960s. This continued in the 1970s (e.g. [69, 70, 101]) and was followed by an extensive production of articles in the 1980s and 1990s with, e.g., Staniforth [279], Beland et al. [19], or Burridge et al. [41], who set the foundations of the operational *Global Environmental Multiscale* (GEM) model [65, 326] of the Canadian Meteorological Center & Meteorological Research Branch (CMC-MRB). In the UK, Untch and Hortal [305] used finite elements for the vertical discretization of a semi-Lagrangian transport scheme and introduced it in the operational version of the European Centre for Medium-Range Weather Forecasts (ECMWF) global spectral model (IFS), with great improvement with respect to the FD version of the code. In the domain of Geophysical Fluid Dynamics, more Galerkin-type models have appeared since the beginning of the new millennium. In, e.g., [34, 193, 228] or [119], different variational formulations mostly based on spectral elements are employed to solve the shallow water equation or the Navier–Stokes and Euler equations for non-hydrostatic atmospheres. More examples of element-based models are the SE-Core [296], CAM-SEM by [74], the SE/DG *Nonhydrostatic Unified Model for the Atmosphere* (NUMA) [117, 176], the SEM *Community Earth System Model* (CESM) [73], the finite element multi physics model ALYA in atmospheric-mode [212, 213].

Possibly, the spectral element method is the most common EBG method used today to develop the next generation research NWP models. Spectral elements were first introduced in geophysical fluid dynamics by Ma [204]. Ma built on the pioneering work of Patera [235], who developed the spectral element method for incompressible CFD and developed an ocean model based on the shallow water equations. In particular Ma stressed the ability of SEM to (1) accurately simulate flows with high Rossby numbers and (2) simulate phenomena with long time durations due to SEM's low dissipation and dispersion error. Although Ma's primary application was coastal ocean modeling, he was explicitly aware of the intimate connection between ocean models and atmospheric/climate models and predicted that his work would serve as a basis for atmosphere and climate studies. Iskandarani [158] built on Ma's ocean model, showing that the accuracy of spectral elements successfully suppressed spurious pressure modes in ocean flows. Both Ma's and Iskandarani's work utilized Cartesian grids suitable for oceanographic problems. Two years later, Taylor et al. [294] developed the first SWE spectral element model using spherical geometry. In particular, this work used a cubed sphere with quadrilateral elements which built on the geometrical flexibility of spectral elements. The cubed sphere grid circumvented the well-known pole problem that is present for traditional latitude–longitude grids (we will get back to

this point shortly). At the same time, a spectral element shallow water code was developed by Haidvogel et al. [131]. As a result, this methodology was extended to solving the hydrostatic primitive equations on the sphere in the Spectral Element Atmospheric Model (SEAM) [98]. By this time, massively distributed memory clusters had become available, thus motivating the development of highly scalable numerical methods; Fournier and coworkers noted the high parallel efficiency of spectral elements, thus making SEM a suitable numerical method for the dynamical core of climate models, which are computationally expensive. Taylor's SEM solver later became the basis for the NCAR's high-order method modeling environment (HOMME), which facilitated the rapid development of next generation atmospheric global circulation models (AGCM).

Taylor's SEM model utilized spherical coordinates to solve atmospheric problems on the sphere; however, since the sphere is a sub-manifold of three-dimensional space, Cartesian coordinates can also be used to solve problems on spheres provided that the fluid is constrained to lie on the sphere using a Lagrange multiplier [63]. Although computationally more expensive because there is an extra degree of freedom, this approach has two advantages over spherical coordinates: (1) analytical Jacobian transformations for the grid do not need to be derived and (2) any spherical grid (including the cubed sphere grid) may be utilized, thereby liberating the solver from the grid. Giraldo utilized this Cartesian SEM approach to solve the SWE using an Icosahedral grid in [113]. Collaborating with the Naval Research Lab, he applied this framework to solve the hydrostatic primitive equations in [119] and develop the U.S. Navy's spectral element atmospheric model (NSEAM) including a semi-implicit solver [114]. As we shall see in the next section, Giraldo and coworkers developed DG concurrently with SEM solvers, thereby exposing the common themes and machinery shared by the two methods. To prove how arbitrary grids can be used to solve the SWE on the sphere, a unified continuous/discontinuous Galerkin model has been recently presented in [211]. Using this model, the equations were solved in Cartesian coordinates using static and dynamic adaptivity using both, continuous and discontinuous Galerkin approximations on the grids illustrated in Fig. 12.

3.4.2 Discontinuous Galerkin

This subsection presents a short overview of the important steps in the historical development of DG towards atmospheric applications. A more general overview of the history of DG until the year 2000 can be found in [52]. Some information can also be found in the textbook by Hesthaven and Warburton [133].

The discontinuous Galerkin method was first introduced by Reed and Hill [248]. Reed and Hill were working on the solution of the stationary linear transport equation of neutrons with a constant velocity $\mathbf{v}$ in two dimensions:

$$\mu\frac{\partial\psi}{\partial x}+\eta\frac{\partial\psi}{\partial y}+\sigma\psi(x,y,\mu,\eta)=S(x,y,\mu,\eta), \tag{49}$$

where ψ is the angular neutron flux in the direction $(\mu,\eta)^{\mathrm{T}}=\mathbf{v}/||\mathbf{v}||_2$, the total macroscopic cross-section for neutron–nucleus interaction σ and the source term S. The source term describes scattering, fission and inhomogeneous sources. This transport equation was solved by Reed and Hill on a triangular mesh. They compared a method allowing discontinuities at the interfaces between different triangles with a continuous method and found that DG was computationally more expensive but slightly more accurate and much more robust. One of the main advantages of the discontinuous method was that it showed fewer oscillations at the boundary between areas with two different values of the cross-section σ. This allowed Reed and Hill to reduce the resolution of the method while still obtaining a less oscillatory result than the continuous method allowed.

The discontinuous method introduced in [248] was analyzed theoretically by Lesaint and Raviart [198]. In this early work the discontinuous Galerkin method was applied to linear equations. The first application of DG to nonlinear conservation equations is attributed to Chavent and Salzano [48]. Chavent and Salzano used first order polynomials for spatial discretization and a simple explicit Euler method for time discretization. A von Neumann analysis showed that this approach is unstable if the time step Δt is proportional to the grid spacing Δx. This approach becomes stable only for $\Delta t \propto \Delta x^{3/2}$. This severe restriction for the time step with explicit time integration was solved by the development of Runge–Kutta discontinuous Galerkin methods (RKDG) by Cockburn and Shu [53].

Discontinuous Galerkin methods were applied to parabolic equations by Jamet [163], displacement of oil by water in a porous slab by Chavent and Salzano [48], viscoelastic flows by Fortin and Fortin [97] and to the solution of the Maxwell equations by Warburton and Karniadakis [312].

The first numerical experiment using DG for the Euler equations of gas dynamics is reported in the 1991 work by Bey and Oden [26] and by Bassi and Rebay [12, 13]. The first application of DG to geophysical problems started with the work on shallow water equations by Schwanenberg et al. [264], followed by Giraldo et al. [116] who introduced inexact integration for DG and extended this to the sphere. In 2008, DG was finally used to solve the Navier–Stokes equations of non-hydrostatic atmospheric flows by Giraldo and Restelli [118]. Discontinuous

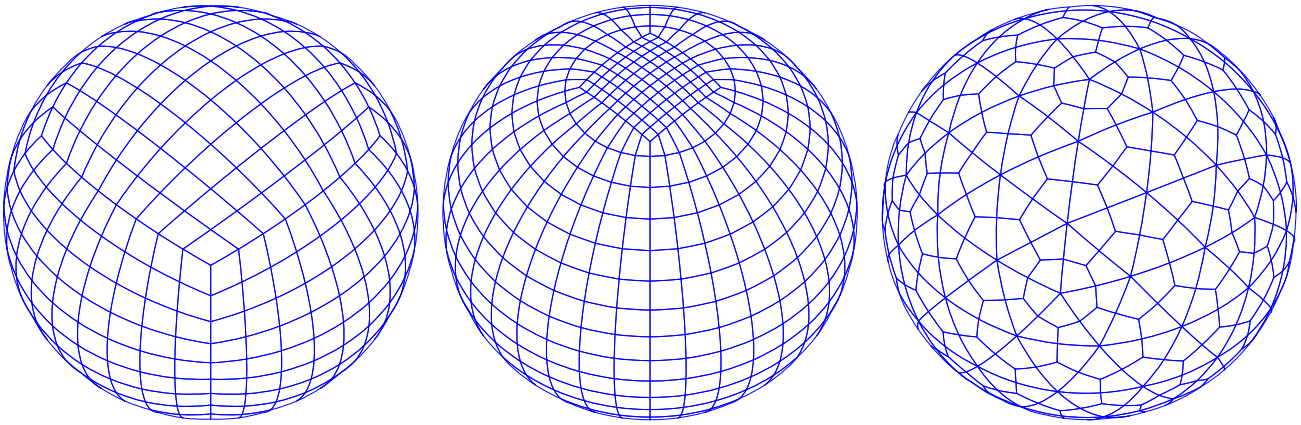

Fig. 12 Examples of spherical grids for the solution of the SWE. From *left* to *right*, classical cubed-sphere, a reduced longitude–latitude, and icosahedral grid. These are high order grids with curved elements on the spherical shell. Figures adapted from [211] with permission of John Wiley & Sons

Galerkin methods have not been used in operational global circulation models (GCM) yet. However, a hydrostatic GCM was presented by Nair et al. [225], followed in 2011 by the German DG model DUNE [34], and, in 2012, by the scalable non-hydrostatic model by Kelly and Giraldo [176]. The linear scalability properties of DG was shown by Wilcox et al. [320] and Kelly and Giraldo [176].

As mentioned earlier, the geometrical flexibility of EBG, including the potential for adaptivity, is a major strength of EBG methods–in particular, DG. An adaptive DG model based on non-conforming quads was introduced in [184]. This paper introduced a tree-based adaptive mesh refinement (AMR) strategy, demonstrated the potential for DG to achieve order-of-magnitude efficiency gains, which are difficult, if not impossible, with traditional finite difference or spectral transform methods. Dynamic adaptivity, using conforming triangular elements, was also explored in [224]. These two papers are not the first ones that report on grid adaptivity using EBG; however, they seem to be the first publications on this topic in the framework of non-hydrostatic atmospheric simulations using DG.

4 Stabilization of EBG for Advection-Dominated Problems

The straight numerical approximation of problems with dominating advection may result in unphysical oscillations in the solution. Finite and spectral element methods are no exceptions [170] and an error estimate of the standard Galerkin approximation of the problem proves it (see, e.g., [247]). Here, we show it by deriving the 1D finite element solution of the advection–diffusion problem with Dirichlet boundary conditions. The problem consists in solving

$$\frac{\partial q}{\partial t} + \mathscr{L}(q) = S, \tag{50}$$

where

$$\mathscr{L}(q) = \mathbf{u} \cdot \nabla q - \nabla \cdot (\nu \nabla q), \tag{51}$$

by linear $(p = 1)$ finite elements. In (65), ν is a positive, uniform, constant diffusivity coefficient, $\mathbf{u} = (u, 0, 0)$ is the velocity vector, and S is a source function. The domain of interest is the unit interval $\Omega = [0, 1]$. A uniform partition $\mathscr{P}^h$ of Ω with $N + 1$ nodes p_k, $k = 0, \ldots, N$, and $n_{el} = N$ elements K of length $h = \|\mathbf{p}_k - \mathbf{p}_{k-1}\|_2$ is assumed. For uniqueness of the solution, $q(0) = 0$ and $q(1) = 1$ are the assigned boundary conditions. Let $W^h \subseteq H^1$ be the space of piece-wise linear Lagrange polynomials of class $\mathscr{C}^0$ (Fig. 8, left) The projection of Eq. (50) onto W^h by the L^2 scalar product and integration by parts of the diffusion term yields the equation

$$\int_{\Omega^h} \psi^h \mathbf{u} \cdot \nabla q^h \, d\Omega^h + \int_{\Omega^h} \nu \nabla \psi^h \cdot \nabla q^h \, d\Omega^h = \int_{\Omega^h} \psi^h S \, d\Omega^h \quad \forall \, \psi^h \in W^h, \tag{52}$$

q^h is expanded by (29). When $S = 0$, the 1D finite element discretization of (52) yields the discrete equation

$$\left(\frac{u}{2} - \frac{\nu}{h}\right) \hat{q}_{k+1} + \frac{2\nu}{h} \hat{q}_k - \left(\frac{u}{2} + \frac{\nu}{h}\right) \hat{q}_{k-1} = 0, \quad k = 1, \ldots, N-1. \tag{53}$$

Equation (53) is equivalent to the 1D discretization of the same problem by second-order finite differences. After algebraic manipulation and given the definition of the element Péclet number

$$Pe = \frac{||\mathbf{u}|| \, h}{2\nu}, \tag{54}$$

Equation (53) is written as a function of (54):

$$(Pe-1)\,\hat{q}_{k+1}+2\,\hat{q}_k-(Pe+1)\,\hat{q}_{k-1}=0,\quad k=1,\ldots,N-1. \tag{55}$$

It represents a tridiagonal linear system in the unknowns $q_k,\quad k=1,N-1$, whose solution is the function (see [246])

$$\hat{q}_k=\frac{\left(\frac{1+Pe}{1-Pe}\right)^k-1}{\left(\frac{1+Pe}{1-Pe}\right)^N-1},\quad k=1,N-1. \tag{56}$$

The power of $(1+\mathrm{Pe})/(1-\mathrm{Pe})$ at the numerator produces an oscillatory behavior of the solution whenever $Pe>1$, as it is shown in Fig. 13. Pe is a linear function of h so that the grid, in principle, could be always constructed in such a way that, for a given value of u and v, $Pe\leq 1$. However, this is not viable for most real problems because of the extremely high number of grid points that may be necessary to achieve such a condition. The only way to solve the problem of numerical instabilities in the Galerkin solution of transport problems with dominant advection remains that of stabilization by proper means. A certain category of stabilization methods applied to the multi-dimensional advection–diffusion equation will be described in the next section.

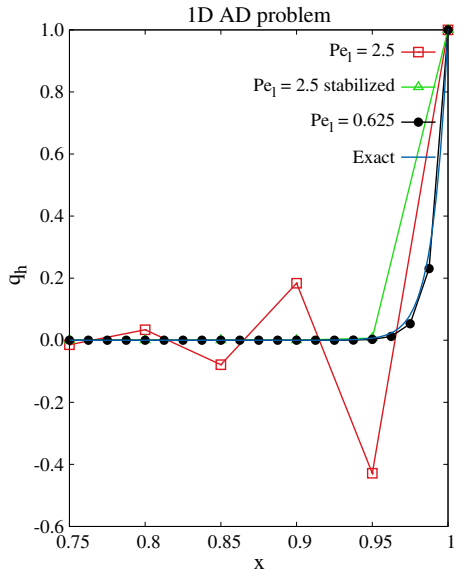

Fig. 13 Finite element solution of the advection diffusion problem (50) using uniform, linear elements. $u=10$, $v=0.1$, in a domain of unitary total length. With these values, the global Péclet is $Pe_g=50$. The plot shows the approximate solutions obtained for different grid spacing ($Pe=2.5$ and $Pe=0.625$) with and without stabilization. It is shown how the computed solution can approach the exact solution by either increasing the number of grid points ($Pe=0.625$), or by maintaining the grid sufficiently coarse but with the addition of a stabilizing term (How this term is built has not been shown yet, but the result gives a hint on what to expect from it)

4.1 Viscosity-Based Stabilization Techniques

Regardless of the numerical method that an atmospheric model is built on, dissipation of some sort is added for various reasons; stabilization is one of them. As is pointed out in [161], the most common mean of dissipation that is found in current research and operational weather forecast models is artificial diffusion (or hyper-diffusion, HV from now on) in the form of

$$HV=\int_{\Omega_h}(-1)^{\alpha+1}\psi^h\nabla^\alpha\cdot(\boldsymbol{\nu}_{2\alpha}\nabla^\alpha q^h)\,d\Omega^h \tag{57}$$

where α is a positive integer and $\boldsymbol{\nu}_{2\alpha}$ is the matrix of the diffusivity coefficients that may vary along different grid directions [125]. When $\alpha=1$, HV reduces to second-order *Artificial Viscosity* [169]. HV is easy to implement and is robust. These are features that make it attractive for models that are not allowed to break during a forecast simulation. HV is found in other fields of computational fluid dynamics as well; the work by [11] is an example where HV is used to stabilize the simulation of high speed flows. One justification that practitioners in NWP give to HV for $\alpha>1$ is its scale selectiveness; it damps higher order frequencies that are usually the result of numerical error and dispersion, but tends to leave the important and physical modes untouched. However, even if HV is indeed scale-selective, it is not physical. Since the artificial term given by Eq. (57) is a perturbation to the original equation, if this perturbation does not go to zero as $h\to 0$, the exact solutions of the original and of the perturbed problems are not equivalent. As is evident in Fig. 14, these methods add an uncontrolled and non-localized diffusion that yields a certain deterioration of the solution. For a stabilizing scheme to preserve the shape of the tracer, dissipation should be avoided in the direction normal to the flow and only act in the direction parallel to the flow [55, 146, 152].

To preserve the correct physical dimensions of the hyper-viscous term, the value of $\nu_{2\alpha}$ must scale with respect to α and the grid spacing. Its selection not only is non-trivial, but has a great impact on the solution of the problem. Jablonowski and Williamson [161] clearly state that "[...] the choice of the ∇^2,∇^4 coefficient is most often motivated by empirical arguments and chosen in a somewhat arbitrary manner [...]." More advanced, and by now classical, stabilizing schemes for finite elements, spectral elements, and discontinuous Galerkin are described in the following subsections.

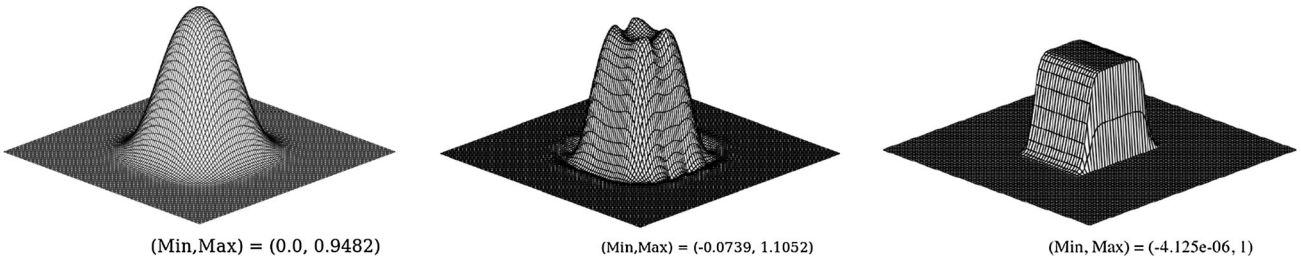

Fig. 14 Pure transport of a square wave in a 2D doubly-periodic channel. The velocity is directed along the x-axis (the *bottom-right* edge of the squared domain in the three plots). From *left* to *right* stabilization achieved by ∇^2, ∇^4, and using a variational multiscale scheme [143]. Adapted from Fig. 21 of [210], with permission by Elsevier

4.2 Filtering of (High-Order) EBG Methods

Both CG and DG, like all higher-order methods, are limited by *Godunov's Theorem*: all linear numerical methods for solving PDEs that do not generate additional extrema (so-called *monotone* schemes), are all first-order accurate. As an immediate consequence, high-order CG and DG are not monotonicity preserving, especially so near sharp gradients. In NWP, this problem is especially problematic for tracer transport, where mass fractions may become negative due to these artificial extrema. Finally, spectral elements and discontinuous Galerkin methods on quadrilateral and hexahedral elements typically use inexact integration to diagonalize the mass matrix; this approximate integration introduces errors which must be stabilized by a filter or a more sophisticated scheme such as the VMS method discussed later.

To circumvent these problems, *filters* were introduced in the development of both spectral methods [96] and, later, spectral elements in [32, 94]. They were also applied to discontinuous Galerkin methods in [116, 118]. Filters reduce the aliasing that occurs in the higher-order modes of the solution that are largely responsible for Gibbs oscillations; hence, filters act upon the modal representation of the solution. Once the offending high modes are eliminated, the modal solution is inverse transformed to physical, or nodal space. Hence, the spectral filtering operation consists of a three-step process: (1) transform the *nodal* solution to a *modal* solution, (2) apply a low-pass filter to eliminate the largest spatial frequencies, and (3) inverse transform the filtered modal solution to nodal space. For SE and DG that utilize LGL basis functions, a modified Legendre transform may be utilized for steps 1 and 3; in addition, it is possible to perform these operations on an element by element basis, thereby eliminating the need for a global assembly operation.

Ideally, the spectral filter applied during step 2 should eliminate all non-physical oscillations while faithfully preserving the physics of the solution. In practice, satisfying both of these requirements is not possible. An effective filter was developed by Boyd [31, 32], based on previous theoretical work by Vandeven [307], resulting in an *erfc-log filter* for polynomials of order p. Letting $\bar{\theta} = k/p - 1/2$, the filter $\sigma(k;p)$ is equal to unity if $\bar{\theta} \leq 0$ and

$$\sigma(k;p) = \frac{1}{2}\ \mathrm{erfc}\left[2\sqrt{p}\bar{\theta}\sqrt{\frac{-\log\left(1-4\bar{\theta}^2\right)}{4\bar{\theta}^2}}\right]. \tag{58}$$

Equation (58) rapidly eliminates all higher-order modes, and completely eliminates the highest order node.

Most DG methods utilize exact integration and therefore may not require filter-based stabilization. It should be kept in mind that filtering may not be sufficient to stabilize the solution; for this reason, the dissipation schemes described so far and later in this section are often considered as a possible option for high order methods as well [44, 45]. Finally, it is difficult to derive idempotent filters; when the filter is not idempotent [172], the solution may vary based on how many times the filter is applied along the simulation.

4.3 Towards Consistent Stabilization Methods

Streamline Upwinding (SU) [146], *Streamline Upwind Petrov–Galerkin* (SUPG) [38], *Galerkin/Least-Squares* (GLS) [150], Galerkin methods with bubble functions [9, 35, 37], or sub-grid projection methods [126] are some of the most used stabilization techniques for finite elements. To bypass some drawbacks of streamline-type schemes such as these, much work was done in the same years on shock capturing, as found in [132, 170, 171]. The Taylor–Galerkin method [78], the Characteristic-Galerkin formulation [241], and the Characteristic-Based Split (CBS) method [331, 332] are more ways for FE stabilization that, however, rely on a reasoning that has no relationship with the methods we are interested in reviewing in this article. We mention them here but we will not delve into their description.

4.3.1 Streamline-Upwind (SU)

The problem of isotropic smearing of the solution mentioned above was partially solved by [146] with the construction of the Streamline-upwind method, although the idea of finite element upwinding can already be found 1 year earlier with the work by [287] and then continued in [288, 289]. With SU, stabilization is projected in the direction of the flow only, as visible from

$$b_{SU} = \int_{\Omega_h} \tau \mathbf{u} \cdot \nabla \psi^h \mathbf{u} \cdot \nabla q^h \, d\Omega^h. \tag{59}$$

However, like HV, SU is not numerically consistent either in the sense that no residual information is used to construct this perturbation term. The Streamline-upwind/Petrov–Galerkin (SUPG) method described below is the consistent evolution of SU and will be among the most common methods of stabilization of finite elements used since its introduction.

4.3.2 Streamline-Upwind/Petrov–Galerkin (SUPG)

The SUPG method was designed by [38] and was later generalized for multidimensional problems by [151]. It is a consistent alternative to the HV approach or to the overly diffusive SU. Its use has been ubiquitous in the solution of transport problems by the finite element method (e.g., [35, 100, 156, 295]). The application of this strategy to higher-order schemes was first tested for spectral methods by Canuto et al. [43], [44, 46], [45], and later by Hughes et al. [148] using *non-uniform rational B-splines* (NURBS). SUPG is a Petrov–Galerkin method in that it does not assume that the basis and test functions live in the same space. We introduce the additional space Ψ^h of test functions w^h defined by

$$\Psi^h \doteq \left\{ w^h \; : \; w^h = \psi^h + \tau \mathbf{u} \cdot \nabla \psi^h \; : \quad \psi^h \in W^h \right\}.$$

We have the problem of finding the function $q^h \in W^h$ such that

$$\int_{\Omega^h} w^h \mathbf{u} \cdot \nabla q^h \, d\Omega^h - \int_{\Omega^h} w^h \nabla \cdot (\nu \nabla q^h) \, d\Omega^h = \int_{\Omega^h} w^h S^h \, d\Omega^h \quad \forall \, w^h \in \Psi^h. \tag{60}$$

Some algebra and rearrangement of (60) yields the problem of finding $q^h \in W^h$ such that

$$\underbrace{\int_{\Omega^h} \psi^h \mathbf{u} \cdot \nabla q^h \, d\Omega^h + \int_{\Omega^h} \nu \nabla \psi^h \cdot \nabla q^h \, d\Omega^h}_{\text{Galerkin}} + b_{SUPG} = \underbrace{\int_{\Omega^h} \psi^h S^h \, d\Omega^h}_{\text{Galerkin}} \quad \forall \, \psi^h \in W^h, \tag{61}$$

where

$$b_{SUPG} = \int_{\Omega} \left(\mathbf{u} \cdot \nabla \psi^h \right) \tau \underbrace{\left[\mathbf{u} \cdot \nabla q^h - \nabla \cdot \left(\nu \nabla q^h \right) - S \right]}_{\mathscr{L}(\mathrm{q^h}) - \mathrm{S}} \, d\Omega^h \tag{62}$$

is the consistent SUPG stabilizing term. In (62), $\mathbf{u} \cdot \nabla q^h - \nabla \cdot \left(\nu \nabla q^h \right) - S$ is the residual of (50) and τ is the stabilization parameter. The definition of τ that yields a nodally exact SUPG solution with continuous piecewise linear finite elements is derived in [50] from 1D analysis. Its generalization to multi-dimensional problems is given by the simple substitution of u with $||\mathbf{u}||$, although, in multi-dimensions this does not necessarily imply nodal exactness. With respect to higher order elements, a thorough analysis of τ for quadratic elements is given by [59]. In the context of VMS, different definitions exist for parameter τ, some of them are discussed in 4.4. For a brief review on SUPG, the reader is also referred to [145] and the report [102].

4.3.3 Galerkin/Least-Square (GLS)

A generalization of SUPG was obtained by [150] as

$$b_{GLS} = \int_{\Omega} \underbrace{\left[\mathbf{u} \cdot \nabla \psi^h - \nabla \cdot \left(\nu \nabla \psi^h \right) \right]}_{\mathscr{L}(\psi^h)} \tau \times \underbrace{\left[\mathbf{u} \cdot \nabla q^h - \nabla \cdot \left(\nu \nabla q^h \right) - S \right]}_{\mathscr{L}(\mathrm{q^h}) - \mathrm{S}} \, d\Omega^h. \tag{63}$$

In analogy with the findings of [82] to stabilize the Stokes equation, a sign change in the Laplace term of the stabilizing term in the perturbed equation proved to yield better stabilization characteristics (more accurate results) than the original generalized SUPG (or GLS) method [100]. In (63), for better properties, instead of using the differential operator $\mathscr{L}$, the method should use the negative part of the adjoint $\mathscr{L}^*$ of the original operator $\mathscr{L}$. We have that the last perturbation term of the original AD equation should be

$$b = - \int_{\Omega} \mathscr{L}^*(\psi^h) \tau \underbrace{\left[\mathbf{u} \cdot \nabla q^h - \nabla \cdot \left(\nu \nabla q^h \right) - S \right]}_{\mathscr{L}(\mathrm{q^h}) - \mathrm{S}} \, d\Omega^h, \tag{64}$$

where

$$\mathscr{L}^* = -\mathbf{u} \cdot \nabla - \nabla \cdot (\nu \nabla) \tag{65}$$

is the adjoint of $\mathscr{L}$.

Based on what was learned on stabilization of the scalar advection–diffusion equation, researchers in fluid

dynamics applied these methods and their evolution to the stabilization of fluid problems. Thanks to the work of [143, 149], the methods that we have just described have been recognized to belong to the same family known as the family of Variational Multiscale Stabilization, or VMS. The VMS approach is summarized below for scalar advection problems, and will be derived and discussed for the Euler equations in Sect. 4.7.

4.4 Variational Multiscale Stabilization (VMS)

In 1995 and 1996, groups of researchers lead by Hughes [143] and Brezzi [36] proposed a theory to explain the reasons of instabilities and a new way to attack the problem. They concluded that the unresolved scales (the scales that cannot be captured by the computational grid) are responsible for the numerical instabilities of the Galerkin solution of the differential problem. The analysis, that continues with [155] and [149], forms the unifying theory of all stabilized finite element methods. According to this theory, stabilized methods are subgrid scale models where the unresolved scales are intimately related to the instabilities at the level of the resolved scales, and thus should be used in the construction of the stabilization term. More specifically, in the formulation of the discrete problem, the effects of the unresolved scales must be introduced by modeling them on the grid. These schemes are known as *Variational Multiscale Stabilization* (VMS) method. Generally speaking, the stabilization term of VMS corresponds to b defined in (64).

VMS has been extensively applied to the solution of the advection–diffusion/advection–diffusion–reaction equations (e.g. [58, 61, 141, 143, 149]), and to the solution of the Navier–Stokes equations for incompressible flows (e.g. [6, 17, 56, 57, 123, 153]). Recently, it was applied to spectral elements in the context of atmospheric flows in [209, 210]. In Sect. 4.7 a review of VMS for the compressible Euler equations is found.

The multiscale description of the stabilization scheme relies on the splitting of the solution into a resolved, q^h, and a sub-grid, unresolved component, $\tilde{q}$, to give $q = q^h + \tilde{q}$. Let $W = W^h \oplus \tilde{W}$ be the space decomposition such that $\tilde{W}$ completes W^h in W. This translates into the decomposition of the solution variables $q = q^h + \tilde{q}$, and of the basis functions $\psi = \psi^h + \tilde{\psi}$. Substituting the decomposition into the general weak form of Eq. (50),

$$\left(\psi, \frac{\partial q}{\partial t}\right) + a(\psi, q) = (\psi, S) \quad \forall\, \psi \in W, \tag{66}$$

where $(\cdot,\cdot)$ is the L^2 inner product and $a(\cdot,\cdot)$ is a bilinear form that satisfies

$$a(\psi, q) = (\psi, u\nabla q) + \nu\,(\nabla\psi, \nabla q),$$

and anticipating that we will consider the quasi-static approximation $\partial_t \tilde{q} = 0$ [57], we obtain:

$$\left(\psi^h + \tilde{\psi}, \frac{\partial q^h}{\partial t}\right) + a(\psi^h + \tilde{\psi}, q^h + \tilde{q}) = (\psi^h + \tilde{\psi}, S) \quad \forall\, \psi^h \in W^h,\ \tilde{\psi} \in \tilde{W}. \tag{67}$$

By virtue of the linear independence of ψ^h and $\tilde{\psi}$ we can first take $\tilde{\psi} = 0$ and then $\psi^h = 0$ and find the split problem:

$$\left(\psi^h, \frac{\partial q^h}{\partial t}\right) + a(\psi^h, q^h) + a(\psi^h, \tilde{q}) = (\psi^h, S) \quad \forall\, \psi^h \in W^h \tag{68a}$$

$$\left(\tilde{\psi}, \frac{\partial q^h}{\partial t}\right) + a(\tilde{\psi}, q^h) + a(\tilde{\psi}, \tilde{q}) = (\tilde{\psi}, S) \quad \forall\, \tilde{\psi} \in \tilde{W}. \tag{68b}$$

In the subgrid Eq. (68b) we come back to the original differential operator $\mathscr{L}$ from Eq. (50). We assume that $\tilde{\psi}(\partial K) = 0$ and $\tilde{q}(\partial K) = 0$, for each element K of the grid and, following [143], in (68a) we integrate by parts the bilinear forms that depend on the subgrid scale and find:

$$\left(\psi^h, \frac{\partial q^h}{\partial t}\right) + a(\psi^h, q^h) + (\mathscr{L}^*\psi^h, \tilde{q}) = (\psi^h, S) \quad \forall\, \psi^h \in W^h \tag{69a}$$

$$\left(\tilde{\psi}, \frac{\partial q^h}{\partial t}\right) + (\tilde{\psi}, \mathscr{L} q^h) + (\tilde{\psi}, \mathscr{L}\tilde{q}) = (\tilde{\psi}, S) \quad \forall\, \tilde{\psi} \in \tilde{W}, \tag{69b}$$

where $\mathscr{L}^*$ [in Eq. (65)] is the adjoint operator of $\mathscr{L}$.

4.5 Approximation of the Sub-grid Scales

The unresolved quantity $\tilde{q}$ has not been defined yet. Equation (69b) is used as the starting point to approximate $\tilde{q}$. By re-arranging the terms in (69b), the equation for the subgrid scales is found,

$$(\tilde{\psi}, \mathscr{L}(\tilde{q})) = (\tilde{\psi}, R(q^h)) \quad \forall\, \tilde{\psi} \in \tilde{W}, \tag{70}$$

where

$$R(q^h) = S - \frac{\partial q^h}{\partial t} - \mathscr{L}(q^h) \tag{71}$$

is the residual of the original equation. The strong form of (70) is considered on each element K

$$\mathscr{L}(\tilde{q}) = R(q^h), \tag{72}$$

and $\tau \approx \mathscr{L}^{-1}$, an algebraic approximation of the inverse of

the differential operator $\mathscr{L}$ is defined. Then the sub-scale has the form

$$\tilde{q} = \tau R(q^h). \tag{73}$$

Expression (73) is plugged into equation (69a), to find the expression for the VMS stabilized Galerkin method as follows: Find $q^h \in W^h$ such that

$$\left(\psi^h, \frac{\partial q^h}{\partial t}\right) + a(\psi^h, q^h) + (\mathscr{L}^*(\psi^h), \tau R(q^h)) = (\psi^h, S) \quad \forall\, \psi^h \in W^h. \tag{74}$$

Equation (74) differs from Eq. (66) by the additional term that models the subgrid scales. The extra term is the viscous-like contribution that stabilizes the equation.

Different formulations for $\tilde{q}$ are found in the literature, some of them are reviewed in 4.5.1–4.5.3. All of them depend on the definition of the stabilization parameter τ. The parameter τ is a topic of active research still today, since a general definition is not known [168]. This statement is true for all the residual-based stabilization methods described so far. The quantity τ is an intrinsic time that is built as a function of the local Pèclet number of the flow which, for stability, should respect the condition $Pe < 1$. Many problems in atmospheric CFD are advection dominated, implying $Pe \gg 1$, so that stabilization is indeed necessary for all the problems that are of any interest for atmospheric modelers.

4.5.1 Approximation Via Green's Functions

This approach is used by Hughes and collaborators [143, 144, 149] to derive τ. In brief, they consider Eq. (72) with $\tilde{q} = 0$ on ∂K, and the associated Green's function problem for the adjoint operator

$$\begin{cases} \mathscr{L}^* g(x,y) = \delta(x,y) & \forall\, x \in K \\ g = 0 \quad \text{on } \partial \mathrm{K}. \end{cases} \tag{75}$$

Then a uniform element-wise definition of τ is obtained as the average value of the exact element Green's function

$$\tau = \frac{1}{|K|^2} \int_K \int_K g(x,y)\, dK_x\, dK_y, \tag{76}$$

where $|K|$ is the measure (volume/area/length) of the domain. For the one-dimensional linear scalar advection–diffusion equation there is an analytical expression for the Green's function and the stabilization parameter is computed from (76) to give

$$\tau = \frac{1}{2}\frac{h}{||\mathbf{u}||}\left(\coth(Pe) - \frac{1}{Pe}\right), \tag{77}$$

for the Péclet number (54). In [38, 147] the same expression (77) is obtained in the context of SUPG stabilization by following an error minimization criterion. For the purely advective case ($\nu = 0$ and $Pe \to \infty$), we find

$$\tau = \frac{1}{2}\frac{h}{||\mathbf{u}||}. \tag{78}$$

Instead of (76), Corsini et al. [61] propose a non-uniform τ on each element:

$$\tau(x) = \frac{1}{|K|} \int_K g(x,y)\, dK_y. \tag{79}$$

4.5.2 Approximation Via Fourier Analysis

The strategy of Codina et al. [58] is explained here for the multidimensional advection–diffusion Eq. (50). The starting point is to transform Eq. (72) into the Fourier space. Which is interesting because the differential operator transform, $\widehat{\mathscr{L}}$, is easy to invert. Let's call $\mathscr{T}$ its inverse, $\mathscr{T} = (\widehat{\mathscr{L}})^{-1}$, thus the Fourier transform of the sub-scale is approximated on each element K as

$$\hat{\tilde{q}}(\boldsymbol{\omega}) = \mathscr{T}(\boldsymbol{\omega})\, \hat{R}(\boldsymbol{\omega}), \tag{80}$$

where $\boldsymbol{\omega}$ is the wave number and

$$\mathscr{T}(\boldsymbol{\omega}) \approx \left(i\,\frac{\mathbf{u}\cdot\boldsymbol{\omega}}{h} + \nu\,\frac{||\boldsymbol{\omega}||^2}{h^2}\right)^{-1}.$$

Considering expressions (80) and (73), the Plancherel's formula and the mean value theorem are applied to obtain an approximated value for the stabilization parameter on each element K as

$$\tau = \frac{1}{2}\frac{h}{||\mathbf{u}||}\frac{Pe}{Pe+1} = \left(\frac{2||\mathbf{u}||}{h} + \frac{4\nu}{h^2}\right)^{-1}. \tag{81}$$

For pure advection problems ($\nu = 0$ and $Pe \to \infty$), the stabilization parameter becomes as in Eq. (78).

4.5.3 Approximation Via Bubble Functions

More options to build the stabilization parameter τ are found in [141] for linear, quadratic, and cubic elements. The space $\tilde{W}$ is made of bubble functions (see [9, 37]), that are vanishing functions on the boundaries of each element. The unresolved scales, $\tilde{q}$, are defined as a function of the bubbles $b(x)$ that are derived as described in the referenced literature. Omitting the details, τ is a function of the bubble as:

$$\tau = \frac{1}{h}\int_0^h b(x)\, dx, \tag{82}$$

that, once evaluated, yields the expression (77) for the parameter τ; the same expression encountered with the Green's function approach.

On the steps of [141], τ for spectral elements of arbitrary order and with unequally spaced element nodes was derived in [210]. The stabilization parameter τ is built inside the element as a function of the bubbles on every segment delimited by two consecutive LGL points. The uneven spacing of the element nodes is the major difference with respect to the definitions derived in previous studies. In this case, the intrinsic time is non-uniform along the element.

4.6 Preserving Positivity

SUPG, GLS, and VMS are not monotonicity preserving. This issue is particularly important for the simulation of tracer dynamics in the atmosphere. If over- and under-shoots affect the solution in the proximity of strong gradients, the net mass balance of the advected tracers, will be negatively affected. To overcome this issue, a controlled crosswind discontinuity capturing can be added to the principal stabilization scheme. For example, the method introduced in [55] was successfully adapted to high-order spectral elements in [210] for standard 2D test cases and, more recently, by [209] to support positivity in the solution of fully 3D cloud simulations. For comparison, we reproduce Fig. 29 of [210] in Fig. 15. In the case of high-order SEM, in [210] the First Order Subcell (FOS) method was introduced. FOS consists in lowering the high-order method to first order in the spectral elements that contain the discontinuity only. The FOS results are encouraging, although there is some overhead coming from looping over the linear sub-elements within the high-order elements that contain the localized over- and under-shoots.

Other positivity preserving schemes such as high-order limiters for both CG and DG are often used as well, as is shown in, e.g., the report by [328].

4.7 VMS Stabilization for the Euler Equations

VMS for compressible flows appears in [222, 251, 252]. A review of residual-based stabilization methods for compressible flows can be found in [145]. Recently, VMS was used to stabilize the FEM solution of the Euler equations of atmospheric non-hydrostatic flows in [212, 213]. VMS was derived for discontinuous Galerkin as well [154] although it has not been applied to atmospheric modeling. In the following, we then limit the analysis to continuous Galerkin (without distinguishing between FEM and SEM in its derivation). For the treatment that follows, it is convenient to express the Euler equations [i.e., the inviscid counterpart of system (1)] in compact form as

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}^i(\mathbf{q})}{\partial x_i} = 0, \quad \mathrm{i} = 1, 2, 3, \tag{83}$$

where the Einstein summation on the repeated indices is assumed, where $\mathbf{q}$ is the vector of the unknowns, and $\mathbf{F}$ is the vector of the flux quantities. Without compromising the generality of the stabilization method, gravity and Coriolis are here omitted. As usual, the problem consists in finding $\mathbf{q}(\mathbf{x}, t)$ that verifies Eq. (83) for all $(\mathbf{x}, t) \in \Omega \times \mathbb{R}^+$. To proceed and derive VMS applied to this set, we write the three-dimensional Euler equations in flux form for the conservative variables $\mathbf{q}$ and define the advective system:

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{A}^i(\mathbf{q}) \, \frac{\partial \mathbf{q}}{\partial x_i} = 0, \tag{84}$$

where

$$\mathbf{A}^i(\mathbf{q}) = \frac{\partial \mathbf{F}^i}{\partial \mathbf{q}} \tag{85}$$

are the Jacobian matrices. As already done in Sect. 3 for scalar problems, the variational form of Eq. (84) can then be written as

$$\int_{\Omega^h} \boldsymbol{\psi}^h \cdot \frac{\partial \mathbf{q}^h}{\partial t} \, d\Omega^h + \int_{\Omega^h} \boldsymbol{\psi}^h \cdot \mathbf{A}^i(\mathbf{q}^h) \, \frac{\partial \mathbf{q}^h}{\partial x_i} \, d\Omega^h = 0 \quad \forall \, \boldsymbol{\psi}^h \in W^h. \tag{86}$$

As it is done in page 40, the decompositions $\mathbf{q} = \mathbf{q}^h + \tilde{\mathbf{q}}$ and $\boldsymbol{\psi} = \boldsymbol{\psi}^h + \tilde{\boldsymbol{\psi}}$ are plugged into the variational problem (86), which hence can be split into the two equations

$$\int_{\Omega^h} \boldsymbol{\psi}^h \cdot \frac{\partial \mathbf{q}^h}{\partial t} \, d\Omega^h + \int_{\Omega^h} \boldsymbol{\psi}^h \cdot \mathbf{A}^i(\mathbf{q}) \, \frac{\partial \mathbf{q}^h}{\partial x_i} \, d\Omega^h + \sum_{m=1}^{n_{\mathrm{el}}} \left(\int_{K^m} \boldsymbol{\psi}^h \cdot \frac{\partial \tilde{\mathbf{q}}}{\partial t} \, dK^m + \int_{K^m} \boldsymbol{\psi}^h \cdot \mathbf{A}^i(\mathbf{q}) \, \frac{\partial \tilde{\mathbf{q}}}{\partial x_i} \, dK^m \right) = 0 \quad \forall \, \boldsymbol{\psi}^h \in W^h \tag{87a}$$

$$\sum_{m=1}^{n_{\mathrm{el}}} \int_{K^m} \tilde{\boldsymbol{\psi}} \cdot \left(\frac{\partial \tilde{\mathbf{q}}}{\partial t} + \mathbf{A}^i(\mathbf{q}) \, \frac{\partial \tilde{\mathbf{q}}}{\partial x_i} \right) dK^m = \sum_{m=1}^{n_{\mathrm{el}}} \int_{K^m} \tilde{\boldsymbol{\psi}} \cdot \mathbf{R}(\mathbf{q}^h) \, dK^m \quad \forall \, \tilde{\boldsymbol{\psi}} \in \widetilde{W} \tag{87b}$$

where

$$\mathbf{R} = -\left(\frac{\partial}{\partial t} + \mathbf{A}^i(\mathbf{q}) \, \frac{\partial}{\partial x_i} \right) \tag{88}$$

is the residual operator of the governing Eq. (84). Equation (87a) is solved numerically on the computational grid, whereas (87b) is the subgrid scale equation from which an

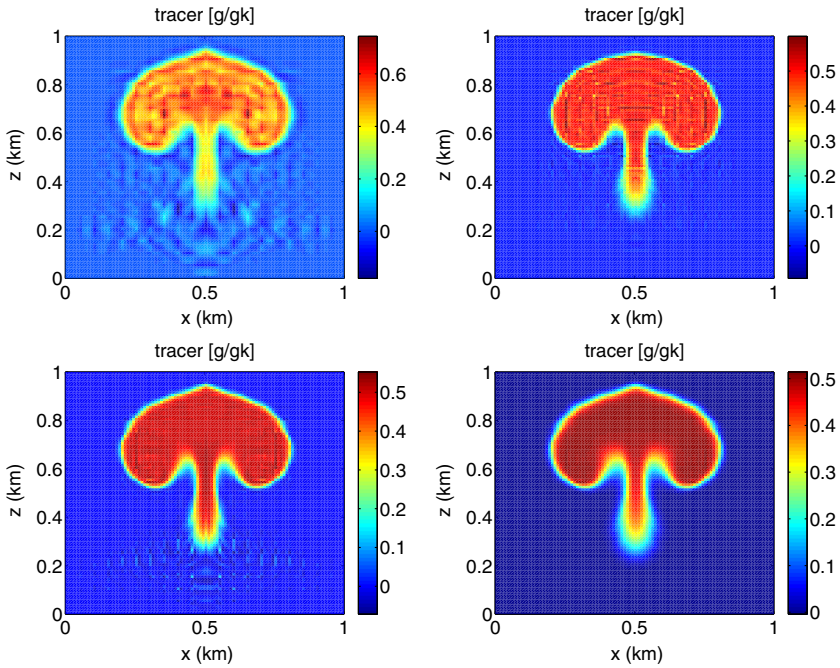

Fig. 15 Stable SEM solutions of the transport equations for a sharp cylinder that is transported by a rising thermal flow (i.e. the tracer is driven by the thermal flow that is modeled by the Euler equations of stratified flows). *Top left* filtered solution using the filter of Sect. 4.2. *Top right* 2nd-order artificial diffusion without discontinuity capturing. *Bottom left* VMS. *Bottom right* VMS with discontinuity capturing. Adapted from [210], with permission by Elsevier

expression for $\tilde{\mathbf{q}}$ is obtained and hence plugged back into (87a). Concerning Eq. (87a) for the large scales, some assumptions should be made. For details, see the referenced literature. In the case of non-viscous problems (i.e. Euler equations), SUPG [194], GLS [267] and VMS end up having the same structure, unless the approximation of the subgrid scales, $\tilde{\mathbf{q}}$, is such that VMS differentiates itself from the other two schemes.

4.7.1 VMS as Implicit Large Eddy Simulation (LES)

Without entering much into this discussion, it is important to underline the fact that VMS is also used as an Implicit Large Eddy Simulation (ILES) scheme that relies on the variational projection of the original equations rather than the traditional filtering. This was first applied to incompressible turbulent flow in [17, 153]. In [91, 183, 234], a turbulent compressible flow is modeled using the VMS framework although the fine scales are modeled by a Smagorinsky model. Similarly, this is done in [60, 199]. In [306], a VMS formulation obtained by extension of the Favre averaging to general projection operators is proposed, where no explicit subgrid modeling is presented. Using SEM, VMS was used in [120] to solve turbulent incompressible flows.

4.7.2 Approximation of the Sub-grid Scales

For the Navier–Stokes equations, analogously to the advection–diffusion equation, the subgrid scale $\tilde{\mathbf{q}}$ is computed from the subscale Eq. (87b) and has the general form of

$$\tilde{\mathbf{q}} = \boldsymbol{\tau}\, \mathbf{R}(\mathbf{q}^h), \tag{89}$$

where $\boldsymbol{\tau}$ is a diagonal matrix. Shakib et al. [267] and Hughes and Mallet [151] compute the parameter τ for GLS to solve the compressible Euler and Navier–Stokes equations. For the same equations, Hughes and Tezduyar [156] and Le Beau and Tezduyar [194] compute $\boldsymbol{\tau}$ for SUPG. Just like for the scalar case, the parameter $\boldsymbol{\tau}$ has been derived in different ways by different authors, although the final expressions seldom differ greatly. In, e.g., [222], $\boldsymbol{\tau}$ is derived from a Fourier analysis. Another approach involves the use of Green's functions as done in [61]. Regardless of the definition of $\boldsymbol{\tau}$, let us notice the local nature of the subscales that only exist where the residuals of the large scales are important. This, with non-constant values, marks the major difference with respect to artificial diffusion. The structure of $\tilde{\mathbf{q}}$ for the problem of a rising thermal is shown in the top two plots of Fig. 16. By comparison with the pattern of potential temperature (bottom left plot) and horizontal velocity (bottom right plot), the structure of the sub-grid scale is clearly tied to the residual.

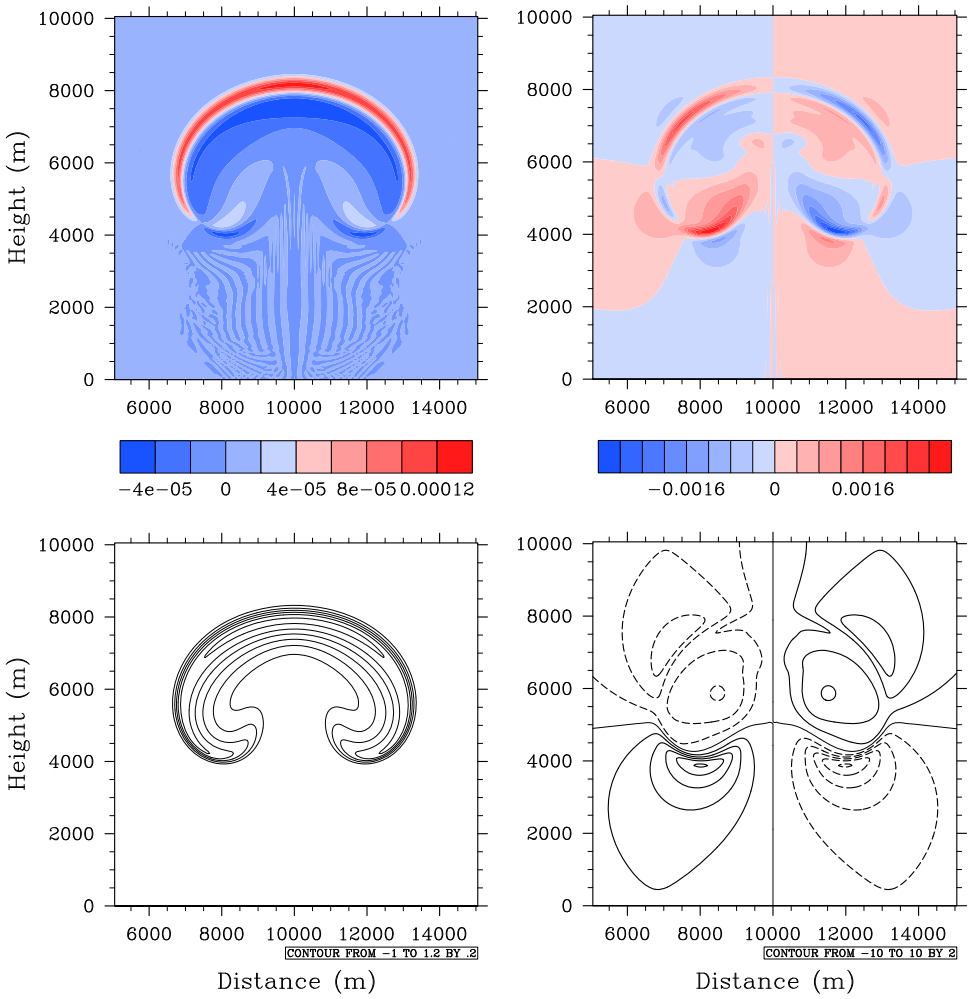

Fig. 16 2D Rising thermal bubble. *Top row* sub-grid scales: $\tilde{\theta}/\theta_{max}$ (*top-left*) and $\tilde{U}/U_{max}$ (*top-right*). *Bottom row* potential temperature θ' (K) (*left*), and horizontal velocity, u (m/s). This vertically displacing flow is triggered by the thermal perturbation θ' of a neutrally stratified flow (i.e. uniform and constant θ_0). The characteristic shape of the perturbation field θ' is shown in the right panel. The plots are adapted from [212]

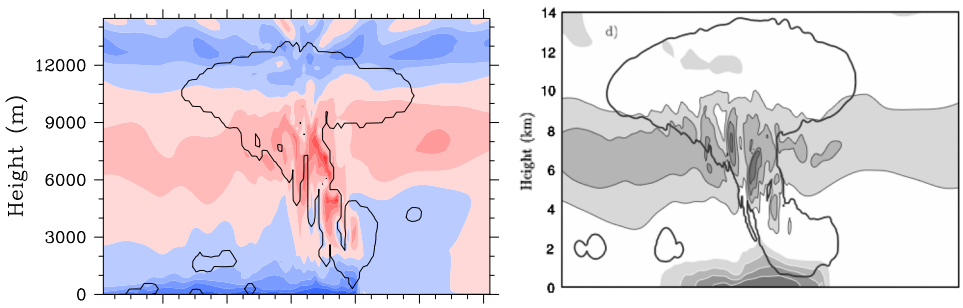

Fig. 17 2D squall line simulation. The cloud content is delimited by the *thick black contour line* in both plots. The *color*/*grey shading* in the *left*/*right plots* is the equivalent potential temperature. Both horizontal domains extend along 240 km. The *left figure* is adapted from [213] where a VMS stabilized FEM solution was computed with linear elements (with permission by Elsevier). The *right figure* is adapted from [103] (with permission of the American Meteorological Society) and the solution was computed with 8th-order SEM stabilized with a constant coefficient diffusion ($\nu = 200\,\mathrm{m}^2/\mathrm{s}$) and a filter

An example of simulation where VMS was used to stabilized both the dynamics (Euler equations) and the advection–diffusion equations of water tracers is shown in Fig. 17, and is compared against the solution obtained in [103] with a filtered high-order spectral element at an equivalent resolution.

4.8 Alternative Consistent Schemes: Spectral Vanishing Viscosity and Entropy Viscosity Method

Also based on a second order operator, the spectral vanishing viscosity (SVV) of [290] is a stabilizing method used by practitioners of high order spectral element and spectral Fourier methods. The idea of SVV comes from an entropy analysis of the problem at hand and is such that the added dissipation satisfies the entropy condition. For more on this, please, see [130, 173, 290] and references therein.

Also tied to the entropy equation, the entropy viscosity method was first introduced by [128]. The fundamental difference between this method and SVV is in the way the entropy equation is used. The entropy viscosity method builds the local and dynamic viscosity of the equations based on the residual of the associated entropy equation [127, 129]. In [334] we find how this regularization of the governing equations is applied to the discontinuous Galerkin method as well.

We finally comment on the use of adaptive viscosity methods for both CG [125] and DG [236]. These two methods are not consistent, but are element-based and dynamic. An adaptive artificial viscosity for non-hydrostatic modeling using DG has been recently proposed in [328].

4.9 Physics-Based Stabilization

A computationally inexpensive and numerically stable subgrid scale model for compressible large-eddy simulation was introduced in [229] for adaptive finite elements. Due to its stabilizing properties, this method was easily adapted to the solution of low Mach number atmospheric flows via high order spectral elements in [215] and [216]. Like VMS, this method is a residual-based alternative to the more classical artificial diffusion in a way that not only is numerically consistent, but could also serve as a turbulence model. Unlike VMS, however, stabilization is attacked starting from the governing equations rather than from their numerical approximation. More specifically, the Euler equations are first filtered to separate the resolved from the un-resolved scales [108, 260]. The filtering operation leads to a new set of equations containing additional terms that are dissipative in nature and that are then modeled in some way. The steps described below (following the treatment of [216]) show how stabilization is then achieved. In LES, given a quantity q (e.g., density, velocity, potential temperature), its large scale (grid resolved) component $\overline{q}$ is obtained via the application of the filter

$$\overline{q}(\mathbf{x}) = \int_{\Omega} G_{\Delta}(\mathbf{x}-\boldsymbol{\chi}) q(\boldsymbol{\chi}) d\boldsymbol{\chi}. \tag{90}$$

Equation (90) is a spatial convolution of the filtering function $G_{\Delta}$with q, where Δ is the filter width. The filter functions can vary; the most commonly used in LES are the Gaussian, the top hat in real space, and the sharp Fourier cutoff functions [197, 240].

Remark 4.1 The barred quantities introduced in Sect. 2.1.2 have no relation with $\overline{q}$ defined in Eq. (90) for Large Eddy Simulation (LES). LES implies a separation between the resolved and unresolved scales, whereas the splitting given in Sect. 2.1.2 was introduced for numerical convenience in the simulation of atmospheric problems and does not affect the way LES is constructed or derived.

For compressible flows, the Favre filter $\widetilde{q} = \overline{\rho q}/\overline{\rho}$ [92], although not necessary, is classically introduced. The application of these filters to equations (6)—excluding the Coriolis terms for simplicity—yield the filtered system

$$\frac{\partial \overline{\rho}\widetilde{\mathbf{u}}}{\partial t} + \nabla \cdot (\overline{\rho}\widetilde{\mathbf{u}} \otimes \widetilde{\mathbf{u}}) + \nabla \overline{p} = \nabla \cdot \boldsymbol{\tau} - \overline{\rho}\mathbf{g}, \tag{91a}$$

$$\frac{\partial \overline{\rho}}{\partial t} + \nabla \cdot (\overline{\rho}\widetilde{\mathbf{u}}) = 0, \tag{91b}$$

$$\frac{\partial \overline{\rho}\widetilde{\theta}}{\partial t} + \nabla \cdot \left(\overline{\rho}\widetilde{\theta}\widetilde{\mathbf{u}}\right) = \nabla \cdot \mathbf{Q}, \tag{91c}$$

where the two derivatives on the right-hand sides of (91a) and (91c) represent the contributions of the unresolved scales. If Favre filtering were not applied, an additional flux term would also appear on the right-hand side of Eq. (91b). With Favre, the filtered density $\overline{\rho}$ is conserved and no modeling is required for the continuity equation. In (91a), $\boldsymbol{\tau}$ is the turbulent stress tensor,

$$\boldsymbol{\tau} = \overline{\rho}\left(\widetilde{\mathbf{u} \otimes \mathbf{u}} - \widetilde{\mathbf{u}} \otimes \widetilde{\mathbf{u}}\right),$$

approximated by

$$\boldsymbol{\tau} = 2\mathbb{D}(\widetilde{\mathbf{u}}), \tag{92}$$

where

$$\mathbb{D}(\widetilde{\mathbf{u}}) = \frac{\mu_n}{2}(\nabla\widetilde{\mathbf{u}} + \nabla\widetilde{\mathbf{u}}^{\mathrm{T}})$$

is the velocity deformation tensor multiplied by a dynamic coefficient μ_n that will be defined shortly. Similarly, in (91c) $\mathbf{Q}$ is the kinematic heat flux defined as

$$\mathbf{Q} = \overline{\rho}\left(\widetilde{\theta\mathbf{u}} - \widetilde{\theta}\widetilde{\mathbf{u}}\right), \tag{93}$$

and is modeled via

$$\mathbf{Q} = \kappa_n \nabla \widetilde{\theta} \tag{94}$$

Like μ_n, the definition of κ_n determines the method proposed in [229]. The coefficients μ_n and κ_n are calculated

element-wise on every high order element Ω_e for a Galerkin approximation of Equations (91). More specifically, for the sensible temperature $T = \theta(p/p_0)^{R/c_p}$ and one finite/spectral element of characteristic length h_{Ω_e}, we start by defining the dynamic viscosities

$$\mu_{\max}|_{\Omega_e} = 0.5 h_{\Omega_e} \| |\widetilde{\mathbf{u}}| + \sqrt{\gamma c_p \widetilde{T}} \|_{\infty,\Omega_e}, \tag{95}$$

and

$$\mu_{\text{res}}|_{\Omega_e} = h_{\Omega_e}^2 \max\left(\frac{\|R(\overline{\rho})\|_{\infty,\Omega_e}}{\|\overline{\rho} - \widehat{\rho}\|_{\infty,\Omega}}, \frac{\|R(\overline{\rho}\widetilde{\mathbf{u}})\|_{\infty,\Omega_e}}{\|\overline{\rho}\widetilde{\mathbf{u}} - \overline{\rho}\widehat{\mathbf{u}}\|_{\infty,\Omega}}, \frac{\|R(\widetilde{\overline{\rho}\theta})\|_{\infty,\Omega_e}}{\|\overline{\rho}\widetilde{\theta} - \overline{\rho}\widehat{\theta}\|_{\infty,\Omega}} \right). \tag{96}$$

In (96) $\widehat{\cdot}$ indicates the space average of the quantity at hand over Ω and the norms $\|\cdot\|_{\infty,\Omega}$ at the denominator are used for normalization to preserve the correct dimension of the resulting equation. Having $\mu_{\max}$ and μ_{res} constructed, the dynamic coefficients of the viscosity terms can be computed as

$$\mu_n|_{\Omega_e} = \min \|\overline{\rho}\|_{\infty,\Omega_e} \left(\mu_{\max}|_{\Omega_e}, \mu_{\text{res}}|_{\Omega_e} \right) \tag{97}$$

and

$$\kappa_n|_{\Omega_e} = \frac{\text{Pr}}{\gamma - 1} \mu_n|_{\Omega_e}, \tag{98}$$

where Pr is an artificial Prandtl number. The residuals in (96) are simply:

$$R(\widetilde{\mathbf{u}}) = \frac{\partial \overline{\rho}\widetilde{\mathbf{u}}}{\partial t} + \nabla \cdot (\overline{\rho}\widetilde{\mathbf{u}} \otimes \widetilde{\mathbf{u}}) + \nabla \overline{p} + \overline{\rho}\mathbf{g}, \tag{99a}$$

$$R(\overline{\rho}) = \frac{\partial \overline{\rho}}{\partial t} + \nabla \cdot (\overline{\rho}\widetilde{\mathbf{u}}), \tag{99b}$$

$$R(\widetilde{\overline{\rho}\theta}) = \frac{\partial \widetilde{\overline{\rho}\theta}}{\partial t} + \nabla \cdot \left(\widetilde{\overline{\rho}\theta}\widetilde{\mathbf{u}} \right). \tag{99c}$$

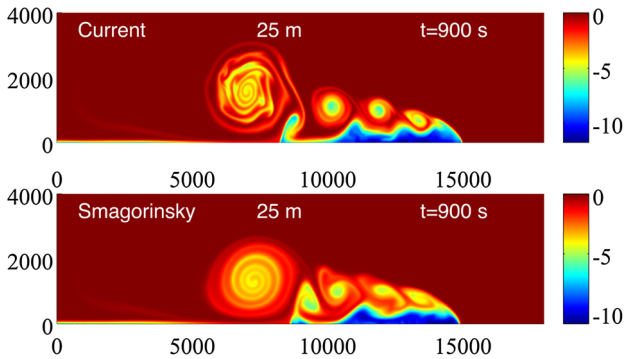

Fig. 18 Stabilized solution of the density current problem [284]. Reproduced from [216]

The time derivatives are to be included or the consistency of the method would be lost. An example of the stabilized spectral element solutions reported in [216] is plotted in Fig. 18, where also the results obtained using a constant coefficient Lilly–Smagorinsky model [201, 274] are given for comparison. Putting together the moist problem briefly described in Sect. 2.4 and the current LES-based stabilization, the simulation of a fully three-dimensional deep convection problem is reported in [215]; in Fig. 19, we reproduce Fig. 3 contained therein.

The multi-scale properties of this scheme have been verified via the simulation of a turbulent flow on the sphere whose radius is that of the earth. As an example, a turbulent flow in a geostrophically balanced atmosphere is shown in Fig. 20, after [216].

5 Vertical Discretization, Computational Grids, and Adaptive Mesh Refinement in NWP

We briefly discuss the issue of vertical discretization in atmospheric models since it is characterized by some constraints that do not apply to more traditional and general CFD models. Because of the classical use of finite differences with Cartesian rectangular grids, the accurate approximation of topography has always been a major concern both in atmospheric and ocean models. The vertical coordinate systems can be separated into two main branches: σ terrain-following [104, 237] and height-coordinates. Terrain-following coordinates have the advantage of the accurate representation of topography and ease of application of boundary conditions as the grid cells follow the shape of the varying bottom of the domain. However, the large truncation errors that increase with increasing topography slope [164, 286] require vertical coordinates that are more suitable for steep topographies. The height-coordinate system was first proposed as the η-system by [219]. It consists of the use of a rectangular grid that intersects the topography and defines the orographic height at

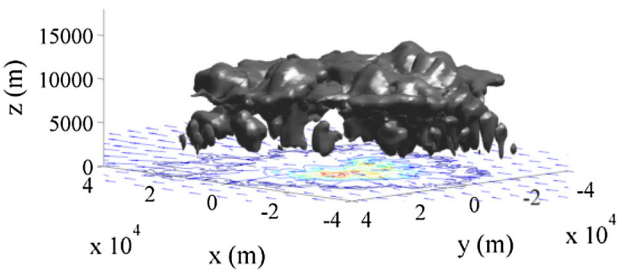

Fig. 19 Deep convection: 3D view of q_c (*grey surface*), surface velocity (*vectors*), and the instantaneous distribution of q_r on the ground (*contours*). Reproduced from Fig. 3 of [215]

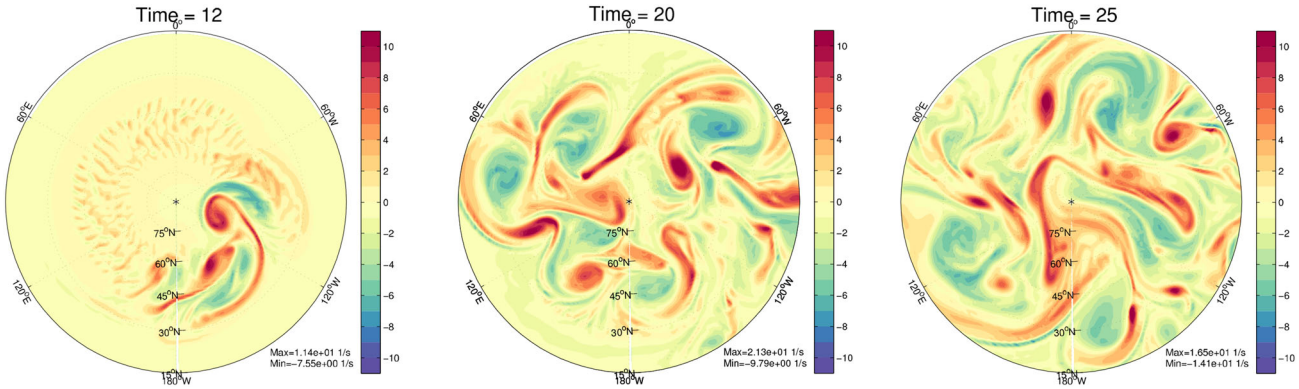

Fig. 20 Turbulent flow on the sphere after 12, 20, and 25 days. *Top-view* looking down onto the northern hemisphere. The radial component of vorticity is *plotted* and *colored* by intensity. Plot adapted from [216]

the cells edges. Modification of both approaches have been later defined. Examples are the hybrid terrain-following coordinates [268] as an improvement of σ, or the shaved-cell method in z-coordinates introduced by [1] for ocean models. Figure 21 shows a schematic of these grids.

The σ grid mentioned above is simple, but on steep topography the regularity of the grid in the inner domain is compromised. To overcome this drawback, [263] introduced the *smooth level vertical* (SLEVE) mapping that helps maintain a sufficient degree of regularity of the node distribution away from the bottom boundary. Given a mountain ridge, a SLEVE grid is obtained from the decomposition of a large and small scale variation of topography (e.g. a Gaussian terrain perturbed by a wave-like function). Through this solution the grid distortion is controlled from bottom to top by means of two free parameters. Somewhere between σ and SLEVE stands the hybrid grid of [268]. The hybrid grid uses the same vertical coordinate σ and combines the topography and the height of the domain through two functions $a(\sigma)$ and $b(\sigma)$ whose values are properly tabulated.

Finite elements and Galerkin methods in general (finite volumes included) are free of all the drawbacks of methods that are not flexible with regard to the grid. Finite elements depend on computational grids of quadrilateral and triangular elements (in 2D) or hexahedra, tetrahedra, and prisms (in 3D) that adjust to the physical geometry to be discretized without affecting the formulation of the governing equations. The grid shape is inherently defined in the numerical formulation of the method. Generally speaking, they are z-coordinate based methods with full control of the shape of the topography. The grid itself looks like a σ-grid, but the fundamental difference is that finite difference methods with σ grids require re-expressing the equations using a coordinate transformation.

Due to the geometrical flexibility of Element-Based Galerkin (EBG) methods, no coordinate transformation is needed to apply the ground boundary condition. Complex orography can be modeled with ease using finer or, perhaps, adaptive grids (see Sect. 5.2), as long as certain criteria on regularity and smoothness of the element shape are respected. Furthermore, in a time when high resolution is the rule, complex orography can be modeled with ease and better grids. High resolution terrain-following coordinates induce grids to lose the property of orthogonality at the boundaries. The internal elements as well would suffer from great stretching up to a point that the grid is no longer sufficiently smooth for the numerical method to perform correctly. For example, if the Jacobian of the transformation from physical to computational space is singular, large

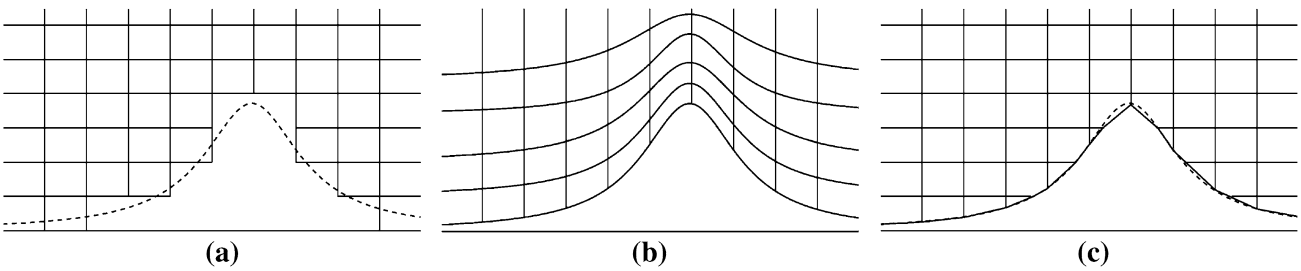

Fig. 21 Representation of a smooth mountain using: **a** height coordinate system with step topography, **b** σ-terrain following coordinates, and **c** height coordinate system with shaved cells

numerical errors and instability in the solution would occur [300]. The application of CFD grid generation techniques for use in atmospheric problems is being considered more and more. Simple and fast structured grid generation with boundary layer grids or elliptic smoothing has been described in, e.g., [208]. Unstructured grids are also becoming of interest, as shown a few years ago by [5] and, more recently, by [275]. However, the inertia from the atmospheric community towards grids that do not have a characteristic column-wise structure is still large. This is because all of the packages that involve the computation of atmospheric parameterizations (e.g., precipitation, radiation) are designed to work on such grids and would have to be adapted (i.e. re-written) to work on different grids. The reasoning behind this inertia is understandable, although steps ahead in this direction must be made now that high-resolution atmospheric modeling is approaching fast.

5.1 3D Grid Generation for Domains With Orography and Bathymetry

Volume grid generation in atmospheric models is commonly performed by a one directional simplification of *Transfinite interpolation* (TFI) [89, 122]. TFI is robust, simple, and arguably the fastest grid generation technique in use in many fields of computational mechanics, of which geophysical fluid dynamics represents a particular case. Nevertheless, generally the quality of TFI grids degenerates when the geometric features of the domain boundaries present sharp corners, quasi-vertical boundary walls, or similar characteristics. This has a direct effect on the quality of the numerical solution of the problem [217]. The problem exists regardless of the underlying numerical method of solution. In NWP, sharp mountain ridges and canyons are an example. With the ever increasing trend towards high spatial resolution that we are experiencing in numerical weather prediction today, sharp topographies are certainly an issue. In the following sections, we describe the current way of generating structured grids in topographical domains and present a few examples to underline the possible limitations. At that point, we introduce the idea behind elliptic grid generation and how grids may be improved in terms of smoothness and orthogonality properties by this simple technique. Most of the ideas presented in this appendix are found in the books by [182] and [298], and in the recent paper by [175].

5.1.1 Algebraic Grid Generation

As we have mentioned above, transfinite interpolation has a major drawback that comes from the constraint on the regularity of the boundaries. If the boundaries of the simply-connected domain are not sufficiently smooth, TFI fails to generate good grids. The sharpness of internal corners given by a possible discontinuity in the space derivative of the boundary functions, reflects into folding grids with unacceptable node overlapping. The problem of folding grids with difficult geometries is usually solved by subdividing the domain into smaller subdomains with more regular boundaries. This technique is robust but difficult to automate. In Fig. 23, although the edges do not cross, the vertical wall on the left-hand side of the hill is a challenge for the grid generator, as it can be noted by the extremely stretched elements in the region of the hill's front.

Nonetheless, because topography is usually smooth in current operational models (at horizontal resolution of 1 km or coarser, all mountain peaks are likely to be smoothed out), TFI is still the perfect and quick solution that can be properly modified for different types of vertical node distributions.

These improved methods are sufficiently good as long as the boundaries are never vertical. This is because the transformations are performed along z only. For full control of the nodes' distribution in all directions, these schemes should be incorporated into a full TFI interpolation.

5.1.2 Elliptic Grid Generation

One simple, yet efficient solution to the generation of smooth grids with sufficiently good properties is given by the solution of the Thompson–Thames–Mastin (TTM) problem [297], an elliptic system of partial differential equations. Two-dimensional elliptic grid generation was introduced for ocean circulation modeling in [259]. The penalty of elliptic equation methods is the higher cost with respect to algebraic methods. To control the point distribution with TTM some parameters must be selected by the user. To overcome the need for parameter selection, in [175] an automatic elliptic grid generation method is proposed. A similar approach is described in [180] for grids around topography. In this recent paper, the author also uses an iterative method to smooth the grid on a layer-by-layer basis, with a check on the grid spacing to avoid the overlapping of grid cells. This check is necessary because the method of [180], unlike the elliptic scheme, is not designed to respect the maximum principle.

5.1.3 Orthogonality

When it comes to high resolution simulations, with very fine LES grids, the boundary layer may be solved explicitly. In this case, boundary layer grids may be necessary for atmospheric models like they are for, e.g., industrial flows at much smaller scales. For how the atmospheric community is responding to the introduction of new grids,

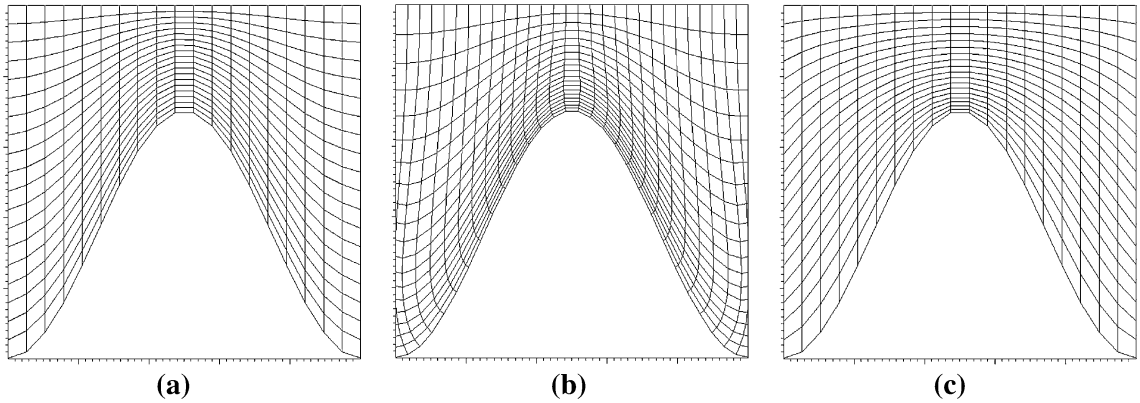

Fig. 22 **a** TFI, **b** orthogonal, **c** elliptic (not orthogonal)

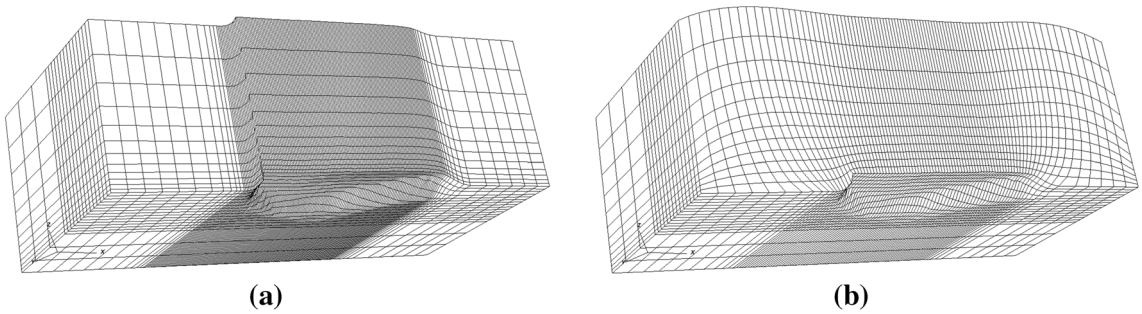

Fig. 23 **a** TFI and **b** elliptic volume grids. In this plot there is no grid control in the proximity of the boundary surface. The elliptic grid is computed with 50 iterations

orthogonal boundary grids may still be seen as futuristic. However, their use is already common in the simulation of atmospheric flows in the micro-scale (i.e. 20–500 m domains) (see, e.g., the Bolund experiment starting from [22]) so that it seems appropriate to mention them here.

Orthogonality in three-dimensional structured grid generation systems is still an active field of work (see [175, 298]). The elliptic grid generation system herein described and implemented in [221] is able to reach reasonable orthogonality properties at the lower boundary by either using Neumann boundary conditions to move the nodes, or by a proper definition of the control functions as done in [175]. Currently, a quasi-orthogonal system is the best that we can achieve with the available algorithms from the literature. Figure 22 shows how a non-orthogonal grid is transformed to a quasi-orthogonal mesh in the proximity of the boundary. This grid was deliberately relaxed to the point where the boundary layer is completely lost. This was done to clearly show orthogonality at the boundary. However, maintaining a proper stretching ratio in the proximity of the boundary without affecting orthogonality remains an open problem. A compromise is needed in building the grid, and experimentation on different topographies may be necessary.

We report a few two- and three-dimensional examples of grids generated using both algebraic and elliptic methods. Figure 22 shows the computational grid around a cosine function obtained by TFI interpolation, TFI with an orthogonal multi-surface method, and with an elliptic grid generator. The geometry is straightforward to mesh. The three methods give similar results; however, using the elliptic method together with a multi-surface technique to achieve orthogonality clearly produces a better boundary layer grid. The problem is taken a little further with the fully three-dimensional mesh of the Bolund hill in Denmark. The improvement in terms of regularity of the grid in the internal volume and in terms of quasi-orthogonality, is evident from panel (b) in Fig. 23, where the elliptic solver was applied with a few iterations to improve the algebraic grid of panel (a).

5.2 Adaptive Mesh Refinement

The term adaptive mesh refinement (AMR) describes mesh generation techniques in which the spatial resolution is adjusted depending on certain properties of the specific application. Within AMR one distinguishes between static and dynamic AMR. In static AMR the mesh is adjusted

once at the beginning of the simulation whereas dynamic AMR adapts the resolution for the whole duration of the simulation as a function of the structure of the solution based on some pre-defined criterion.

The idea to increase the resolution in part of the domain has a long history in scientific computing and also in meteorology. Usually this adjustment of the resolution is done by coupling two numerical models with different resolutions (so called nesting). The easiest way to implement nesting is to first run a full coarse simulation and then use the result of this coarse simulation as boundary conditions for a higher resolved simulation in a smaller domain. In this approach the result of the higher resolution simulation cannot affect the coarse simulation. For this reason this approach is called one-way nesting. An example for one-way nesting can be found in the 1976 work by Davies [72] and the work of Miyakoda and Rosati [220]. More difficult but also more accurate is two-way nesting in which both numerical models are allowed to interact which each other. This means that the result on the coarse mesh is not only used as an initial and boundary condition of the finer mesh but the result on the finer mesh is also used to improve the accuracy of the simulation using the coarse mesh. An example of two-way nesting is the work by Zhang et al. [329]. Nesting does not need to be static. The domain of the higher resolution simulation can move within the domain of the coarse resolution simulation like Ley and Elsberry did in 1976 [200]. Nesting is not restricted to combining two different simulations. More than two different resolutions can be combined, as in Ginis et al. [109].

An alternative to increasing the spatial resolution via nesting is to use variable mesh spacing in the different directions like in Staniforth and Mitchell [280], or to adjust the mesh with the help of a coordinate transformation, as in Dietachmeyer and Droegemeier [75].

Dynamic AMR in which the mesh is repeatedly adjusted according to the current intermediate result of the simulation has been used in engineering applications for a long time [23, 24]. The first application of this kind of dynamic AMR in atmospheric sciences was done by Skamarock et al. [272] and Skamarock and Klemp [270]. A first approach to use dynamic adaptive mesh refinement operationally was given by the OMEGA model (OMEGA stands for Operational Multiscale Environment Model with Grid Adaptivity). The OMEGA model was presented in the work of Bacon et al. [8]. Simulations of hurricane tracks by Gopalakrishnan et al. [121] demonstrate that the accuracy of the hurricane simulation can be improved significantly by using dynamic AMR while at the same time reducing the runtime of the simulation. There are however still many open questions that need to be resolved for a broader application of dynamic AMR in atmospheric sciences [315]. More details about the historical evolution of AMR can be found in [18] and [159].

Within dynamic mesh refinement there are three possible approaches to adjust the accuracy of the simulation according to the current flow:

- h-adaptive mesh refinement: the spatial resolution is adjusted by adding or removing grid points in the mesh. In an element-based method this is done by subdividing elements into smaller elements or merging elements into larger elements. The total number of elements and grid points is allowed to change in this approach. This makes it necessary to redistribute elements sometimes when multiple computing nodes are used for the computation. We discuss this approach more in detail below.
- r-adaptive mesh refinement (or moving mesh): the grid points and therefore elements are moved and deformed in such a way that the spatial resolution gets finer in those parts of the domain where the accuracy of the simulation needs to be increased. This reduces automatically the density of grid points in other parts of the domain and therefore reduces accuracy in those parts. The total number of grid points and elements is constant in this approach. An example of this approach can be found in the work of Kühnlein et al. [188], Budd and Williams [40], and Bauer et al. [16].
- p-adaptive mesh refinement: the accuracy of the simulation is adjusted by changing the polynomial order of the spatial discretization. The size and location of the elements remains unchanged in this approach. One of the first descriptions of this approach can be found in the work of Babuska et al. [7]. Application of this approach to geophysical modeling can be found recently in [303] and, earlier on in [90].

These three approaches for dynamic AMR can be combined with each other like in the work by Lang et al. [190], Pigott et al. [239], and [111] and references therein.

We concentrate in the following on h-adaptive mesh refinement more in detail. Within this approach we can distinguish between conforming AMR and non-conforming AMR. A rising warm air bubble simulation using conforming and non-conforming AMR is shown in Fig. 24.

5.3 Non-conforming Mesh Refinement

As explained in the section above, mesh refinement techniques create either conforming or non-conforming meshes. In conforming meshes each element has only one neighbor per element face (*h*-conforming), so a situation where more than two elements share the same face is not allowed. Also, the elements have to be *p*-conforming, that is the approximating polynomials in both neighboring

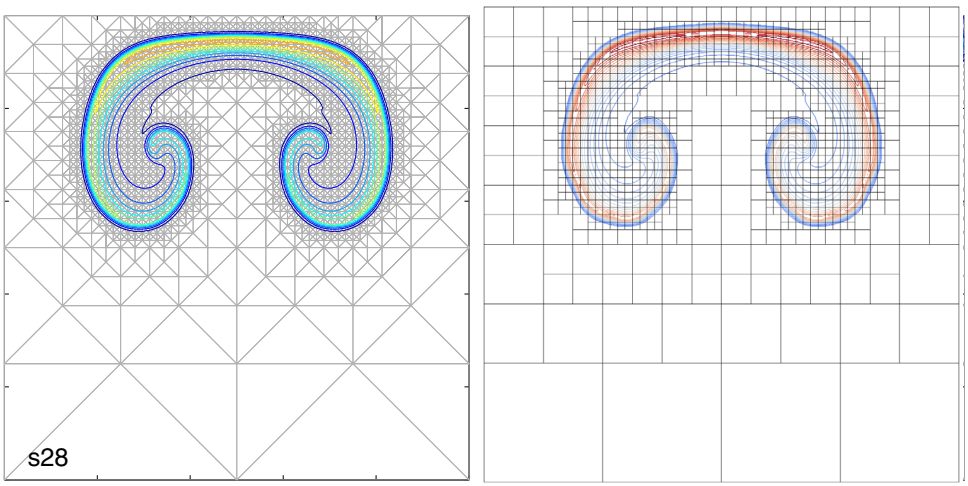

Fig. 24 Conforming (*left*) and non-conforming (*right*) AMR simulations of a rising thermal bubble. The *left figure* is adapted from [224]. The *right one* is adapted from [184]

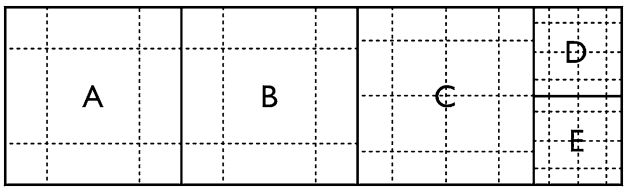

Fig. 25 Different element interfaces, where *A–B* is conforming, *B–C* *p*-non-conforming, and *C–D–E* *h*-non-conforming. *Dashed lines* symbolize the higher order mesh within the elements

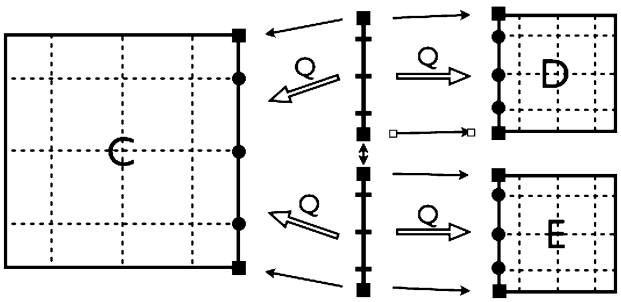

Fig. 26 Schematic of mortar element method. The mortars are binding non-conforming elements that sum the contribution from element edges and apply an L_2 projection of the mortar data back to the element edges. A *single arrow* represents direct assignment of the vertex value, while *double arrow* represents L_2 projection

elements are of the same order and the nodal points on both sides of the face coincide. An example of a *hp*-conforming element interface is shown in Fig. 25, where elements A and B exclusively share the same face (side) and have the same polynomial expansion order. In such a situation there are no additional requirements for the numerical method. The burden of creating a conforming mesh after the refinement falls entirely on the AMR algorithm.

In non-conforming meshes, however, one needs to account for faces that are shared by more than two elements (*h*-non-conforming, see interface between elements C, D, and E in Fig. 25) or with different polynomial approximation on both sides (*p*-non-conforming interface between elements B and C in Fig. 25). Mind that Fig. 25 does not illustrate all the possibilities of non-conforming configurations. One other possibility is a *hp*-non-conforming interface, where an edge (or face) is shared by more than two elements of different polynomial orders. To complete the discussion of meshes and element-based Galerkin methods, in this section we provide an overview of methods used to reconcile non-conforming elements. Note that conforming AMR grids do not require any special handling by the numerical method and so we will discuss conforming AMR no further, but rather, shall concentrate on non-conforming AMR.

5.3.1 Mortar Element Method

The first to introduce a non-conforming formulation for spectral element methods were Maday et al. [205], who presented the mortar element method (MEM), where the domain is split into blocks of conforming elements, and a new trace space, namely mortars, is introduced to couple the non-conforming blocks. The MEM was an extension of classical non-conforming methods in the finite element community [51, 81, 285] with the difference that, besides being applied to the spectral element method, it did not rely on Lagrange multipliers or master-slave relations of non-conforming edges of the elements.

In MEM, the mortars are one-dimensional constructs (in 2D; they are two-dimensional in 3D) with a polynomial space defined on them. The task of the mortar is to reconcile the C^0 continuity condition between the non-conforming elements that the mortar is connecting. In other words, the mortar is an interface between the non-conforming element faces (see Fig. 26). The polynomial order

on the mortar is typically chosen to match the highest order expansion among the elements contributing to the mortar. The end-point values of the mortar solutions are constrained to match the values at corresponding vertices of the original elements (represented by single arrows in Fig. 26). The integral projection operation (L_2 projection) is defined to project the solution from the mortar to the interior points of the non-conforming element edges (double arrows in Fig. 26). If we write this operation in matrix form as $Q_{n\times m}$, where n is the number of nodes on the element edge, and m is the number of nodal points on the mortar, then the operation Q^T will sum the contributions from the element edges to the mortar. To reconcile the C^0 continuity condition we first sum the contributions from element edges on the mortar (Q^T), perform weighted averaging, and project the result back to the element edges (Q). Mind that this method only minimizes the discontinuity but does not enforce a strict C^0 continuity. Even though here we present only a limited spectrum of possible non-conforming configurations, MEM is very general and can be applied in more complicated situations [205].

5.3.2 Pointwise-Matching Method

Another method, stemming from the finite element community, is the pointwise matching method (PMM), or the interpolation-based method [47, 93, 254]. In this approach both h and p-non-conforming elements are allowed, however it is assumed that for h-non-conforming elements there is one parent edge on one side of the interface, and two children edges on the other side. In the case of the p-non-conforming, the parent edge is the one with lower polynomial order.

In Fig. 27 the parent edge belongs to element C and the parent points are marked with filled circles. The values at the child points are interpolated from the parent points. Here Q denotes this interpolation. To ensure strict C^0 continuity, the solution from the child points is first added to the parent points via the operation Q^T. The solution is averaged at the parent edge and interpolated onto the child edge via Q. We mark the corner points of children elements with a filled circle, as the interpolation between those points and corresponding points at the parent edge is trivial. Unlike MEM, the continuity here is strictly enforced.

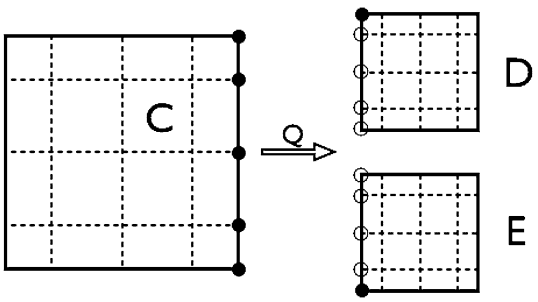

Fig. 27 Schematic of the pointwise-matching method. Parent points are marked with *filled circles*. Values at children points depend on the values of the parent points. Here operation Q marks the interpolation from the parent element to children elements

It is possible to express PMM using mortars and hence bring those two methods into one framework. This can be achieved by using the mortar infrastructure and replacing the L_2 projection by interpolation. In such an approach, the choice of Q (projection or interpolation) will define the method. Traditionally, in MEM we choose the size of mortars to correspond to short (children) edges, while in PMM we use the parent edge as a mortar analogue. [265] investigates how different choices of the size of mortars affects the performance of both PMM and MEM.

5.3.3 Mortar Elements for DG and Application to Atmospheric Simulations

The early work on MEM focused mainly on elliptic problems and spectral element methods [25, 205]. Kopriva [186] applied the MEM to compressible flows and the DG method by imposing an additional condition on the global conservation of the mortar approximation, as well as *outflow conditions*. The outflow condition required that the solution from the "upwind side" of the mortar after projection onto the mortar and back to the face remains unchanged. This work was used later for atmospheric simulations in [33, 184]. Based on the two-dimensional work reported in [184], Marras et al. [211] used MEM in a unified CG/DG shallow water model on the sphere (with static and dynamic adaptivity). An application of the pointwise-matching scheme to geophysical simulations can be found in [258]. In [185], Kopera and Giraldo presented a unified framwework including both CG and DG methods, as well as integral projection (for DG) and pointwise-matching (for CG) schemes for non-conforming interfaces and found that similar mass conservation properties can be obtained for both configurations.

5.3.4 Unified CG/DG Non-conforming Method

The idea of a unified CG/DG method stems from the similarity of both approaches. Much of the mathematical operations, and therefore much of the code implementation is the same, with an exception of communication between elements. A time-step of a unified CG/DG method can be described by the following algorithm.

1. Evaluate volume integrals for each element and store as right-hand-side *RHS*.
2. Perform inter-element communication:

For CG, perform Direct Stiffness Summation on *RHS* to ensure C^0 continuity,
For DG, evaluate element-boundary integrals (fluxes) and update *RHS*.

3. Divide by the mass matrix. Notice that the mass matrix for CG corresponds to the assembled DG mass matrix.
4. Perform the time-step $\frac{\partial q}{\partial t} = RHS$.

This recipe can be applied regardless of whether one constructs a conforming or non-conforming method. The non-conforming element treatment will affect only the inter-element communication step. Even though this step is different between the methods, it is desirable to construct both non-conforming edge algorithms in a similar fashion. As discussed in previous sections, using the mortar element method one can incorporate the integral projection method by [186] as well as the pointwise-matching method into the same framework. The implementation of a unified CG/DG method with non-conforming interfaces is described in detail in [184, 185]. Here we outline the general approach to both CG (using pointwise-matching method) and DG (using integral projection method) treatment of non-conforming interfaces.

Figure 28 shows a schematic of the unified approach to inter-element communication of CG and DG methods for conforming and non-conforming interfaces. Panel (a) shows the interpolation-based mortar point wise matching method used for CG, while panel (b) presents the integral projection method for DG. In both situations the solution from non-conforming edges is first communicated onto mortars, then an appropriate action is performed on the conforming mortars and the result is communicated back to the element edges. The first difference between the two approaches is the choice of the mortar. For the CG method we choose the mortar to be conforming with the longer parent edge (so called *long rule* [265]), while for the DG method we choose the shorter, children edges to define the size of the mortar (*short rule*). Additionally, for the CG method we need to define an additional, point mortar to ensure the communication between vertex neighboring elements.

The second difference between the two methods is the choice of non-conforming communication matrices. For CG we perform an interpolation with the matrix J_k:

$$J_{k,ij} = h_i(\xi^k(z_j)), \quad k = 1, 2,$$

where h_i are the basis functions defined on the mortar, ξ^k is a map from the element edge coordinate z to the mortar coordinate ξ, and z_j is the coordinate of the jth nodal point at the element edge. The interpolation matrices J_k will

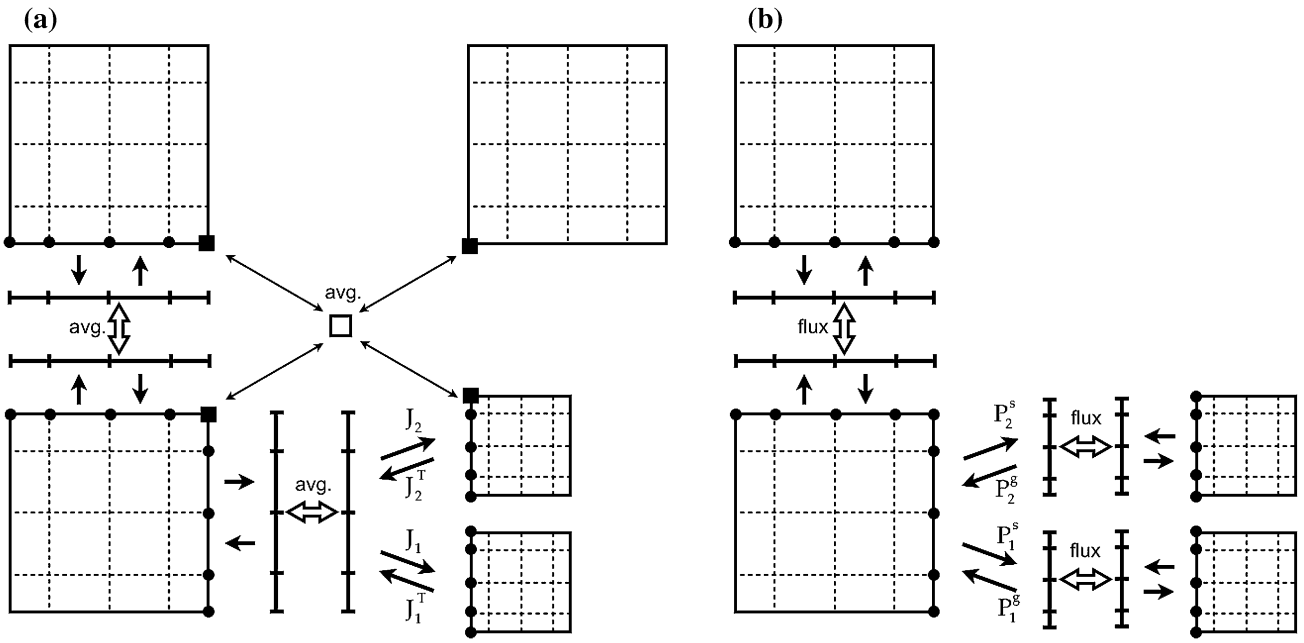

Fig. 28 Schematic of the mortar based non-conforming methods for both CG (**a**) and DG (**b**). Both methods follow the same algorithm of communicating data to the mortar, performing operations on the mortar and communicating the data back. The differences lie in the choice of mortars (mortar conforming with the long edge plus an additional point mortar to communicate between vertex neighbors for CG, mortars conforming with short edges for DG) and matrices used to communicate data to and from the non-conforming mortar. In the case of CG, the data is interpolated using matrices $J^T_{1,2}$ and $J_{1,2}$; for DG we use projection matrices $P^s_{1,2}$ and $P^g_{1,2}$

scatter the solution from the mortar to two children element edges. The opposite action, gather, is performed by the transpose matrices J_k^T. The matrices J_k correspond to the operation Q showed in Fig. 27.

For the DG method the communication from parent element (long edge) to mortars requires projection matrices, defined in [186] by imposing the integral condition on a non-conforming interface

$$\int_{-1}^{1} (q_M^k(\xi) - q(\xi^k(z)))\psi(\xi)d\xi = 0, \quad k = 1, 2$$

where $q_M^k(\xi)$ is the solution projected at the kth mortar, $q(\xi^k(z))$ is the solution at the parent edge, ξ is the coordinate defined on the mortar, z is the coordinate of the parent edge and $\xi^k(z)$ is the map between the parent edge and kth mortar coordinates. This integral condition can be expressed in matrix form as:

$$Mq_M^k - S^k q = 0,$$

where $M_{ij} = \int_{-1}^{1} \psi_i(\xi)\psi_j(\xi)d\xi$ and $S_{ij}^k = \int_{-1}^{1} \psi_i(\xi)\psi_j(\xi^k(z))$. The projection operation is defined as follows

$$q_M^k = M^{-1}S^k q = P_k^s q.$$

P_k^s is the scatter projection matrix used to communicate the solution from parent edge to a mortar. The communication in the opposite direction is achieved by a gather projection matrix

$$q = \sum_{k=1}^{2} P_k^g q_M^k,$$

where $P_k^g = \frac{1}{2}M^{-1}(S^k)^T$. The details are presented in [184, 186].

Finally, the third difference between inter-element communication of CG and DG methods lies in the action performed on the mortar. Once the solutions from element edges are interpolated/projected onto mortars, the CG method averages the solution before scattering it back to respective edges, while in the DG method we need to compute a flux, which is then projected back to the elements. This action is identical between conforming and non-conforming edges, as mortars are by design conforming to each other. Note that whenever the mortar and element edge spaces are conforming, the communication matrix (regardless of whether it is interpolation or projection) is always the identity matrix. It is also important to note that while Fig. 28 presents schematics for both methods, the actual implementation in the code may differ. Details of CG interpolation method implementation is presented in [185], along with a unified implementation of the CG/DG method.

5.3.5 *Performance of the Non-conforming Adaptive Mesh Refinement*

Adaptive mesh refinement and element-based Galerkin methods are a natural match due to the element-local nature of most computationally intensive calculations. The edge-based computations can also be implemented efficiently even for non-conforming edges. Figure 29 presents the plot of speed-up achieved by using the AMR algorithm against the ratio of the element count in a reference simulation to the average element count of an AMR simulation. If there were no AMR overhead, the simulation with element ratio of 10 would yield a speed-up of 10, which is represented by the solid black line. The study was performed for three different time integration methods, namely the explicit Runge–Kutta 3rd order 5-stage method, and two implicit–explicit second order methods BDF2 and ARK2. The explicit method virtually overlaps with the ideal speed-up line, while the IMEX methods show more AMR overhead. The sources of this decreased speed-up are studied in [184]. This result shows, however, that the adaptive mesh algorithm can be implemented with minimal overhead.

Table 3 shows the breakdown of time spent in different parts of the code that was run on a single CPU with detailed timings for different parts of the AMR algorithm. The biggest share of time spent in adapting the mesh falls on the criterion evaluation, even for the very simple threshold criterion used in this study. The time spent on evaluating fluxes on non-conforming interfaces is in the range of 2–3 % for all the methods, however this cannot be considered an overhead. A single non-conforming interface

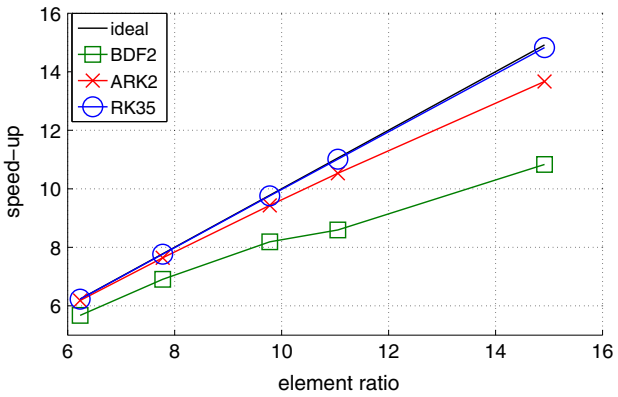

Fig. 29 Performance of the adaptive mesh algorithm with the DG method. Speed-up is defined as the ratio of runtime of a reference simulation with a fully refined mesh to the runtime of the adaptive simulation. The element ratio measures the ratio of element count of the fully refined simulation to the average element count of the AMR simulation. *Black solid line* represents ideal speed-up. Results are reported for an explicit Runge–Kutta time integration (RK35) and two implicit–explicit schemes (BDF2 and ARK2). The simulation was set-up as a standard density current test case. *Source*: [184]

Table 3 Timings (in seconds of runtime) and percentage breakdown of different AMR components for the density current case

	RK35	BDF2	ARK2
Total	5292 (100 %)	643.6 (100 %)	967.8 (100 %)
Volume integrals	2876.1 (54.3 %)	274.46 (42.6 %)	362.9 (37.5 %)
Face integrals	507.67 (9.59 %)	66.59 (10.3 %)	80.1 (8.29 %)
Non-conforming faces	116.79 (2.21 %)	21.33 (3.31 %)	23.65 (2.44 %)
AMR:	6.030 (0.11 %)	6.291 (0.98 %)	6.025 (0.62 %)
Criterion evaluation	3.463	3.517	3.527
Mesh manipulation	0.051	0.05	0.051
Data projection	0.965	1.247	0.965
Other	1.412	1.450	1.454

Source: [184]

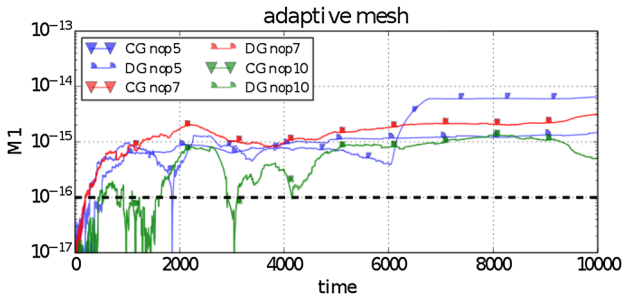

Fig. 30 Mass conservation as a function of simulation time for a dynamically adaptive mesh. *Triangular markers* denote the CG results and *circular markers* represent the DG results. *Colors* represent different polynomial orders

replaces two regular faces, therefore the time spent in computing those fluxes should be compared with the time the code would spent on computing the fluxes on the faces that the non-conforming interface replaces. Additionally, due to the non-conforming faces we are able to reduce the number of elements in the domain and therefore reduce the time spent on the volume integration. The important message is that the computation of fluxes on non-conforming interfaces has a minor cost compared to other parts of the code.

Finally, the element-based Galerkin methods preserve their original mass conservation properties even when used with non-conforming meshes. This is discussed in detail in [185] and presented Fig. 30. Both CG and DG methods conserve mass to machine precision even for very long simulation times. In the density current case presented in Fig. 30 the typical simulation time is 900 s, while the test was performed until 10,000 s.

6 Summary

After briefly reviewing the different mathematical models of the atmosphere (i.e. equation sets), in this article we have concentrated on their numerical solution by means of element-based Galerkin methods (EBG). The emphasis was given to the finite and spectral element methods on the one hand and nodal discontinuous Galerkin on the other. Element-based Galerkin methods appear to be perfectly suited for building next-generation climate and numerical weather prediction (NWP) models because of their geometric flexibility (unstructured and AMR) and high parallel-scalability (on both CPU and GPU-based architectures; see, e.g. [231] and citations therein). There are issues that remain essentially unsolved (e.g., stabilization) but, as we have shown in this manuscript, there are numerous options that appear to satisfy the requirements for building robust, efficient, and positivity-preserving numerical models. Finally, because atmospheric modeling is aiming at higher and higher resolution simulations and, hence, well resolved topography, we discussed the issue of grid generation and gave some insight on its evolution in NWP.

Acknowledgments The authors would like to thank Prof. Eugenio Oñate (U. Politècnica de Catalunya) for his invitation to submit this review article. They are also thankful to Prof. Dale Durran (U. Washington), Dr. Tommaso Benacchio (Met Office), and Dr. Matias Avila (BSC-CNS) for their comments and corrections, as well as insightful discussion with Sam Watson, Consulting Software Engineer (Exa Corp.). Most of the contribution to this article by the first author stems from his Ph.D. thesis carried out at the Barcelona Supercomputing Center (BSC-CNS) and Universitat Politècnica de Catalunya, Spain, supported by a BSC-CNS student grant, by Iberdrola Enrgías Renovables, and by Grant N62909-09-1-4083 of the Office of Naval Research Global. At N.P.S., S.M., A.M., M.K., and F.X.G. were supported by the Office of Naval Research through program element PE-0602435N, the Air Force Office of Scientific Research through the Computational Mathematics program, and the National Science Foundation (Division of Mathematical Sciences) through program element 121670. The scalability studies of the atmospheric model NUMA that are presented in this paper used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357. S.M., M.K., and A.M. are grateful to the National Research Council of the National Academies.

References

1. Adcroft A, Hill C, Marshall J (1997) Representation of topography by shaved cells in a height coordinate ocean model. Mon Weather Rev 125:2293–2315
2. Ahmad N, Lindeman J (2007) Euler solutions using flux-based wave decomposition. Int J Numer Methods Fluids 54:47–72
3. Arakawa A, Konor CS (2009) Unification of the anelastic and quasi-hydrostatic systems of equations. Mon Weather Rev 137:710–726
4. Argyris JH, Kelsey S (1960) Energy theorems and structural analysis. Butterworths, London. Reprented from a series of article in Aircraft Eng 19, 742
5. Aubry R, Vázquez M, Houzeaux G, Cela JM, Marras S (2010) An unstructured CFD approach to numerical weather prediction. In: Proceedings: 48th AIAA aerospace sciences meeting, 4–7 January 2010, Orlando, Florida. AIAA paper, pp 691–783
6. Avila M, Codina R, Principe J (2014) Large eddy simulation of low mach number flows using dynamic and orthogonal subgrid scales. Comput Fluids 99:44–66
7. Babuska I, Szabo BA, Katz IN (1981) The p-version of the finite element method. SIAM J Numer Anal 18(3):515–545
8. Bacon D, Ahmad N, Boybeyi Z, Dunn T, Hall M, Lee C, Sarma R, Turner M (2000) A dynamically adaptive weather and dispersion model: the operational multiscale environment model with grid adaptivity (OMEGA). Mon Weather Rev 128:2044–2075
9. Baiocchi C, Brezzi F, Franca L (1993) Virtual bubbles and the Galerkin/least-squares type methods (Ga.L.S.). Comput Methods Appl Mech Eng 105:121–141
10. Bannon P (1996) On the anelastic approximation for a compressible atmosphere. J Atmos Sci 53:3618–3628
11. Baruzzi G, Habashi W, Hefez N (1992) A second order accurate finite element method for the solutions of the Euler and Navier-Stokes equations. In: Proceedings of the 13th international conference on numerical methods in fluid dynamics, Rome. Springer, pp 509–513
12. Bassi F, Rebay S (1997) A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier-stokes equations. J Comput Phys 131:267–279. doi:10.1006/jcph.1996.5572
13. Bassi F, Rebay S (1997) High-order accurate discontinuous finite element solution of the 2D Euler equations. J Comput Phys 138:251–285. doi:10.1006/jcph.1997.5454
14. Bastos J, Sadowski N (2003) Electromagnetic modeling by finite element methods, 1st edn. CRC, Boca Raton
15. Batchelor G (1953) The condition for dynamical similarities of motions of a frictionless perfect-gas atmosphere. Q J R Meteorol Soc 79:224–235
16. Bauer W, Baumann M, Scheck L, Gassmann A, Heuveline V, Jones S (2014) Simulation of tropical-cyclone-like vortices in shallow-water ICON-hex using goal-oriented r-adaptivity. Theor Comput Fluid Dyn 28(1):107–128
17. Bazilevs Y, Calo V, Cottrell JA, Hughes TJR, Reali A, Scovazzi G (2007) Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. Comput Methods Appl Mech Eng 197:173–201
18. Behrens J (2006) Adaptive atmospheric modeling. Key techniques in grid generation, data structures, and numerical operations with applications. Springer, Berlin
19. Beland M, Coté J, Staniforth A (1983) The accuracy of a finite-element vertical discretization scheme for primitive equation models: comparison with a finite-difference scheme. Mon Weather Rev 111:2298–2318
20. Benacchio T, O'Neill WP, Klein R (2014) A blended soundproof-to-compressible numerical model for small to meso-scale atmospheric dynamics. Mon Weather Rev 142:4416–4438
21. Benoit R, Desgagne M, Pellerin P, Pellerin S, Chartier Y, Desjardins S (1997) The Canadian MC2: a semi-Lagrangian, semi-implicit wideband atmospheric model suited for finescale process studies and simulation. Mon Weather Rev 125:2382–2415
22. Berg J, Mann J, Bechmann A, Courtney M, Jørgensen HE (2011) The bolund experiment, part i: flow over a steep, three-dimensional hill. Bound Layer Meteorol 141:219–243
23. Berger MJ, Colella P (1989) Local adaptive mesh refinement for shock hydrodynamics. J Comput Phys 82(1):64–84
24. Berger MJ, Oliger J (1984) Adaptive mesh refinement for hyperbolic partial differential equations. J Comput Phys 53(3):484–512
25. Bernardi C, Maday Y, Patera AT (1993) Domain decomposition by the mortar element method. In: Asymptotic and numerical methods for partial differential equations with critical parameters. Springer, pp 269–286
26. Bey K, Oden JT (1991) A Runge–Kutta discontinuous finite element method for high speed flows. In: AIAA computational fluid dynamics conference, 10th, Honolulu, HI, pp 541–555
27. Bolton D (1980) The computation of equivalent potential temperature. Mon Weather Rev 108:1046–1053
28. Boman EG, Catalyurek UV, Chevalier C, Devine KD (2012) The Zoltan and Isorropia parallel toolkits for combinatorial scientific computing: partitioning, ordering, and coloring. Sci Program 20(2):129–150
29. Bonaventura L (2000) A semi-implicit, semi-Lagrangian scheme using the height coordinate for a nonhydrostatic and fully elastic model of atmospheric flows. J Comput Phys 158:186–213
30. Botta N, Klein R, Langenberg S, Lutzenkirchen S (2004) Well balanced finite volume methods for nearly hydrostatic flows. J Comput Phys 196:539–565
31. Boyd JP (1996) The erfc-log filter and the asymptotics of the Euler and Vandeven sequence accelerations.In: Ilin AV, Scott LR (eds) Proceedings of the third international conference on spectral and high order methods, Houston Journal of Mathematics, pp 267–276
32. Boyd JP (1998) Two comments on filtering for Chebyshev and Legendre spectral and spectral element methods. J Comput Phys 143:283–288
33. Brdar S (2012) A higher order locally adaptive discontinuous Galerkin approach for atmospheric simulations. Ph.D. thesis, Universitätsbibliothek Freiburg
34. Brdar S, Baldauf M, Dedner A, Klöfkorn R (2012) Comparison of dynamical cores for NWP models: comparison of COSMO and DUNE. Theor Comput Fluid Dyn 27:453–472
35. Brezzi F, Bristeau M, Franca L, Mallet M, Rogé G (1992) A relationship between stabilized finite element methods and the Galerkin method with bubble functions. Comput Methods Appl Mech Eng 96:117–129
36. Brezzi F, Franca L, Hughes TJR, Russo A (1996) Stabilization techniques and subgrid scales capturing. Tech. Rep. http://ccm.ucdenver.edu/reports/rep083.pdf
37. Brezzi F, Franca LP, Hughes TR (1997) $b = \int g$. Comput Methods Appl Mech Eng 145:329–339
38. Brooks AN, Hughes TJR (1982) Streamline upwind/Petrov–Galerkin formulations for convective dominated flows with particular emphasis on the incompressible navier-stokes equations. Comput Methods Appl Mech Eng 32:199–259
39. Bryan GH, Morrison H (2011) Sensitivity of a simulated squall line to horizontal resolution and parameterization of microphysics. Mon Weather Rev 140:202–225
40. Budd CJ, Williams JF (2009) Moving mesh generation using the parabolic Monge–Ampére equation. SIAM J Sci Comput 31:3438–3465

41. Burridge D, Steppeler J, Struffing R (1986) Finite element schemes for the vertical discretization of the ecmwf forecast model using linear elements. Tech. Rep. 54, ECMWF, Sheffild Park, Reading, UK
42. Burstedde C, Wilcox LC, Ghattas O (2011) p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. SIAM J Sci Comput 33(3):1103–1133
43. Canuto C (1994) Stabilization of spectral methods by finite element bubble functions. Comput Methods Appl Mech Eng 116:13–26
44. Canuto C, Puppo G (1994) Bubble stabilization of spectral Legendre methods for the advection–diffusion equation. Comput Methods Appl Mech Eng 118:239–263
45. Canuto C, Russo A, Van Kemenade V (1998) Stabilized spectral methods for the Navier–Stokes equations: residual-free bubbles and preconditioning. Comput Methods Appl Mech Eng 166:65–83
46. Canuto C, Van Kemenade V (1996) Bubble-stabilized spectral methods for the incomplressible Navier–Stokes equations. Comput Methods Appl Mech Eng 135:35–61
47. Chang RY, Hsu CH (1996) A variable-order spectral element method for incompressible viscous flow simulation. Int J Numer Methods Eng 39(17):2865–2887
48. Chavent G, Salzano G (1982) A finite-element method for the 1-d water flooding problem with gravity. J Comput Phys 45(3):307–344
49. Chevalier C, Pellegrini F (2008) Pt-scotch: a tool for efficient parallel graph ordering. Parallel Comput 34(6):318–331
50. Christie I, Griffiths D, Mitchell A, Zienkiewicz OC (1976) Finite element methods for second order differential equations with significant first derivatives. Int J Numer Methods Eng 10:1389–1396
51. Ciarlet PG (1978) The finite element method for elliptic problems. Elsevier, Amsterdam
52. Cockburn B, Karniadakis GE, Shu CW (2000) The development of discontinuous Galerkin methods. UMSI research report/University of Minnesota (Minneapolis, MN). Supercomputer Institute, vol 99, p 220
53. Cockburn B, Shu CW (1991) The Runge–Kutta local projection p1-discontinuous Galerkin finite element method for scalar conservation laws. Rairo-Math Model Numer 25(3):337–361
54. Cockburn B, Shu CW (1998) The local discontinuous Galerkin method for time-dependent convection–diffusion systems. SIAM J Numer Anal 35(6):2440–2463
55. Codina R (1993) A discontinuity-capturing crosswind-dissipation for the finite element solution of the convection–diffusion equation. Comput Methods Appl Mech Eng 110:325–342
56. Codina R (2000) Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. Comput Methods Appl Mech Eng 190:1579–1599
57. Codina R (2002) Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. Comput Methods Appl Mech Eng 191:4295–4321
58. Codina R, Blasco J (2002) Analysis of a stabilized finite element approximation of the transient convection–diffusion–reaction equation using orthogonal subscales. Comput Vis Sci 4:167–174
59. Codina R, Oñate E, Cervera M (1992) The intrinsic time for the streamline upwind/Petrov–Galerkin formulation using quadratic elements. Comput Methods Appl Mech Eng 94:239–262
60. Collis SS, Chang Y (2002) The DG/VMS method for unified turbulence simulation. AIAA Pap 3124:24–27
61. Corsini A, Rispoli F, Santoriello A (2005) A variational multiscale higher-order finite element formulation for turbomachinery flow computations. Comput Methods Appl Mech Eng 194(45–47):4797–4823
62. COSMO P (1998) Consortium for small-scale modeling. COSMO project. www.cosmo-model.org/content/model/documentation
63. Coté J (1988) A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere. Q J R Meteorol Soc 114(483):1347–1352
64. Coté J, Desmarais J, Gravel S, Methot A, Patoine A, Roch M, Staniforth A (1998) The operational CMC-MRB global environmental multiscale GEM model. Part II: results. Mon Weather Rev 126:1397–1418
65. Coté J, Gravel S, Methot A, Patoine A, Roch M, Staniforth A (1998) The operational CMC-MRB global environmental multiscale (gem) model. Part i: design considerations and formulation. Mon Weather Rev 126:1373–1395
66. Courant R (1943) Variational methods for the solution of problems of equilibrium and vibrat. Bull Am Math Soc 49:1–23
67. Courant R, Friedrichs K, Lewy H (1928) Uber die partiellen differenzengleichungen der mathematischen physik. Math Ann 100:32–74
68. Courtier P, Freydier C, Geleyn J, Rabier F, Rochas M (1991) The arpege project at meteo-france. In: ECMWF workshop on numerical methods in atmospheric modelling vol II, 2, pp 193–231
69. Cullen M (1973) A simple finite element method for meteorological problems. J Inst Math Appl 11:15–31
70. Cullen M (1974) A finite element method for a non-linear initial value problem. IMA J Appl Math 13:233–247
71. Cullen MJP (1990) A test of a semi-implicit integration technique for a fully compressible nonhydrostatic model. Q J R Meteorol Soc 116:1253–1258
72. Davies HC (1976) A lateral boundary formulation for multilevel prediction models. Q J R Meteorol Soc 102:405–418
73. Dennis J, Vertenstein M, Worley P, Mirin A, Craig A, Jacob R, Mickelson S (2012) Computational performance of ultra-high-resolution capability in the community earth system model. Int J High Perf Comput Appl 26:43–53
74. Dennis JM, Edwards J, Evans KJ, Guba O, Lauritzen PH, Mirin AA, St-Cyr A, Taylor MA, Worley PH (2012) CAM-SE: a scalable spectral element dynamical core for the community atmosphere model. Int J High Perf Comput Appl 26
75. Dietachmayer GS, Droegemeier KK (1992) Application of continuous dynamic grid adaptation techniques to meteorological modeling. Part 1: basic formulation and accuracy. Mon Weather Rev 120(8):1675–1706
76. Dolejši V, Feistauer M (2004) A semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow. J Comput Phys 198:727–746
77. Doms G, Schattler U (2002) A description of the nonhydrostatic regional model LM. Part I: dynamics and numerics. Consortium for small-scale modelling (COSMO) LM F90 2.18. Tech. rep., DWD, Germany, www.cosmo-model.org
78. Donea J (1984) A Taylor–Galerkin method for convection transport problems. Int J Numer Methods Eng 20:101–119
79. Donea J, Huerta A (2003) Finite element methods for flow problems, 1st edn. Wiley, New York
80. Dongarra JJ, Luszczek P, Petitet A (2003) The LINPACK benchmark: past, present, and future. concurrency and computation: practice and experience. Concurrency and computation: practice and experience 15, 2003
81. Dorr MR (1988) Domain decomposition via Lagrange multipliers. UCRL-98532, Lawrence Livermore National Laboratory, Livermore, CA
82. Douglas J, Wang J (1989) An absolutely stabilized finite element method. Math Comput 52:495–508

83. Dudhia J (1993) A nonhydrostatic version of the penn state-ncar mesoscale model: validation tests and simulation of the atlantic cyclone and cold front. Mon Weather Rev 121:1493–1513
84. Durran D (1989) Improving the anelastic approximation. J Atmos Sci 46:1453–1461
85. Durran D (1998) Numerical methods for wave equations in geophysical fluid dynamics, 1st edn. Springer, Berlin
86. Durran D (2008) A physically motivated approach for filtering acoustic waves from the equations governing compressible stratified flow. J Fluid Mech 601:365–379
87. Durran D, Blossey P (2012) Implicit-explicit multistep methods for fast-wave-slow-wave problems. Mon Weather Rev 140:1307–1325
88. Emanuel KA (1994) Atmospheric convection. Oxford University Press, Oxford
89. Eriksson LE (1982) Generation of boundary conforming grids around wing-body configurations using transfinite interpolation. AIAA J. 20:1313–1320
90. Eskilsson C, Sherwin SJ (2004) A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations. Int J Numer Methods Fluids 45:605–623
91. Farhat C, Rajasekharan A, Koobus B (2006) A dynamic variational multiscale method for large eddy simulations on unstructured meshes. Comput Methods Appl Mech Eng 195(13–16):1667–1691. doi:10.1016/j.cma.2005.05.045. http://www.sciencedirect.com/science/article/pii/S0045782505003014. A Tribute to Thomas J.R. Hughes on the Occasion of his 60th Birthday
92. Favre A (1983) Turbulence: space–time statistical properties and behavior in supers onic flows. Phys Fluids 26:2851–2863
93. Fischer PF, Kruse GW, Loth F (2002) Spectral element methods for transitional flows in complex geometries. J Sci Comput 17(1–4):81–98
94. Fischer PF, Mullen JS (2001) Filter-based stabilization of spectral element methods. C R Acad Sci Ser I Math 332:265–270
95. Fletcher C (1987) Computational techniques for fluid dynamics—vol I: fundamentals and general techniques, 1st edn. Springer, Berlin
96. Fornberg B (1998) A practical guide to pseudospectral methods, vol 1. Cambridge University Press, Cambridge
97. Fortin M, Fortin A (1989) A new approach for the fem simulation of viscoelastic flows. J Non-Newton Fluid Mech 32(3):295–310
98. Fournier A, Taylor MJ, Tribbia JJ (2004) The spectral element atmospheric model (SEAM): high-resolution parallel computation and localized resolution of regional dynamics. Mon Weather Rev 132:726–748
99. Fraedrich K, Kirk E, Luksch U, Lunkeit F (2005) The portable university model of the atmosphere (PUMA): Storm track dynamics and low frequency variabilit. Meteorol Z 14:735–745
100. Franca L, Frey S, Hughes T (1992) Stabilized finite element methods. I: application to the advective–diffusive model. Comput Methods Appl Mech Eng 95(2):253–276
101. Francis P (1972) The possible use of Laguerre polynomials for representing the vertical structure of numerical models of the atmosphere. Q J R Meteorol Soc 98:662–667
102. Fries TP, Matthies HG (2004) A review of Petrov-Galerkin stabilization approaches and an extension to meshfree methods. Tech. Rep. 1, Institute of Scientific Computing. Technical University Braunschweig
103. Gaberšek S, Giraldo FX, Doyle J (2012) Dry and moist idealized experiments with a two-dimensional spectral element model. Mon Weather Rev 140:3163–3182
104. Gal-Chen T, Somerville R (1975) On the use of a coordinate transformation for the solution of the Navier–Stokes equations. J Comput Phys 17:209–228
105. Galerkin BG (1915) Series solution of some problems of elastic equilibrium of rods and plates. Vestn Inzh Tekh 19:897–908
106. Gassmann A (2005) An improved two-time-level split-explicit integration scheme for non-hydrostatic compressible models. Meteorol Atmos Phys 88:23–38
107. Gassmann A, Herzog HJ (2008) Towards a consistent numerical compressible non-hydrostatic model using generalized hamiltonian tools. Q J R Meteorol Soc 134:1597–1613
108. Geurts BJ (2004) Elements of direct and large eddy simulation. Edwards, Philadelphia
109. Ginis I, Richardson RA, Rothstein LM (1998) Design of a multiply nested primitive equation ocean model. Mon Weather Rev 126(4):1054–1079
110. Giraldo FX (1997) Lagrange–Galerkin methods on spherical geodesic grids. J Comput Phys 136:197–213
111. Giraldo FX (2000) The Lagrange–Galerkin method for the two-dimensional shallow water equations on adaptive grids. Int J Numer Methods Fluids 33(6):789–832
112. Giraldo FX (2001) A spectral element shallow water model on spherical geodesic grids. Int J Numer Methods Fluids 35:869–901
113. Giraldo FX (2001) A spectral element shallow water model on spherical geodesic grids. Int J Numer Methods Fluids 35:869–901
114. Giraldo FX (2005) Semi-implicit time-integrators for a scalable spectral element atmospheric model. Q J R Meteorol Soc 131:2431–2454
115. Giraldo FX (2015) Element-based Galerkin methods on tensor-product bases. In: Lecture notes, pp 430
116. Giraldo FX, Hesthaven JS, Warburton T (2002) Nodal high-order discontinuous Galerkin methods for spherical shallow water equations. J Comput Phys 181:499–525
117. Giraldo FX, Kelly JF, Constantinescu EM (2013) Implicit–explicit formulations for a 3D nonhydrostatic unified model of the atmosphere (NUMA). SIAM J Sci Comput 35:1162–1194
118. Giraldo FX, Restelli M (2008) A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: equation sets and test cases. J Comput Phys 227:3849–3877
119. Giraldo FX, Rosmond T (2004) A scalable spectral element eulerian atmospheric model (see-am) for numerical weather prediction: dynamical core tests. Mon Weather Rev 132:133–153
120. Gjesdal T, Wasberg CE, Reif BAP, Andreassen Ø (2009) Variational multiscale turbulence modelling in a high order spectral element method. J Comput Phys 228:7333–7356
121. Gopalakrishnan SG, Bacon DP, Ahmad NN, Boybeyi Z, Dunn TJ, Hall MS, Jin Y, Lee PCS, Mays DE, Madala RV (2002) An operational multiscale hurricane forecasting system. Mont Weather Rev 130(7):1830–1847
122. Gordon WN, Hall CA (1973) Construction of curvilinear coordinate systems and application to mesh generation. Int J Numer Methods Eng 7:461–477
123. Gravemeier V (2003) The variational multiscale method for laminar and turbulent incompressible flow. Ph.D. thesis, Universitat Stuttgart
124. Grell G, Dudhia J, Stauffer D (1995) A description of the fifth-generation penn state/ncar mesoscale model (mm5). Tech. rep., NCAR Technical Note NCART/TN-398+STR
125. Guba O, Taylor MA, Ullrich PA, Overfelt JR, Levy MN (2014) The spectral element method on variable resolution grids: evaluating grid sensitivity and resolution-aware n umerical viscosity. Geosci Model Dev 7:4081–4117
126. Guermond J, Marra A, Quartapelle L (2006) Subgrid stabilized projection method for 2d unsteady flows at high reynolds numbers. Comput Methods Appl Mech Eng 195:5857–5876

127. Guermond J, Pasquetti R (2009) Entropy viscosity method for high-order approximations of conservation laws. In: Proceedings of the ICOSAHOM 2009 conference, Trondheim, Norway. Springer
128. Guermond JL, Pasquetti R (2008) Entropy-based nonlinear viscosity for Fourier approximations of conservation laws. C R Acad Sci Ser I 346:801–806
129. Guermond JL, Pasquetti R, Popov B (2011) Entropy viscosity method for nonlinear conservation laws. J Comput Phys 230:4248–4267
130. Guo BY, Ma HP, Tadmor E (2001) Spectral vanishing viscosity method for nonlinear conservation laws. SIAM J Numer Anal 39:1254
131. Haidvogel DB, Curchitser E, Iskandarani M, Hughes R, Taylor M (1997) Global modelling of the ocean and atmosphere using the spectral element method. Atmos Ocean 35:505–531
132. Hansbo P (1993) Explicit streamline diffusion finite element methods for the compressible Euler equations in conservation variables. J Comput Phys 109:274–288
133. Hesthaven JS, Warburton T (2008) Nodal discontinuous Galerkin methods: algorithms, analysis, and applications, vol 54. Springer, New York
134. Heus T, van Heerwaarden CC, Jonker H, Siebesma AP, Axelsen S, van den Dries K, Geoffroy O, Moene AF, Pino D, de Roode SR, Vilà-Guerau de Arellano J (2010) Formulation of the Dutch atmospheric large-eddy simulation (DALE) and overview of its applications. Geosci Model Dev 3:415–555
135. Hodur R (1997) The naval research laboratory's coupled ocean/atmosphere mesoscale prediction system (coamps). Mon Weather Rev 125:1414–1430
136. Hogan TF, Liu M, Ridout JA, Peng MS, Whitcomb TR, Ruston BC, Reynolds CA, Eckermann SD, Moskaitis JR, Baker NL (2014) The navy global environmental model. Oceanography 27(3):116–125
137. Holmstrom I (1963) On a method for parametric representation of the state of the atmosphere. Tellus 15:127–149
138. Holton J (2004) An introduction to dynamic meteorology, 4th edn. Elsevier, Amsterdam Internation Geophysics Series: Vol. 88
139. Houze RA (1993) Cloud dynamics. Academic Press, San Diego
140. Houzeaux G, Aubry R, Vázquez M (2011) Extension of fractional step techniques for incompressible flows: the preconditioned orthomin(1) for the pressure schur complement. Comput Fluids 44:297–313
141. Houzeaux G, Eguzkitza B, Vázquez M (2009) A variational multiscale model for the advection–diffusion–reaction equation. Commun Numer Methods Eng 25:787–809
142. Houzeaux G, Vázquez M, Aubry R, Cela JM (2009) A massively parallel fractional step solver for incompressible flows. J Comput Phys 228:6316–6332
143. Hughes TJR (1995) Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. Comput Methods Appl Mech Eng 127:387–401
144. Hughes TJR, Sangalli G (2007) Variational multiscale analysis: the finie-scale Green's function, projection, optimization, localization, and stabilized methods. SIAM J Numer Anal 45:539–557
145. Hughes TJR, Scovazzi G, Tezduyar TE (2010) Stabilized methods for compressible flows. J Sci Comput 43:343–368
146. Hughes TJR, Brooks AN (1979) A multidimensional upwind scheme with no crosswind diffusion. In: Hughes TJR (ed) Finite element methods for convection dominated flows, vol 32. ASME, New York, pp 19–35
147. Hughes TJR, Brooks AN (1982) A theoretical framework for Petrov–Galerkin methods with discontinuous weighting functions: application to the streamline-upwind procedure. Finite Elem Fluids 4:47–65
148. Hughes TJR, Cottrell JA, Bazilevs Y (2005) Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput Methods Appl Mech Eng 194:4135–4195
149. Hughes TJR, Feijóo G, Mazzei L, Quincy J (1998) The variational multiscale method—a paradigm for computational mechanics. Comput Methods Appl Mech Eng 166:3–24
150. Hughes TJR, Franca LP, Hulbert GM (1989) A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Last-squares method for advective–diffusive equations. Comput Methods Appl Mech Eng 73:173–189
151. Hughes TJR, Mallet M (1986) A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective–diffusive systems. Comput Methods Appl Mech Eng 58:305–328
152. Hughes TJR, Mallet M, Mizukami A (1986) A new finite element formulation for computational fluid dynamics: II. Beyond SUPG. Comput Methods Appl Mech Eng 54:341–355
153. Hughes TJR, Mazzei L, Jansen KE (2000) Large eddy simulation and the variational multiscale method. Comput Vis Sci 3:47–59
154. Hughes TJR, Scovazzi G, Bochev PB, Buffa A (2006) A multiscale discontinuous Galerkin method with the computational structure of a continuous Galerkin method. Comput Methods Appl Mech Eng 195:2761–2787
155. Hughes TJR, Stewart J (1996) A space–time formulation for multiscale phenomena. J Comput Appl Math 74:217–229
156. Hughes TJR, Tezduyar T (1984) Finite element methods for first-order hyperbolic systems with particular emphasis pn the compressible Euler equations. Comput Methods Appl Mech Eng 45:217–284
157. Hughes TJR (2000) The finite element method: linear static and dynamics finite element analysis, 2nd edn. Dover, New York
158. Iskandarani M, Haidvogel DB, Boyd JP (1995) A staggered spectral element model with application to the oceanic shallow water equations. Int J Numer Methods Fluids 20:393–414
159. Jablonowski C (2004) Adaptive grids in weather and climate modeling. Ph.D. thesis, The University of Michigan
160. Jablonowski C, Williamson D (2006) A baroclinic instability test case for atmospheric model dynamical cores. Q J R Meteorol Soc 132:2943–2975
161. Jablonowski C, Williamson DL (2011) The pros and cons of diffusion, filters and fixers in atmospheric general circulation models. In: Lauritzen PH, Jablonowski C, Taylor MA, Nair RD (eds) Numerical techniques for global atmospheric models. Lecture notes in computational science and engineering, vol 80. Springer, pp 381–482
162. Jähn M, Knoth O, König M, Vogelsberg U (2014) ASAM v2.7: a compressible atmospheric model with a Cartesian cut cell approach. Geosci Model Dev Disc 7(4):4463–4525
163. Jamet P (1978) Galerkin-type approximations which are discontinuous in time for parabolic equations in a variable domain. SIAM J Numer Anal 15:912–928
164. Janjic Z (1989) On the pressure gradient force error in σ-coordinate spectral models. Mon Weather Rev 117:2285–2292
165. Janjic Z (1994) The step-mountain Eta coordinate model: further developments of the convection, viscous sublayer, and turbulence closure schemes. Mon Weather Rev 122:927–945
166. Janjic Z (2003) A nonhydrostatic model based on a new approach. Meteorol Atmos Phys 82:271–285
167. Janjic Z, Gerrity J, Nickovic S (2001) An alternative approach to non-hydrostatic modeling. Mon Weather Rev 129:1164–1178
168. John V, Knobloch P (2007) On spurious oscillations at layers diminishing (*sold*) methods for convection–diffusion equations: part I—a review. Comput Methods Appl Mech Eng 196:2197–2215

169. Johnson C (1987) Numerical solution of partial differential equations by the finite element method. Cambridge University Press, Cambridge
170. Johnson C, Nävert U, Pitkaranta J (1984) Finite element methods for linear hyperbolic problems. Comput Methods Appl Mech Eng 45:285–312
171. Johnson C, Szepessy A (1988) Shock-capturing streamline diffusion finite element methods for nonlinear conservation laws. In: Hughes TJR, Tezduyar T (eds) AMD, The American Society of Mechanical Engineers, vol 95
172. Kanevsky A, Carpenter MH, Hesthaven JS (2006) Idempotent filtering in spectral and spectral element methods. J Comput Phys 220:41–58
173. Karniadakis G, Sherwin S (2005) Spectral/hp element methods for computational fluid dynamics, 2nd edn. Oxford University Press, Oxford
174. Karypis G, Kumar V (1998) A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J Sci Comput 20(1):359–392
175. Kaul KU (2010) Three-dimensional elliptic grid generation with fully automatic boundary constraints. J Comput Phys 229:5966–5979
176. Kelly JF, Giraldo FX (2012) Continuous and discontinuous Galerkin methods for a scalable three-dimensional nonhydrostatic atmospheric model: limited-area mode. J Comput Phys 231:7988–8008
177. Kessler E (1969) On the distribution and continuity of water substance in atmospheric circulation. Meteorol Monogr 10:32
178. Klein R (2000) Asymptotic analyses for atmospheric flows and the construction of asymptotically adaptive numerical methods. ZAMM 80:765–777
179. Klein R, Achatz U, Bresch D, Knio O, Smolarkiewicz PK (2010) Regime of validity of soundproof atmospheric flow models. J Atmos Sci 67:3226–3237
180. Klemp J (2011) A terrain-following coordinate with smoothed coordinate surfaces. Mon Weather Rev 139:2163–2169
181. Klemp J, Wilhelmson R (1978) The simulation of three-dimensional convective storm dynamics. J Atmos Sci 35:1070–1096
182. Knupp PM, Steinberg S (1993) Fundamentals of grid generation. CRC-Press, Boca Raton
183. Koobus B, Farhat C (2004) A variational multiscale method for the large eddy simulation of compressible turbulent flows on unstructured meshes - application to vortex shedding. Comput Methods Appl Mech Eng 193:1367–1383
184. Kopera MA, Giraldo FX (2014) Analysis of adaptive mesh refinement for IMEX discontinuous Galerkin solutions of the compressible Euler equations with application to atmospheric simulations. J Comput Phys 275:92–117
185. Kopera MA, Giraldo FX (2014) Mass conservation of the unified continuous and discontinuous element-based Galerkin methods on dynamically adaptive grids with application to atmospheric simulations. J Comput Phys. doi:10.1016/j.jcp.2015.05.010 (to appear)
186. Kopriva DA (1996) A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method. J Comput Phys 128(2):475–488
187. Krivodonova L (2007) Limiters for high-order discontinuous Galerkin methods. J Comput Phys 226:879–896
188. Kühnlein C, Smolarkiewicz PK, Dörnbrack A (2012) Modelling atmospheric flows with adaptive moving meshes. J Comput Phys 231(7):2741–2763
189. Kwizak M, Robert A (1971) A semi-implicit scheme for grid point atmospheric models of the primitive equations. Mon Weather Rev 99:32–36
190. Lang J, Cao W, Huang W, Russell RD (2003) A two-dimensional moving finite element method with local refinement based on a posteriori error estimates. Appl Numer Math 46(1):75–94
191. Lanser D, Blom JG, Verwer JG (2001) Time integration of the shallow water equations in spherical geometry. J Comput Phys 171:373–393
192. Laprise R (1992) The euler equations of motion with hydrostatic pressure as an independent variable. Mon Weather Rev 120:197–207
193. Lauter M, Giraldo FX, Handorf D, Dethloff K (2008) A discontinuous Galerkin method for the shallow water equations using spherical triangular coordinates. J Comput Phys 227:10226–10242
194. Le Beau GJ, Ray SE, Aliabadi SK, Tezduyar TE (1993) SUPG finite element computation of compressible flows with the entropy and conservation variables formulations. Comput Methods Appl Mech Eng 104:397–422
195. Lee J, Bleck R, MacDonald A, Bao J, Benjamin S, Middlecoff J, Wang N, Brown J (2008) Fim: a vertically flow-following, finite-volume icosahedral model. In: 22nd Conference on weather analysis forecasting/18th conference on numerical weather prediction, Park City, UT, Am Meteorol Soc (preprints)
196. Lee JLL, MacDonald AE (2009) A finite-volume icosahedral shallow-water model on a local coordinate. Mon Weather Rev 137(4):1422–1437
197. Leonard A (1974) Energy cascade in large eddy simulations of turbulent fluid flows. Adv Geophys 18:237–248
198. Lesaint P, Raviart PA (1974) On a finite element method for solving the neutron transport equation. Academic Press, San Diego
199. Levasseur V, Sagaut P, Chalot F, Davroux A (2006) An entropy-variable-based vms/gls method for the simulation of compressible flows on unstructured grids. Comput Methods Appl Mech Eng 195(9–12):1154–1179. doi:10.1016/j.cma.2005.04.009
200. Ley GW, Elsberry RL (1976) Forecasts of typhoon irma using a nested grid model. Mon Weather Rev 104:1154
201. Lilly DK (1962) On the numerical simulation of buoyant convection. Tellus 14:148–172
202. Lipps F, Hemler R (1982) A scale analysis of deep moist convection and some related numerical calculations. J Atmos Sci 29:2192–2210
203. Lohner R, Mut F, Jebral J, Aubry R, Houzeaux G (2011) Deflated preconditioned conjugate gradient solvers for the pressure-Poisson equation: extensions and improvements. Int J Numer Methods Fluids 87:2–14
204. Ma H (1993) A spectral element basin model for the shallow water equations. J Comput Phys 109:133–149
205. Maday Y, Mavriplis C, Patera AT (1988) Nonconforming mortar element methods: application to spectral discretizations. Institute for Computer Applications in Science and Engineering, NASA Langley Research Center
206. Majewski D, Liermann D, Prohl P, Ritter B, Buchhold M, Hanisch T, Paul G, Wergen W (2002) The operational global icosahedral–hexagonal gridpoint model GME: description and high-resolution tests. Mon Weather Rev 130:319–338
207. Marchuk GI (1974) Numerical methods in weather prediction. Academic Press, San Diego
208. Marras S (2012) Variational multiscale stabilization of finite and spectral elements for dry and moist atmospheric problems. Ph.D. thesis, Universitat Politécnica de Catalunya
209. Marras S, Giraldo FX (2015) A parameter-free dynamic alternative to hyper-viscosity for coupled transport equations: application to the simulation of 3D squall lines using spectral elements. J Comput Phys 283:360–373

210. Marras S, Kelly JF, Giraldo FX, Vázquez M (2012) Variational multiscale stabilization of high-order spectral elements for the advection–diffusion equation. J Comput Phys 231:7187–7213
211. Marras S, Kopera M, Giraldo FX (2014) Simulation of shallow water jets with a unified element-based continuous/discontinuous Galerkin model with grid flexibility on the sphere. Q J R Meteorol Soc. doi:10.1002/qj.2474
212. Marras S, Moragues M, Vázquez M, Jorba O, Houzeaux G (2013) A variational multiscale stabilized finite element method for the solution of the Euler equations of nonhydrostatic stratified flows. J Comput Phys 236:380–407
213. Marras S, Moragues M, Vázquez MR, Jorba O, Houzeaux G (2013) Simulations of moist convection by a variational multiscale stabilized finite element method. J Comput Phys 252:195–218
214. Marras S, Müller A, Giraldo, FX (2014) An LES-like stabilization of the spectral element solution of the Euler equations for atmospheric flows. In: WCCM XI—ECCM V—ECFD VI, Barcelona, Spain, pp 1–22
215. Marras S, Müller A, Giraldo FX (2014) Physics-based stabilization of spectral elements for the 3d Euler equations of moist atmospheric convection. In: Proceedings of the ICOSAHOM 2014, LNCS, Salt Lake City, UT. Springer (accepted)
216. Marras S, Nazarov M, Giraldo FX (2015) A stabilized spectral element method based on a dynamic SGS model for LES. Submitted for review (see pre-print). https://www.researchgate.net/publication/271526242_A_stabilized_spectral_element_method_based_on_a_dynamic_SGS_model_for_LES._Euler_and_nonlinear_scalar_equations
217. Mastin CW, Thompson JF (1978) Transformation of three-dimensional regions onto rectangular regions by elliptic systems. Numer Math 29:397–407
218. McGregor JL, Dix MR (2001) The CSIRO conformal-cubic atmospheric gcm. In: Hodnett PF (ed) IUTAM symposium on advances in mathematical modelling of atmosphere and ocean dynamics. Kluwer, Dordrecht, pp 197–202
219. Mesinger F, Janjic Z, Nickovic S, Gavrilov D, Deaven D (1988) The step-mountain coordinate: model description and performance for cases of alpine lee cyclogenesis and for a case of an appalachian redevelopmen. Mon Weather Rev 116:1493–1518
220. Miyakoda K, Rosati A (1977) One-way nested grid models: the interface conditions and the numerical accuracy. Mon Weather Rev 105:1092–1107
221. MMesh3D: a 3D elliptic mesh generation tool for simply connected domains with topography. http://mmesh3d.wikispaces.com/ (2010)
222. Moragues M, Vázquez M, Houzeaux G, Aubry R (2010) Variational multiscale stabilization of compressible flows in parallel architectures. In: International conference on parallel CFD, Kaoshiung, Taiwan
223. Morrison H, Grabowski WW (2008) Modeling supersaturation and subgrid-scale mixing with two-moment bulk warm microphysics. J Atmos Sci 65:792–812
224. Müller A, Behrens J, Giraldo FX, Wirth V (2013) Comparison between adaptive and uniform deiscontinuous Galerkin simulations in 2D dry bubble experiments. J Comput Phys 235:371–393
225. Nair RD, Choi HW, Tufo HM (2009) Computational aspects of a scalable high-order discontinuous Galerkin atmospheric dynamical core. Comput Fluids 38(2):309–319
226. Nair RD, Levy MN, Lauritzen PH (2011) Emerging numerical methods for atmospheric modeling. In: Lauritzen PH, Jablonowski C, Taylor MA, Nair RD (eds) Numerical techniques for global atmospheric models. Lecture notes in computational science and engineering, vol 80. Springer, pp 251–311
227. Nair RD, Thomas SJ, Loft RD (2005) A discontinuous Galerkin global shallow water model. Mon Weather Rev 133:876–888
228. Nair RD, Thomas SJ, Loft RD (2005) A discontinuous Galerkin transport scheme on the cubed sphere. Mon Weather Rev 133:814–828
229. Nazarov M, Hoffman J (2013) Residual-based artificial viscosity for simulation of turbulent compressible flow using adaptive finite element methods. Int J Numer Methods Fluids 71:339–357
230. Neale RB, Chen C, Gettelman A, Lauritzen PH, Park S, Williamson DL, Conley AJ, Garcia R, Kinnison D, Lamarque JF, Marsh D, Mills M, Smith AK, Tilmes S, Vitt F, Morrison H, Cameron-Smith P, Collins WD, Iacono MJ, Easter RC, Ghan SJ, Liu X, Rasch PJ, Taylor MA (2010) Description of the NCAR community atmosphere model (CAM 5.0). Tech. rep., National Center for Atmospheric Research, NCAR
231. Norman MR (2013) Targeting atmosphjeric simulation algorithms for large, distributed-memory, GPU-accelerated computers. In: Yuen DA, Wang L, Chi X, Johnsson L, Ge W, Shi Y (eds) GPU solutions to multi-scale problems in science and engineering, Lecture notes in earth system sciences. Springer, pp 271–282
232. Ockendon H, Ockendon JR (2004) Waves and compressible flow. Springer, Berlin
233. Ogura Y, Phillips N (1962) Scale analysis of deep and shallow convection in the atmosphere. J Atmos Sci 19:173–179
234. Ouvrard H, Koobus B, Dervieux A, Salvetti MV (2010) Classical and variational multiscale LES of the flow around a circular cylinder on unstructured grids. Comput Fluids 39(7):1083–1094. doi:10.1016/j.compfluid.2010.01.017
235. Patera AT (1984) A spectral method for fluid dynamics: laminar flow in a channel expansion. J Comput Phys 54:468–488
236. Persson PO, Peraire J (2006) Sub-cell shock capturing for discontinuous Galerkin methods. In: Proceedings of the 44th AIAA aerospace sciences meeting and exhibit AIAA-2006-112
237. Phillips N (1957) A coordinate system having some special advantages for numerical forecasting. J Meteorol 14:184–185
238. Pielke R, Cotton W, Walko R, Tremback C, Lyons W, Grasso L, Nicholls M, Moran M, Wesley D, Lee T, Copeland J (1992) A comprehensive meteorological modeling system—RAMS. Metorol Atmos Phys 49:69–91
239. Piggott P, Pain P, Gorman GJ, Power P, Goddard AJH (2005) h, r, and hr adaptivity with applications in numerical ocean modelling. Ocean Model 10(1–2):95–113
240. Piomelli U (1993) Application of LES in engineering: an overview. In: Galperin B, Orszag SA (eds) Large eddy simulation of complex engineering and geophysical flows. Cambridge University Press, Cambridge, pp 119–137
241. Pironneau O, Liou J, Tezduyar T (1992) Characteristic-Galerkin and Galerkin/least-squares space–time formulations for the advection–diffusion equation with time-dependent domains. Comput Methods Appl Mech Eng 100:117–141
242. Priestley A (1992) The Taylor–Galerkin method for the shallow-water equations on the sphere. Mon Weather Rev 120:3003–3015
243. Proctor FH (1988) The terminal area simulation system, volume I: theoretical formulation. Tech. rep., NASA, Contractor Report 4046, DOT/FAA/PM-85/50
244. Prusa JM, Smolarkiewicz PK, Wyszogrodzki AA (2008) EULAG, a computational model for multiscale flows. Comput Fluids 37:1193–1207
245. Qaddouri A, Pudykiewicz J, Tanguay M, Girard C, Coté J (2012) Experiments with different discretizations for the shallow-water equations on a sphere. Q J R Meteorol Soc 138:989–1003
246. Quarteroni A, Sacco R, Saleri F (2000) Numerical mathematics, texts in applied mathematics, 1st edn. Springer, Berlin

247. Quarteroni A, Valli A (1994) Numerical approximation of partial differential equations. Springer, Berlin
248. Reed WH, Hill TR (1973) Triangular mesh methods for the neutron transport equation. Tech. Rep. 73, Los Alamos Scientific Laboratory—LA-UR-73-479
249. Restelli M, Giraldo FX (2009) A conservative discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in nonhydrostatic mesoscale modeling. SIAM J Sci Comput 31:2231–2257
250. Richardson L (1922) Weather prediction by numerical process, 1st edn. Cambridge University Press, Cambridge
251. Rispoli F, Saavedra R (2006) A stabilized finite element method based on sgs models for compressible flows. Comput Methods Appl Mech Eng 196:652–664
252. Rispoli F, Saavedra R, Corsini A, Tezduyar TE (2007) Computation of inviscid compressible flow with v-sgs stabilization and $yz\beta$ shock capturing. Int J Numer Methods Fluids 54:695–706
253. Ritz w (1909) über eine neue methode zur lösung gewisser variationsprobleme der mathematischen physik. J Reine Angew Math 135:1–61
254. Ronquist EM (1996) Convection treatment using spectral elements of different order. Int J Numer Methods Fluids 22:241–264
255. Rood RB, Space G, Earth DOTG, Lyster P, Sawyer W, Takacs LL (1997) Design of the goddard earth observing system (geos) parallel general circulation model (gcm)
256. Room R (2001) Nonhydrostatic adiabatic kernel for hirlam. Part i: Fundamentals of nonhydrostatic dynamics in pressure-related coordinates. Tech. Rep. 25, HIRLAM Technical Report—MeteoFr and Consortium
257. Room R (2002) Nonhydrostatic adiabatic kernel for hirlam. Part III: semi-implicit eulerian scheme. Tech. Rep. 55, HIRLAM Technical Report—MeteoFr and Consortium
258. Rosenberg D, Fournier A, Fischer P, Pouquet A (2006) Geophysical–astrophysical spectral-element adaptive refinement (gaspar): object-oriented h-adaptive fluid dynamics simulation. J Comput Phys 215:59–80
259. Russell WS, Eiseman PR (1998) A boundary conforming structured grid for global ocean circulation studies. Int J Numer Methods Fluids 28:761–788
260. Sagaut P (2000) Large eddy simulation for incompressible flows. An introduction. Springer, Berlin
261. Saito K, Fujita T, Yamada Y, Ishida JI, Kumagai Y, Aranami K, Ohmori S, Nagasawa R, Kumagai S, Muroi C, Kato T, Erro H, Yamazaki Y (2006) The operational JMA nonhydrostatic mesoscale model. Mon Weather Rev 134:1266–1298
262. Satoh M, Matsuno T, Tomita H, Miura H, Nasuno T, Iga S (2008) Nonhydrostatic icosahedral atmospheric model (nicam) for global cloud resolving simulations. J Comput Phys 227:3486–3514
263. Schar C, Leuenberger D, Fuhrer O, Luthic D, Girard C (2002) A new terrain-following vertical coordinate formulation for atmospheric prediction models. Mon Weather Rev 130:2459–2480
264. Schwanenberg D, Kiem R, Kongeter J (2000) Discontinuous Galerkin method for the shallow water equations. Springer, Heidelberg, pp 289–309
265. Sert C, Beskok A (2006) Spectral element formulations on nonconforming grids: a comparative study of pointwise matching and integral projection methods. J Comput Phys 211(1):300–325
266. Shahbazi K, Fischer PF, Ethier CR (2007) A high-order discontinuous Galerkin method for the unsteady incompressible Navier–Stokes equations. J Comput Phys 222(1):391–407
267. Shakib F, Hughes TJR, Johan Z (1991) A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations. Comput Methods Appl Mech Eng 89:141–291
268. Simmons A, Burridge D (1981) An energy and angular-momentum conserving vertical finite-difference scheme and hybrid vertical coordinates. Mon Weather Rev 109:758–766
269. Simons T (1968) A three-dimensional spectral prediction equation. J Atmos Sci 127:1–27
270. Skamarock W, Klemp J (1993) Adaptive grid refinement for two-dimensional and three-dimensional nonhydrostatic atmospheric flow. Mon Weather Rev 121:788–804
271. Skamarock W, Klemp J, Dudhia J, Gill D, Barker D, Wang W, Powers J (2007) A description of the advanced research WRF version 2. Tech. Rep. 468, NCAR TN STR
272. Skamarock W, Oliger J, Street RL (1989) Adaptive grid refinement for numerical weather prediction. J Comput Phys 80(1):27–60
273. Skamarock WC, Klemp JB, Duda MG, Fowler LD, Park SHH, Ringler TD (2012) A multiscale nonhydrostatic atmospheric model using centroidal voronoi tesselations and c-grid staggering. Mon Weather Rev 140(9):3090–3105
274. Smagorinsky J (1963) General circulation experiments with the primitive equations: I. The basic experiement. Mon Weather Rev 91:99–164
275. Smolarkiewicz PK, Szmelter J, Wyszogrodzki AA (2013) An unstructured-mesh atmospheric model for nonhydrostatic dynamics. J Comput Phys 254:184–199
276. Soong S, Ogura Y (1973) A comparison between axisymmetric abd slab-symmetric cumulus cloud models. J. Atmos. Sci. 30:879–893
277. Soto O, Lohner R, Camelli F (2003) A linelet preconditioner for incompressible flow solvers. Int J Numer Methods Heat Fluid Flow 13:133–147
278. St-Cyr A, Jablonowski C, Dennis JM, Tufo HM, Thomas SJ (2008) A comparison of two shallow-water models with nonconforming adaptive grids. Mon Weather Rev 136:1898–1922
279. Staniforth A (1984) The application of the finite-element method to meteorological simulations—a review. Int J Numer Methods Fluids 4:1–12
280. Staniforth AN, Mitchell HL (1978) A variable-resolution finite-element technique for regional forecasting with the primitive equations. Mon Weather Rev 106:439–447
281. Steppeler J, Bitzer H, Bonaventura L (2002) Nonhydrostatic atmospheric modelling using a z-coordinate representation. Mon Weather Rev 130:2143–2149
282. Steppeler J, Hess R, Schattler U, Bonaventura L (2003) Review of numerical methods for nonhydrostatic weather prediction models. Meteorol Atmos Phys 82:287–301
283. Stevens B, Giorgetta M, Esch M, Mauritsen T, Crueger T, Ras S, Salzmann M, Schmidt H, Bader J, Block K, Brokopf R, Fast I, Kinne S, Kornblueh L, Lohmann U, Pincus R, Reichler T, Roechner E (2013) Atmospheric component of the MPI-M earth system model: ECHAM6. J Adv Model Earth Syst 5:146–172
284. Straka J, Wilhelmson R, Wicker L, Anderson J, Droegemeier K (1993) Numerical solution of a nonlinear density current: a benchmark solution and comparisons. Int J Numer Methods Fluids 17:1–22
285. Strang G, Fix GJ (1973) An analysis of the finite element method, vol 212. Wellesley-Cambridge, Wellesley
286. Sundqvist H (1976) On vertical interpolation and truncation in connection with the use of sigma system models. Atmosphere 14:37–52
287. Tabata M (1978) Uniform convergence of the upwind finite element approximation for semilinear parabolic problems. J Math Kyoto Univ 18:327–351
288. Tabata M (1979) Some applications of the upwind finite element method. Theor Appl Mech 27:277–282

289. Tabata M (1985) Symmetric finite element approximations for convection–diffusion problems. Theor Appl Mech 33:445–453
290. Tadmor E (1989) Convergence of spectral methods for nonlinear conservation laws. SIAM J Numer Anal 26:30–44
291. Tanguay M, Robert A, Laprise R (1990) A semi-implicit semi-Lagrangian fully compressible regional forecast model. Mon Weather Rev 118:1970–1980
292. Tapp MC, White PW (1976) A non-hydrostatic mesoscale model. Q J R Meteorol Soc 102:277–296
293. Tatsumi Y (1983) An economical explicit time integration scheme for a primitive model. J Meteorol Soc Jpn 61:269–287
294. Taylor M, Tribbia J, Iskandarani M (1997) The spectral element method for the shallow water equations on the sphere. J Comput Phys 130:92–108
295. Tezduyar T, Senga M (2007) SUPG finite element computation of inviscid supersonic flows with $yz\beta$ shock-capturing. Comput Fluids 36:147–159
296. Thomas SJ, Loft R (2005) The NCAR spectral element climate dynamical core: semi-implicit Eulerian formulation. J Sci Comput 25:307–322
297. Thompson JF, Mastin CW, Thames FC (1974) Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies. J Comput Phys 15(3):299–319
298. Thompson JF, Warsi ZUA, Mastin CW (1985) Numerical Grid Generation: foundations and applications. North-Holland, Amsterdam
299. Thuburn J (2011) Some basic dynamics relevant to the design of atmospheric model dynamical cores. In: Lauritzen PH, Jablonowski C, Taylor MA, Nair RD (eds) Numerical techniques for global atmospheric Models, Lecture notes in computational science and engineering, vol 80. Springer, pp 3–27
300. Thuburn J (2011) Vertical discretizations: some basic ideas. In: Lauritzen PH, Jablonowski C, Taylor MA, Nair RD (eds) Numerical techniques for global atmospheric models, Lecture notes in computational science and engineering, vol 80. Springer, pp 59–74
301. Thuburn J, Cotter CJ (2015) A primal-dual mimetic finite element scheme for the rotating shallow water equations on polygonal spherical meshes. J Comput Phys 290:274–297
302. Toro E (1999) Riemann solvers and numerical methods for fluid dynamics, 2nd edn. Springer, Berlin
303. Tumolo G, Bonaventura L, Restelli M (2013) A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations. J Comput Phys 232:46–67
304. Ullrich P, Jablonowski C (2012) MCore: a non-hydrostatic atmospheric dynamical core utilizing high-order finite-volume methods. J Comput Phys 231(15):5078–5108
305. Untch A, Hortal M (2004) A finite-element scheme for the vertical discretization of the semi-Lagrangian version of the ecmwf forecast model. Q J R Meteorol Soc 130:1505–1530
306. van der Bos F, van der Vegt JJ, Geurts BJ (2007) A multiscale formulation for compressible turbulent flow suitable for general variational discretization techniques. Comput Methods Appl Mech Eng 196:2863–2875
307. Vandeven H (1991) Family of spectral filters for discontinuous problems. J Sci Comput 6:159–192
308. Vázquez M, Houzeaux G, Korik S, Artigues A, Aguado-Sierra J, Aris R, Mira D, Calet H, Cucchietti F, Owen H, Taha A, Cela JM (2014) Alya: towards exascale for engineering simulation codes. arXiv:1404.4881
309. Walko RL, Avissar R (2008) The ocean–land–atmosphere model (OLAM). Part I: shallow-water tests. Mon Weather Rev 136:4033–4044
310. Walko RL, Avissar R (2008) The ocean–land–atmosphere model (OLAM). Part II: formulation and tests of the nonydrostatic dynamic core. Mon Weather Rev 136:4045–4064
311. Wan H, Giorgetta MA, Zängl G, Restelli M, Majewski D, Bonaventura L, Fröhlich K, Reinert D, Rípodas P, Kornblueh L (2013) The ICON-1.2 hydrostatic atmospheric dynamical core on triangular grids—part 1: formulation and performance of the baseline version. Geosci Model Dev Disc 6:59–119
312. Warburton TC, Karniadakis GE (1999) A discontinuous Galerkin method for the viscous MHD equations. J Comput Phys 152(2):608–641
313. Weller H, Shahrokhi A (2014) Curl free pressure gradients over orography in a solution of the fully compressibe Euler equations with implicit treatment of acoustic and gravity waves. Tech. rep., U. Reading, UK
314. Weller H, Lock SJ, Wood N (2013) Runge–Kutta IMEX schemes for the horizontally explicit/vertically implicit (HEVI) solution of wave equations. J Comput Phys 252:365–381
315. Weller H, Ringler T, Piggott M, Wood N (2010) Challenges facing adaptive mesh modeling of the atmosphere and ocean. Bull Am Meteorol Soc 91:105–108
316. Weller H, Weller H, Fournier A (2009) Voronoi, Delaunay, and block-structured mesh refinement for solution of the shallow-water equations on the sphere. Mon Weather Rev 137:4208–4224
317. White AA, Hoskins BJ, Roulstone I, Staniforth A (2005) Consistent approximate models of the global atmosphere: shallow, deep, hydrostatic, quasi-hydrostatic and non-hydrostatic. Q J R Meteorol Soc 131:2081–2107
318. Wicker L, Skamarock W (1998) A time-splitting scheme for the elastic equations incorporating second-order Runge–Kutta time differencing. Mon Weather Rev 126:1992–1999
319. Wicker L, Skamarock W (2002) Time-splitting methods for elastic models using forward time schemes. Mon Weather Rev 130:2088–2097
320. Wilcox LC, Stadler G, Burstedde C, Ghattas O (2010) A high-order discontinuous Galerkin method for wave propagation through coupled elastic–acoustic media. J Comput Phys 229(24):9373–9396
321. Williamson D (2007) The evolution of dynamical cores for global atmospheric models. J Meteorol Soc Jpn 85B:241–269
322. Wood N, Staniforth A, White A, Allen T, Diamantakis M, Gross M, Melvin T, Smith C, Vosper S, Zerroukat M, Thuburn J (2014) An inherently mass-conserving semi-implicit semi-Lagrangian discretization of the deep-atmosphere global non-hydrostatic equations. Q J R Meteorol Soc 140:1505–1520
323. Xue M, Droegemeier K, Wong V (2000) The advanced regional prediction system (arps)—a multi-scale nonhydrostatic atmospheric simulation and prediction model. Part i: model dynamics and verification. Meteorol Atmos Phys 75:161–193
324. Yang H (1985) Finite element structural analysis, international series in civil engineering and engineering mechanics, 1st edn. Prentice-Hall, Englewood Cliffs
325. Yang X, Hu J, Chen D, Zhang H, Shen X, Chen J, Ji L (2008) Verification of GRAPES unified global and regional numerical weather prediction model dynamic core. Chin Sci Bull 53:3458–3464
326. Yeh K, Coté J, Gravel S, Methot A, Patoine A, Roch M, Staniforth A (2002) The CMC-MRB global environmental multiscale GEM model. Part III: nonhydrostatic formulation. Mon Weather Rev 130:339–356
327. Yelash L, Müller A, Lukáčová-Medvid'ová M, Giraldo FX, Wirth V (2014) Adaptive discontinuous evolution Galerkin method for dry atmospheric flow. J Comput Phys 268:106–133
328. Yu M, Giraldo FX, Peng M, Wang ZJ (2014) Localized artificial viscosity stabilization of discontinuous Galerkin m, ehods for

nonhydrostatic mesoscale atmospheric modeling. Technical report, Kansas University
329. Zhang DL, Chang HR, Seaman NL, Warner TT, Fritsch JM (1986) A two-way interactive nesting procedure with variable terrain resolution. Mon Weather Rev 114:1330–1339
330. Zhao M, Held I, Lin S, Vecchi G (2009) Simulations of global hurricane climatology, interannual variability, and response to global warming using a 50-km resolution GCM. J Clim 22:6653–6678
331. Zienkiewcz O, Nithiarasu P, Codina R, Vázquez M, Ortiz P (1999) The characteristic-based split procedure: an efficient and accurate algorithm for fluid problems. Int J Numer Methods Fluids 31:359–392
332. Zienkiewicz O, Codina R (1995) A general algorithm for compressible and incompressible flow—part i. The split, characteristic-based scheme. Int J Numer Methods Fluids 20:869–885
333. Zienkiewicz O, Taylor R, Nithiarasu P (2005) The finite element method for fluid dynamics, 6th edn. Elsevier, Amsterdam
334. Zingan V, Guermond JL, Morel J, Popov B (2013) Implementation of the entropy viscosity method with the discontinuous Galerkin method. Comput Methods Appl Math Eng 253:479–490