Adaptivity in geophysical flows

Based on joint work with Jan Giesselmann (Stuttgart) and Charalambos Makridakis (Crete)

Tristan Pryer,
Climate Change Initiative for Adaptation, Resilience and Mitigation
University of Reading

Outline

Motivation

Finite element and finite volume methods

A posteriori analysis and mesh adaptivity

Model adaptivity

Conclusions, open problems, future work directions

Outline

- Global numerical weather forecasts.
- Incoorporate observations from planes, ships, air balloons, satelittes, ground based observatiories, etc.
- Able to resolve small scale features when it matters.

Main ingredients in an atmospheric model

- A PDE hierarchy of models

Main ingredients in an atmospheric model

- A PDE hierarchy of models
- A numerical method for a specific PDE model

Main ingredients in an atmospheric model

- A PDE hierarchy of models
- A numerical method for a specific PDE model
- Data assimilation tools

Difficulties

- Which PDE model is the correct one to use?

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system. Stability of numerical schemes for fast fluid flows equations is a challenging topic.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.
- Computational complexity.

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE system. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.
- Computational complexity. As computational capacity grows, one typically invests in increased resolution, more ensemble runs, addition of PDEs governing new physical processes. This has a massive impact on the computational complexity.

Purpose of this talk

I will show some novel techniques for reducing the computational complexity of geophysical simulations. The focus being:

Purpose of this talk

I will show some novel techniques for reducing the computational complexity of geophysical simulations. The focus being:

- Automated mesh adaptivity;

Purpose of this talk

I will show some novel techniques for reducing the computational complexity of geophysical simulations. The focus being:

- Automated mesh adaptivity;
- Automated PDE model adaptivity.

The key word is 'automated', meaning no user input.

Outline

Motivation

Finite element and finite volume methods

A posteriori analysis and mesh adaptivity

Model adaptivity

Conclusions, open problems, future work directions

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.
- Computational complexity. As computational capacity grows, one typically invests in increased resolution, more ensemble runs, addition of PDEs governing new physical processes. This has a massive impact on the computational complexity.

Stable discretisations

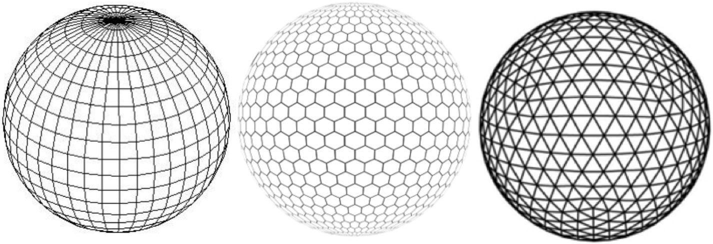
- Finite Element Methods are attractive for their simplicity and robustness, especially in structural mechanics and heat flow simulations, owing to their variational interpretation and origins.

Stable discretisations

- Finite Element Methods are attractive for their simplicity and robustness, especially in structural mechanics and heat flow simulations, owing to their variational interpretation and origins.
- Discontinuous Galerkin and finite volume methods, on the other hand, are popular in fluid flows and fast convection simulations, due to their superior numerical stability properties, stemming from the ability to incorporate general numerical flux functions seamlessly.

Admissible meshes

DG and FV methods are particularly easy to defined over general meshes.



A 1d linear example

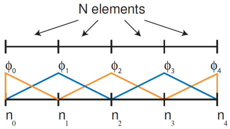
Consider the transport equation

$$u_t + u_x = 0 \text{ over } [0, 1],$$

with periodic boundary condtions

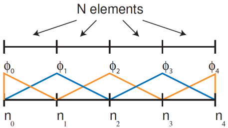
Continous FEM

The standard (continuous) finite element procedure is to write down a finite dimensional basis:



Continous FEM

The standard (continuous) finite element procedure is to write down a finite dimensional basis:

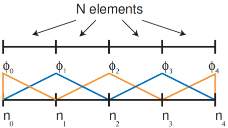


Make an ansatz the $u(x) \approx u_h(x) := \sum_{j=0}^{N} U_j(t)\phi_j(x)$, write the problem weakly: To seek u_h such that

$$\int_0^1 u_{h,t}\phi_i - u_h\phi_{i,x}\,\mathrm{d}x + [u_h\phi_i]_{x=0}^{x=1} = 0 \quad \forall\, i = 0, \dots, N$$

Continous FEM

The standard (continuous) finite element procedure is to write down a finite dimensional basis:



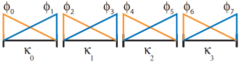
Make an ansatz the $u(x) \approx u_h(x) := \sum_{j=0}^{N} U_j(t)\phi_j(x)$, write the problem weakly: To seek u_h such that

$$\int_0^1 u_{h,t}\phi_i - u_h\phi_{i,x}\,\mathrm{d}x + [u_h\phi_i]_{x=0}^{x=1} = 0 \quad \forall\, i = 0,\ldots,N$$

which yields a linear system of ODEs $\boldsymbol{A}\boldsymbol{U}(t) = \boldsymbol{F}$, for the unknowns $\boldsymbol{U}(t) = (U_0(t), U_1(t), \ldots, U_N(t))^{\intercal}$. Apply your favourite Runge-Kutta method.

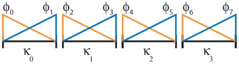
Discontinuous Galerkin

The standard discontinuous Galerkin procedure differs in the finite dimensional basis chosen:



Discontinuous Galerkin

The standard discontinuous Galerkin procedure differs in the finite dimensional basis chosen:

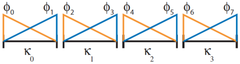


It doesn't look too different, but there are more basis functions and continuity is no longer enforced in the solution. Again, make an ansatz the $u(x) \approx u_h(x) := \sum_{j=0}^{2N-1} U_j(t)\phi_j(x)$ and write the problem weakly, this time it needs to be elementwise: To seek u_h such that

$$\sum_K \int_K u_{h,t}\phi_i - u_h\phi_{i,x}\,\mathrm{d}x + [u_h\phi_i]_{n-1/2}^{n+1/2} = 0 \quad \forall\, i = 0, \dots, N.$$

Discontinuous Galerkin

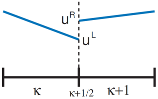
The standard discontinuous Galerkin procedure differs in the finite dimensional basis chosen:



It doesn't look too different, but there are more basis functions and continuity is no longer enforced in the solution. Again, make an ansatz the $u(x) \approx u_h(x) := \sum_{j=0}^{2N-1} U_j(t)\phi_j(x)$ and write the problem weakly, this time it needs to be elementwise: To seek u_h such that

$$\sum_K \int_K u_{h,t}\phi_i - u_h \phi_{i,x} \, \mathrm{d}x + [u_h \phi_i]_{n-1/2}^{n+1/2} = 0 \quad \forall\, i = 0, \dots, N.$$

Notice though, that u_h will be multi-valued at the edges



Numerical fluxes

This problem is well studied, however, and the idea is to use a numerical flux function

$$[u_h]_{n-1/2}^{n+1/2} \to H(u_h^L, u_h^R, 1),$$

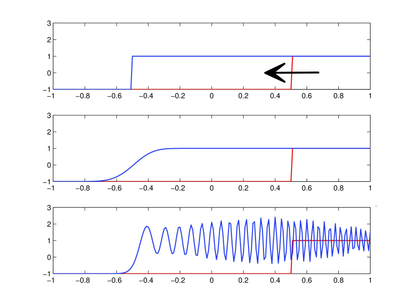
which here will correspond to upwinding.

Numerical fluxes

This problem is well studied, however, and the idea is to use a numerical flux function

$$[u_h]_{n-1/2}^{n+1/2} \to H(u_h^L, u_h^R, 1),$$

which here will correspond to upwinding.



Main ideas

- Discretise the domain you're interested in.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Other topics that I'm brushing under the carpet

- How to generalise schemes to multi dimensional problems.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Other topics that I'm brushing under the carpet

- How to generalise schemes to multi dimensional problems.
- How to generalise to nonlinear problems.

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Other topics that I'm brushing under the carpet

- How to generalise schemes to multi dimensional problems.
- How to generalise to nonlinear problems.
- Which flux functions do a good job?

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Other topics that I'm brushing under the carpet

- How to generalise schemes to multi dimensional problems.
- How to generalise to nonlinear problems.
- Which flux functions do a good job?
- How do you solve the system of ODEs?

Main ideas

- Discretise the domain you're interested in.
- Write the PDE system weakly over each element.
- Take into account the fact that numerical solutions are discontinuous.
- Introduce *appropriate* flux functions for stabilisation.

Other topics that I'm brushing under the carpet

- How to generalise schemes to multi dimensional problems.
- How to generalise to nonlinear problems.
- Which flux functions do a good job?
- How do you solve the system of ODEs?
- Linear/nonlinear solvers.

Outline

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.
- Computational complexity. As computational capacity grows, one typically invests in increased resolution, more ensemble runs, addition of PDEs governing new physical processes. This has a massive impact on the computational complexity.

Aim of an a posteriori estimator

We wish to construct an explicit estimator $\mathscr{E} = \mathscr{E}(u_h)$, dependant *only* on the numerical solution and problem data such that

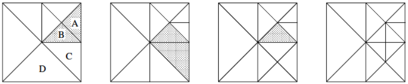
$$\|u - u_h\|_{L^\infty(0,T;L^2(I))} \leq \mathscr{E}(u_h).$$

AND that $\mathscr{E}(u_h)$ should be of the same order as the error. It should also be localisable.

What can I use this for?

These estimators are a crucial component in mesh adaptivity. Mesh adaptivity allows for increased resolution in places where deemed necessary, near features of interest, singularities, shocks, etc.

How does mesh refinement work? Example: newest vertex bisection



- Mark some elements for refinement based on some criteria.
- Split the elements into two "children".
- Check for "hanging nodes", if there is one, refine neighbour.
- The result is called a "conforming triangulation".

This is only one possible algorithm. It's nice as it is simple to implement and provably guarantees shape regularity and that you will not just globally refine the mesh.

When to refine?

Now we know how to refine, let's try to justify when we should refine. Consider the Euler equations of gas dynamics:

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \mathbf{0} \\ \partial_t e + \operatorname{div}((e + p)\boldsymbol{v}) &= 0. \end{aligned}$$

These constitute prognastic equations for density, momentum and energy and 1 diagnostic equation for pressure. We'll look at a 2D situation over a rectangle so there are four prognastic equations.

dG scheme for Euler's equations

We can rewrite Euler's equations as a system with $\boldsymbol{w} = (\rho, \rho\boldsymbol{v}, e)$ for the unknowns and

$$\partial_t \boldsymbol{w} + \operatorname{div}\left(\boldsymbol{f}(\boldsymbol{w})\right) = \boldsymbol{0},$$

with

$$\boldsymbol{f}(\boldsymbol{w}) = \begin{bmatrix} \rho v_1 & \rho v_2 \\ \rho v_1^2 + p & \rho v_1 v_2 \\ \rho v_2 v_1 & \rho v_2^2 + p \\ (e+p)v_1 & (e+p)v_2 \end{bmatrix}.$$

and try to find a $\boldsymbol{w}_h$ such that

$$\int_{\mathscr{T}} \partial_t \boldsymbol{w}_h \cdot \boldsymbol{\phi} - \boldsymbol{f}(\boldsymbol{w}_h) \cdot \nabla \boldsymbol{\phi} + \int_{\mathscr{E}} \boldsymbol{F}(\boldsymbol{w}_h^-, \boldsymbol{w}_h^+) \left[\!\left[\boldsymbol{\phi}\right]\!\right] = \boldsymbol{0} \quad \text{for all } \boldsymbol{\phi} \in \mathbb{V}_q,$$

$$u_h(0) = \mathscr{P}_q[u_0],$$

where $\boldsymbol{F}$ is an appropriate flux function.

A posteriori techniques

Now we have a scheme we can apply an a posteriori argument. For a scalar conservation law there are at least two approaches available to us:

- Those based on L^1-Kruskov techniques [DMO11]
- Those based on L^2-relative entropy techniques [GMP15, GD16].

For systems only the relative entropy is appropriate.

Entropy/ entropy flux pair

The pair $\eta : U \to \mathbb{R}$, and $q : U \to \mathbb{R}$ satisfy

$$\nabla \eta D\boldsymbol{f} = \nabla q. \tag{*}$$

Strong solutions of

$$u_t + \boldsymbol{f}(u)_x = 0 \tag{**}$$

satisfy

$$\eta(u)_t + q(u)_x = 0.$$

Definition

A weak solution of (∗∗) is called an entropy solution with respect to the entropy/ entropy flux pair (η, q) provided it satisfies

$$\eta(u)_t + q(u)_x \leq 0$$

in the sense of distributions. The solution concept is motivated by a vanishing viscosity approach.

Remarks on entropy solutions

- In the scalar case every function $\eta : U \to \mathbb{R}$ is an entropy.
- In the scalar case entropy solutions (satisfying the inequality for all convex entropies) are unique.
- For systems of hyperbolic conservation laws there is usually only one (physically motivated) entropy/entropy flux pair.
- (For systems) entropy solutions need not be unique.
- In most cases the entropy is convex. But there are important cases where it is not.
- The entropy inequality is thought to ensure compatibility with 2^{nd}-law of thermodynamics.

Stability and some PDE theory

Theorem (Dafermos, e.g.)

Given a system of conservation laws with an entropy/entropy flux pair (η, q) *and let* $\mathrm{D}^2\eta$ *be positive definite and bounded. Let* u *be an entropy solution and* v *a Lipschitz solution, then there exist* $C, \tilde{C} > 0$ *such that*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^2(I)} \leq Ce^{\tilde{C}(t)}\|u(0,\cdot) - v(0,\cdot)\|_{L^2(I)}.$$

Stability and some PDE theory

Theorem (Dafermos, e.g.)

Given a system of conservation laws with an entropy/entropy flux pair (η, q) *and let* $\mathrm{D}^2\eta$ *be positive definite and bounded. Let* u *be an entropy solution and* v *a Lipschitz solution, then there exist* $C, \tilde{C} > 0$ *such that*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^2(I)} \leq Ce^{\tilde{C}(t)}\|u(0,\cdot) - v(0,\cdot)\|_{L^2(I)}.$$

Remark 1: *This shows uniqueness of Lipschitz solutions in the class of entropy solutions.*

Stability and some PDE theory

Theorem (Dafermos, e.g.)

Given a system of conservation laws with an entropy/entropy flux pair (η, q) *and let* $\mathrm{D}^2\eta$ *be positive definite and bounded. Let* u *be an entropy solution and* v *a Lipschitz solution, then there exist* $C, \tilde{C} > 0$ *such that*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^2(I)} \leq Ce^{\tilde{C}(t)}\|u(0,\cdot) - v(0,\cdot)\|_{L^2(I)}.$$

Remark 1: *This shows uniqueness of Lipschitz solutions in the class of entropy solutions.*
Remark 2: *In the scalar case the abundance of entropies leads to a much stronger stability result:*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^1(I)} \leq \|u(0,\cdot) - v(0,\cdot)\|_{L^1(I)}.$$

Stability and some PDE theory

Theorem (Dafermos, e.g.)

Given a system of conservation laws with an entropy/entropy flux pair (η, q) *and let* $\mathrm{D}^2\eta$ *be positive definite and bounded. Let* u *be an entropy solution and* v *a Lipschitz solution, then there exist* $C, \tilde{C} > 0$ *such that*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^2(I)} \leq Ce^{\tilde{C}(t)} \|u(0,\cdot) - v(0,\cdot)\|_{L^2(I)}.$$

Remark 1: *This shows uniqueness of Lipschitz solutions in the class of entropy solutions.*
Remark 2: *In the scalar case the abundance of entropies leads to a much stronger stability result:*

$$\|u(t,\cdot) - v(t,\cdot)\|_{L^1(I)} \leq \|u(0,\cdot) - v(0,\cdot)\|_{L^1(I)}.$$

Remark 3: *Note that the estimate is in* L^1 *in the scalar case and in* L^2 *for systems.*

Theorem (The case with residuals)

If $\boldsymbol{v}$ *is a Lipschitz solution of a perturbed problem:*

$$\boldsymbol{u}_t + \operatorname{div}(\boldsymbol{f}(\boldsymbol{u})) = \boldsymbol{0}, \quad \boldsymbol{v}_t + \operatorname{div}(\boldsymbol{f}(\boldsymbol{v})) = \boldsymbol{R},$$

(think of $\boldsymbol{R}$ *as a discrete residual) we have that*

$$\|\boldsymbol{u}(\cdot,t) - \boldsymbol{v}(\cdot,t)\|^2_{\mathrm{L}^2(I)} \leq C\left(\|\boldsymbol{u}_0 - \boldsymbol{v}_0\|^2_{\mathrm{L}^2(I)} + \exp\left(t\,\|\boldsymbol{v}\|_{\mathrm{W}^{1,\infty}(I)}\right)\|\boldsymbol{R}\|^2_{\mathrm{L}^2(I\times(0,t))}\right)$$

Theorem (The case with residuals)

If $\boldsymbol{v}$ *is a Lipschitz solution of a perturbed problem:*

$$\boldsymbol{u}_t + \operatorname{div}\left(\boldsymbol{f}(\boldsymbol{u})\right) = \boldsymbol{0}, \quad \boldsymbol{v}_t + \operatorname{div}\left(\boldsymbol{f}(\boldsymbol{v})\right) = \boldsymbol{R},$$

(think of $\boldsymbol{R}$ *as a discrete residual) we have that*

$$\|\boldsymbol{u}(\cdot,t) - \boldsymbol{v}(\cdot,t)\|^2_{\mathrm{L}^2(I)} \leq C\left(\|\boldsymbol{u}_0 - \boldsymbol{v}_0\|^2_{\mathrm{L}^2(I)} + \exp\left(t\,\|\boldsymbol{v}\|_{\mathrm{W}^{1,\infty}(I)}\right)\|\boldsymbol{R}\|^2_{\mathrm{L}^2(I\times(0,t))}\right)$$

Notice that $\boldsymbol{v}$ *must be Lipschitz!*

The need for reconstruction

We would like to use the relative entropy framework to estimate the difference between exact and approximate solution.

There are two main problems:

- $\boldsymbol{u}_h(t,\cdot)$, $\boldsymbol{u}(t,\cdot)$ are not Lipschitz continuous.
- $\partial_t \boldsymbol{u}_h + \operatorname{div}(\boldsymbol{f}(\boldsymbol{u}_h)) =: \boldsymbol{R}_h$ is just measure valued, $\boldsymbol{R}_h(t,\cdot) \notin L^2(I)$.

The need for reconstruction

We would like to use the relative entropy framework to estimate the difference between exact and approximate solution.

There are two main problems:

- $\boldsymbol{u}_h(t,\cdot)$, $\boldsymbol{u}(t,\cdot)$ are not Lipschitz continuous.
- $\partial_t \boldsymbol{u}_h + \operatorname{div}(\boldsymbol{f}(\boldsymbol{u}_h)) =: \boldsymbol{R}_h$ is just measure valued, $\boldsymbol{R}_h(t,\cdot) \notin L^2(I)$.

Basic idea of reconstruction:
Introduce intermediate quantity $\hat{\boldsymbol{u}}$ such that

- $\hat{\boldsymbol{u}} - \boldsymbol{u}_h$ can be explicitly bounded and is of optimal order.
- $\hat{\boldsymbol{u}}$ is Lipschitz continuous and satisfies a perturbed PDE with an L^2 residual.
- Use $\|\boldsymbol{u} - \boldsymbol{u}_h\| \le \|\boldsymbol{u} - \hat{\boldsymbol{u}}\| + \|\hat{\boldsymbol{u}} - \boldsymbol{u}_h\|$.

Error estimate

- Making use of this stability result and reconstruction techniques it has been shown that for the Richtmer dG scheme, defining

$$\widehat{\boldsymbol{u}}_t + \operatorname{div}\left(\boldsymbol{f}(\widehat{\boldsymbol{u}})\right) = \partial_t(\widehat{\boldsymbol{u}} - \boldsymbol{u}_h) + \operatorname{div}\left(f(\widehat{\boldsymbol{u}}) - \widehat{\boldsymbol{f}}\right) =: \boldsymbol{R}_h$$

and

$$\begin{aligned} L(t) &:= \|\operatorname{div}\left(\widehat{\boldsymbol{u}}(t,\cdot)\right)\|_\infty \\ E(t) &:= \|R_h(t,\cdot)\|^2_{L^2(I)}. \end{aligned}$$

$$\begin{aligned} \|\boldsymbol{u}(t,\cdot) - \boldsymbol{u}_h(t,\cdot)\|^2_{L^2(I)} \leq C\Bigg(\|\widehat{\boldsymbol{u}}(t,\cdot) - \boldsymbol{u}_h(t,\cdot)\|^2_{\mathrm{L}^2(I)} + \int_0^t E(s)\Big(L(s)+1\Big) \\ \times \exp\left(\tilde{C}\int_s^t L(\sigma) + 1 \,\mathrm{d}\,\sigma\right) \mathrm{d}\,s + E(t)\Bigg). \end{aligned}$$

There is also a contribution of the initial error.
- $E(t)$ and $L(t)$ are explicity computable.
- Both $\boldsymbol{u}_h$ and $\widehat{\boldsymbol{u}}$ are computable.

Remarks on error estimate

- The estimate depends exponentially on time.
- The estimate also depends exponentially on $\|\operatorname{div}(\widehat{\boldsymbol{u}})\|_{L^\infty(I)}$. This to be bounded as $h \to 0$ if u is Lipschitz.
- Thus, $\mathscr{E}_1(u_h)$ will not converge for discontinuous u.
- For Lipschitz u we expect $\mathscr{E}(u_h)$ to be of the same order as the error (which depends on polynomial degree and the regularity of u.)

Estimators in action

Applying these estimators to a 2D simulation of Euler's equations over a rectangle with a 'mixing' term.

Outline

Difficulties

- Which PDE model is the correct one to use? The most physically accurate would be compressible Navier-Stokes on a rotating sphere. The current ECMWF model is comprised of 10 evolution equations in 3D. NS forms part of this.
- Find a stable, accurate numerical method for the chosen PDE. Stability of numerical schemes for fast fluid flows equations is a challenging topic.
- Under resolution of small scale features. Different length and time scales present can, in general, not be represented on current generation computational models.
- Computational complexity. As computational capacity grows, one typically invests in increased resolution, more ensemble runs, addition of PDEs governing new physical processes. This has a massive impact on the computational complexity.

Main idea

Rather than trying to approximate the original *full* PDE model globally, a *reduced* PDE model is approximated over some of the domain. The two models are coupled. The numerical scheme "chooses" when and where to switch between the two based on an a posteriori estimator.

The main idea via illustrative example

The “exact” model is given by isothermal Navier-Stokes

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) &= \operatorname{div}(\mu \nabla \boldsymbol{v}) \end{aligned}$$

where ρ denotes density, $\boldsymbol{v}$ denotes velocity and $p = p(\rho)$ is the pressure, given by a constitutive relation as a monotone function of density, and $\mu \geq 0$ is the viscosity parameter.
The reduced model is given by

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) &= 0 \end{aligned}$$

Some motivation

- Arguably the NS system provides a more accurate description of reality since viscous effects which are neglected in Euler's equation play a dominant role in certain flow regimes like thin regions near obstacles, for example, aerofoils exhibiting Prandtls's boundary layers [Nickel 1973].

Some motivation

- Arguably the NS system provides a more accurate description of reality since viscous effects which are neglected in Euler's equation play a dominant role in certain flow regimes like thin regions near obstacles, for example, aerofoils exhibiting Prandtls's boundary layers [Nickel 1973].
- However, viscous effects are negligible in large parts of the computational domain where convective effects dominate [Brezzi, Canuto, and Russo 1989, Coclici and Wendland 2001, Discacciati, Gervasio and Quarteroni 2012].

Some motivation

- Arguably the NS system provides a more accurate description of reality since viscous effects which are neglected in Euler's equation play a dominant role in certain flow regimes like thin regions near obstacles, for example, aerofoils exhibiting Prandtls's boundary layers [Nickel 1973].
- However, viscous effects are negligible in large parts of the computational domain where convective effects dominate [Brezzi, Canuto, and Russo 1989, Coclici and Wendland 2001, Discacciati, Gervasio and Quarteroni 2012].
- Thus, it is desirable to avoid the effort of handling the viscous terms in these parts of the domain, that is, to use the NS system only where needed and simpler models, e.g., (linearised) Euler everywhere else.

Some more motivation

- This insight has lead to the development, of a certain type, of heterogeneous domain decomposition methods where on a certain part of the computational domain the NS equations are solved numerically, whereas the (linearised) Euler equations are used for farfield computations [Utzmann, Schwartzkop, Dumbser and Munz 2006, Borrel, Halpern and Ryan 2011., e.g.].

Some more motivation

- This insight has lead to the development, of a certain type, of heterogeneous domain decomposition methods where on a certain part of the computational domain the NS equations are solved numerically, whereas the (linearised) Euler equations are used for farfield computations [Utzmann, Schwartzkop, Dumbser and Munz 2006, Borrel, Halpern and Ryan 2011., e.g.].
- In those works the domain was decomposed a priori *before* the start of the numerical computation. The accuracy and effciency of those schemes depends sensitively on the placement of the domains. Thus, the user is required to have some physical intuition on where to solve each model.

A different spin

Can this be done in an a posteriori fashion? That is, can we find a way to appropriately decompose the domain into areas where we use the *simple* and *complex* models automatically with no user knowledge or input into the code?

The main idea via illustrative example

Remember we're trying to approximate isothermal Navier-Stokes with fixed viscosity

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) &= \operatorname{div}(\mu \nabla \boldsymbol{v}) \end{aligned}$$

by a non viscous equation

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) &= \boldsymbol{0} \end{aligned}$$

over parts of the domain.

The main idea via illustrative example

Remember we're trying to approximate isothermal Navier-Stokes with fixed viscosity

$$\partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) = 0$$
$$\partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) = \operatorname{div}(\mu \nabla \boldsymbol{v})$$

by a non viscous equation

$$\partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) = 0$$
$$\partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) = \mathbf{0}$$

over parts of the domain. To do this we introduce the coupled problem

$$\partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) = 0$$
$$\partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla(p(\rho)) = \operatorname{div}(\widehat{\mu} \nabla \boldsymbol{v}),$$

where $\widehat{\mu}$ is a piecewise constant function taking values either 0 or μ.

A posteriori modelling error control

Let $\boldsymbol{w} = (\rho, \boldsymbol{v})$ be a weak solution to NS and $\boldsymbol{w}_h = (\rho_h, \boldsymbol{v}_h)$ be an approximation. Under some further technical assumptions, the following a posteriori error estimate holds:

$$\|\boldsymbol{w}(\cdot,t) - \boldsymbol{w}_h(\cdot,t)\|^2_{\mathrm{L}^2(\mathbb{T}^d)} + \int_{\mathbb{T}^d \times (0,t)} \frac{\mu}{4k} \|\nabla \boldsymbol{v} - \nabla \boldsymbol{v}_h\|^2_{\mathrm{L}^2(\mathbb{T}^d)}$$
$$\leq C \left(\|\boldsymbol{w}(\cdot,0) - \boldsymbol{w}_h(\cdot,0)\|^2_{\mathrm{L}^2(\mathbb{T}^d)} + \mathscr{E}_D + \mathscr{E}_M \right)$$

with C, k being constants and

$$\mathscr{E}_M := \|(\mu - \widehat{\mu}) \nabla \widehat{\boldsymbol{v}}\|^2_{\mathrm{L}^2(\mathbb{T}^d \times (0,t))}$$
$$\mathscr{E}_D := \frac{k^2}{\mu} \|\mathscr{R}_P\|^2_{\mathrm{L}^2(0,t;\mathrm{H}^{-1}(\mathbb{T}^d))} + \|\mathscr{R}_H\|^2_{\mathrm{L}^2(\mathbb{T}^d \times (0,t))} .$$

Specifics

Consider the use of model adaptivity on the Isothermal Navier-Stokes system in a situation where Kármán vortices are produced by a flow over a cylinder with a Reynolds number of 100. We impose slip boundary conditions on the top and bottom of the rectangular region, an inflow and outflow on the left and right hand side respectively, compatible with the initial conditions and a no slip condition on the cylinder itself.

The full problem

Now consider the Navier-Stokes-Fourier problem with fixed viscosity

$$\begin{aligned}\partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0\\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \operatorname{div}(\mu \nabla \boldsymbol{v})\\ \partial_t e + \operatorname{div}((e+p)\boldsymbol{v}) &= \operatorname{div}(\mu(\nabla \boldsymbol{v}) \cdot \boldsymbol{v} + \kappa \nabla T).\end{aligned}$$

We're approximating this by the non viscous Euler's equation

$$\begin{aligned}\partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0\\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \boldsymbol{0}\\ \partial_t e + \operatorname{div}((e+p)\boldsymbol{v}) &= 0.\end{aligned}$$

over parts of the domain.

The full problem

Now consider the Navier-Stokes-Fourier problem with fixed viscosity

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \operatorname{div}(\mu \nabla \boldsymbol{v}) \\ \partial_t e + \operatorname{div}((e+p)\boldsymbol{v}) &= \operatorname{div}(\mu (\nabla \boldsymbol{v}) \cdot \boldsymbol{v} + \kappa \nabla T). \end{aligned}$$

We're approximating this by the non viscous Euler's equation

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \mathbf{0} \\ \partial_t e + \operatorname{div}((e+p)\boldsymbol{v}) &= 0. \end{aligned}$$

over parts of the domain. We use the coupled problem

$$\begin{aligned} \partial_t \rho + \operatorname{div}(\rho \boldsymbol{v}) &= 0 \\ \partial_t(\rho \boldsymbol{v}) + \operatorname{div}(\rho \boldsymbol{v} \otimes \boldsymbol{v}) + \nabla p &= \operatorname{div}(\widehat{\mu} \nabla \boldsymbol{v}) \\ \partial_t e + \operatorname{div}((e+p)\boldsymbol{v}) &= \operatorname{div}(\widehat{\mu} (\nabla \boldsymbol{v}) \cdot \boldsymbol{v} + \widehat{\kappa} \nabla T), \end{aligned}$$

where $\widehat{\mu}$ and $\widehat{\kappa}$ are piecewise constant functions taking values either 0 or μ or 0 and κ respectively.

Full Navier-Stokes-Fourier

Consider the use of model adaptivity on the Navier-Stokes-Fourier system in a supersonic flow around an aerofoil. Here we are travelling at Mach 1. We impose slip boundary conditions on the top and bottom of the rectangular region, an inflow and outflow on the left and right hand side respectively, compatible with the initial conditions and a no slip condition on the aerofoil itself.

Full Navier-Stokes-Fourier

Consider the use of model adaptivity on the Navier-Stokes-Fourier system in a supersonic flow over a forward facing step. Here we are travelling at Mach 3.66.

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.
- This procedure is automated and does not rely on a priori knowledge of any features of the model.

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.
- This procedure is automated and does not rely on a priori knowledge of any features of the model.
- Offers the potential for significant speedup in computational runs, by solving a reduced dimensional PDE model *and* a reduced complexity simulation.

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.
- This procedure is automated and does not rely on a priori knowledge of any features of the model.
- Offers the potential for significant speedup in computational runs, by solving a reduced dimensional PDE model *and* a reduced complexity simulation.

Open problems and future work

- How does one couple the two models best in practice? Is a piecewise discontinuous switch between the models too sharp? Additional interface conditions? Multigrid coupling?

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.
- This procedure is automated and does not rely on a priori knowledge of any features of the model.
- Offers the potential for significant speedup in computational runs, by solving a reduced dimensional PDE model *and* a reduced complexity simulation.

Open problems and future work

- How does one couple the two models best in practice? Is a piecewise discontinuous switch between the models too sharp? Additional interface conditions? Multigrid coupling?
- Generalisation to other PDE model hierarchies.

Conclusions

- We have derived a Mathematical framework able to produce a posteriori bounds to drive both mesh refinement and model switching within a numerical simulation.
- This procedure is automated and does not rely on a priori knowledge of any features of the model.
- Offers the potential for significant speedup in computational runs, by solving a reduced dimensional PDE model *and* a reduced complexity simulation.

Open problems and future work

- How does one couple the two models best in practice? Is a piecewise discontinuous switch between the models too sharp? Additional interface conditions? Multigrid coupling?
- Generalisation to other PDE model hierarchies.
- Efficient implementation of a coupled model to really see the benefits in terms of CPU time. Whilst we saw some speedup, we can do better utlising paralellisation in a clever way.

References

- Giesselmann, J., Makridakis, C. and Pryer, T. (2015). A posteriori analysis of discontinuous Galerkin schemes for systems of hyperbolic conservation laws. SIAM Journal on Numerical Analysis, 53(3), 1280-1303.
- Giesselmann, J. and Pryer, T. (2017). A posteriori analysis for dynamic model adaptation in convection dominated problems. To appear in Mathematical Models and Methods in Applied Sciences.