Advances in SBP-SAT-based methods for the prediction and control of compressible turbulent flows

July 10, 2014

Daniel J. Bodony
Department of Aerospace Engineering
University of Illinois at Urbana-Champaign

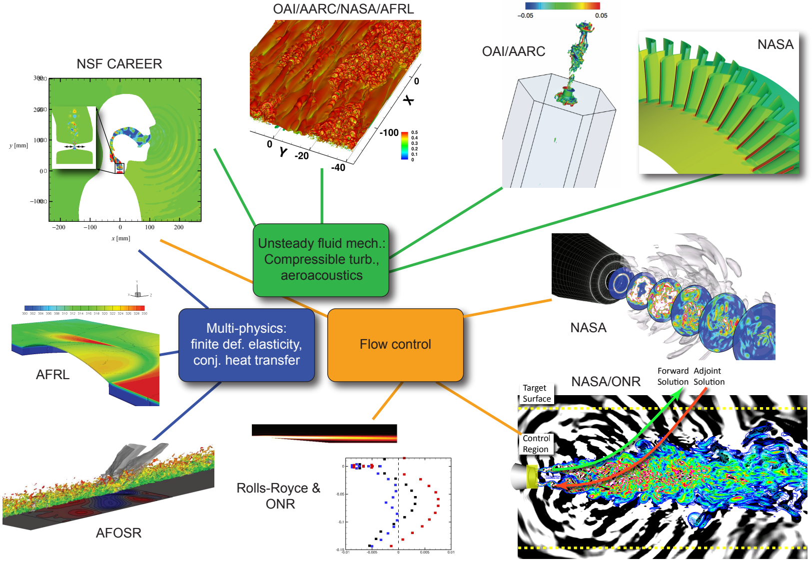
Acknowledgements

Dr. Scott Murman

Office of Naval Research
Dr. Joseph Doychak

In collaboration with Professor Carlos Pantano (MechSE) and Jonathan Freund (MechSE/AE)

Motivation (1/2)



Accurate and Stable Numerical Methods

Massive Parallelism on Exascale Computing Platforms (DOE)

Motivation (2/2)

Capabilities we want

- Simulate turbulent flows using LES or DNS
- Handle complex geometries
- Prediction and design
- Multiphysics (*e.g.*, thermoelastic)

Motivation (2/2)

Capabilities we want

- Simulate turbulent flows using LES or DNS
- Handle complex geometries
- Prediction and design
- Multiphysics (*e.g.*, thermoelastic)

Properties we want

- (Robust) – stability
- Variable order
- No artificial dissipation
- High FLOP performance ($>$ 30% peak)
- High scalability

Motivation (2/2)

Capabilities we want

- Simulate turbulent flows using LES or DNS
- Handle complex geometries
- Prediction and design
- Multiphysics (*e.g.*, thermoelastic)

Properties we want

- (Robust) – stability
- Variable order
- No artificial dissipation
- High FLOP performance ($>$ 30% peak)
- High scalability

Stencil-based finite differences on structured, overset meshes is a viable path forward but several improvements are needed

Outline

1. Background on basic flow solver
2. Stable methods for overset grids
3. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

4. Dual consistent, discrete adjoint

Flow Solver *MPSolve*

- Developed at UIUC from 2006 with support from DOE, AFOSR, AFRL, ONR, NSF, OAI, NASA
- Solves the compressible Navier-Stokes equations

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} = S_\rho \tag{1}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij} - \tau_{ij}) = S_{\rho u_i} \tag{2}$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_j}([\rho E + p]u_j - u_i\tau_{ij} + q_j) = S_{\rho E} \tag{3}$$

in conservative form

$$\frac{\partial}{\partial t}\left(\frac{Q}{J}\right) + \frac{\partial(\hat{F}_i - \hat{F}_v)}{\partial \xi} + \frac{\partial(\hat{G}_i - \hat{G}_v)}{\partial \eta} + \frac{\partial(\hat{H}_i - \hat{H}_v)}{\partial \zeta} = \frac{Z}{J} \tag{4}$$

with additive Runge-Kutta methods.

Flow Solver *MPSolve* — Discretization

Summation-by-parts operators (Strand, 1994)

We approximate

$$\left.\frac{\partial u}{\partial x}\right|_{i=1,\ldots,n} \quad \text{by} \quad D\vec{u} = P^{-1}Q\vec{u}$$

where $P = P^T > 0$ and $Q + Q^T = \mathrm{diag}(-1, 0, \ldots, 0, 1)$.

Flow Solver *MPSolve* — Discretization

Summation-by-parts operators (Strand, 1994)

We approximate

$$\left.\frac{\partial u}{\partial x}\right|_{i=1,\ldots,n} \quad \text{by} \quad D\vec{u} = P^{-1}Q\vec{u}$$

where $P = P^T > 0$ and $Q + Q^T = \text{diag}(-1, 0, \ldots, 0, 1)$.

Simultaneous-approximation-term boundary conditions

We enforce all BCs through a penalization framework

$$\frac{\partial \vec{q}}{\partial t} = \underbrace{\mathcal{F}(\vec{q})}_{\text{original}} + \underbrace{\sigma^{I1} P^{-1} E_1 A^{+}(\vec{q} - \vec{g}^{I1})}_{\text{inviscid BC}} + \underbrace{\frac{\sigma^{I2}}{\text{Re}} P^{-1} E_1 I(\vec{q} - \vec{g}^{I2})}_{\text{viscous BC}}$$

where σ^{I1} and σ^{I2} are semi-bounded parameters.

Flow Solver *MPSolve* — Discretization

Summation-by-parts operators (Strand, 1994)

We approximate

$$\left.\frac{\partial u}{\partial x}\right|_{i=1,\ldots,n} \quad \text{by} \quad D\vec{u} = P^{-1}Q\vec{u}$$

where $P = P^T > 0$ and $Q + Q^T = \text{diag}(-1, 0, \ldots, 0, 1)$.

Simultaneous-approximation-term boundary conditions

We enforce all BCs through a penalization framework

$$\frac{\partial \vec{q}}{\partial t} = \underbrace{\mathcal{F}(\vec{q})}_{\text{original}} + \underbrace{\sigma^{I1} P^{-1} E_1 A^+ (\vec{q} - \vec{g}^{I1})}_{\text{inviscid BC}} + \underbrace{\frac{\sigma^{I2}}{\text{Re}} P^{-1} E_1 I (\vec{q} - \vec{g}^{I2})}_{\text{viscous BC}}$$

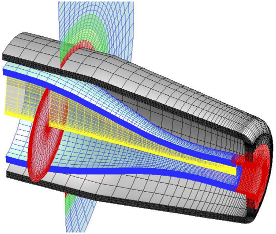
where σ^{I1} and σ^{I2} are semi-bounded parameters.

SBP + SAT yields provable stability for (linearized) CNS $\Longrightarrow$ robust(er)

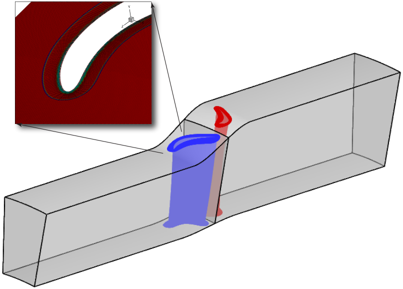
Flow Solver *MPSolve* — Overset & Interpolation

Overset (chimera)—overlapping meshes to describe a geometry

- individual meshes are logically structured
- meshes are assembled in a globally unstructured (PEGASUS, Ogen)
- Lagrange interpolation between meshes

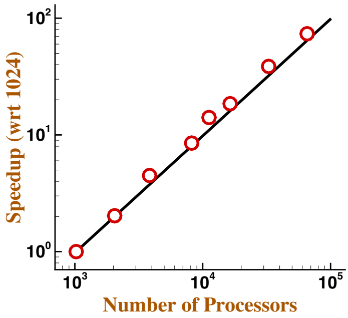


7-mesh solution of a Mach 1.3 nozzle

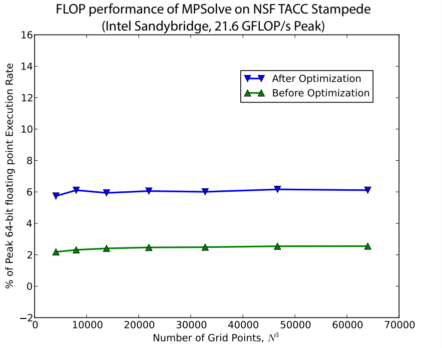


4-mesh solution of a LPT

Flow Solver *MPSolve* — Details & Performance

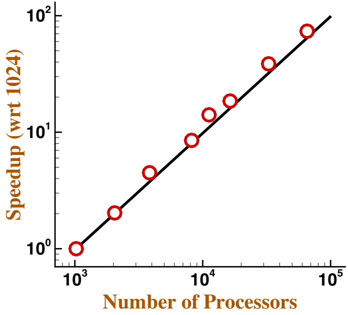


Strong scaling on DOE's Jaguar

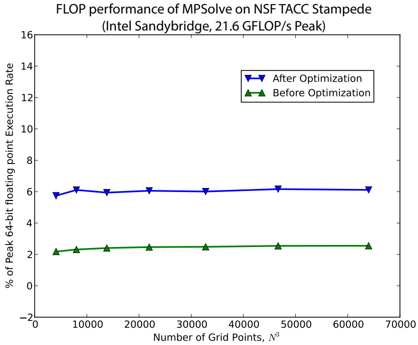


TAU-based performance measurement

Flow Solver *MPSolve* — Details & Performance



Strong scaling on DOE's Jaguar



TAU-based performance measurement

- Written in Fortran 90/C/C++
- Fully parallelized with MPI, including I/O
- Some core routines available on GPUs/accelerators
- ANSI-compliant with good portability

Outline

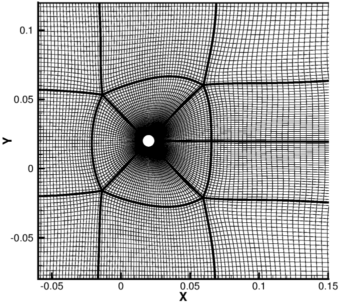
1. Background on basic flow solver
2. Stable methods for overset grids
3. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

4. Dual consistent, discrete adjoint

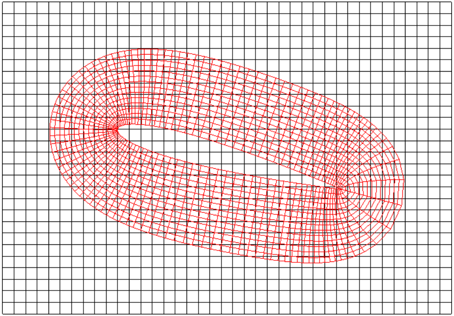
Overset

Multiblock Grid



Source: duns.sourceforge.net

Overset Grid



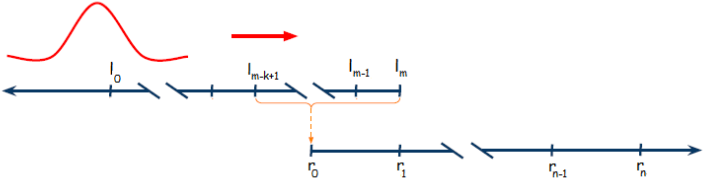
Source: http://celeritassimtech.com

- "Lots" of results for multiblock grids with matching nodes
- "Limited" results for multiblock grids without matching nodes
- Almost no results for overset grids that do not require dissipation

Existing stability for overset grids

- Pärt-Enander & Sjögreen (1994) – Euler equations with FV methods and artifical dissipation
- Scheme for electrodynamics by Henshaw (2006)—Godunov-Ryabenkii stable interface conditions for matching interfaces
- Godunov-Ryabenkii stable method for heat transfer in fluid structure systems by Henshaw & Chand (2009)
- G-K-S stable method by Appelö *et al.* (2012) for linear elasticity
- Energy stable methods for hyperbolic system using GSBP operators by Reichert *et al.* (2012)

Overset stability problem definition



Consider the hyperbolic system

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \quad \text{for } -1 \leq x \leq 1,\ t > 0$$

subject to the conditions that

$$u(x, 0) = f(x), \quad u(-1, t) = g(t) = 0.$$

Overset scheme

- Discretization for the left and right domain

$$\frac{d\mathbf{u}}{dt} = -P_L^{-1}Q_L\mathbf{u} - \tau_0 E_0^L(u_0 - g) + \gamma h_L R_L \mathbf{u},$$

$$\frac{d\mathbf{v}}{dt} = -P_R^{-1}Q_R\mathbf{v} - \tau_R E_0^R(v_0 - I_L^T \mathbf{u}),$$

$$E_0^{L,R} = H_{L,R}^{-1}[1, 0, ..., 0]^T.$$

- R_L denotes second derivative approximation at selected points (determined from stability analysis)

$$R_L = \frac{1}{h_L^2}\begin{bmatrix} 0 & .. & .. & .. & .. & 0 \\ . & .. & .. & .. & .. & . \\ 0 & .. & .. & .. & .. & 0 \\ 0 & .. & .. & 1 & -2 & 1 \\ 0 & .. & .. & .. & .. & 0 \end{bmatrix}$$

Overset scheme—stability and proof

The proof outline is as follows:

1. Want to show that

$$\frac{\mathrm{d}}{\mathrm{d}t}||\vec{w}||_H^2 \le 0, \quad \vec{w}^T = [\vec{u}^T\ \vec{v}^T]^T$$

   for some H.

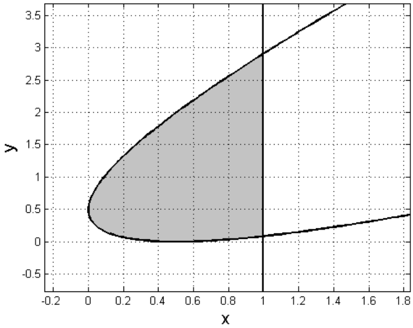
2. Set $H = \mathrm{diag}(H_L, H_R) = \mathrm{diag}(C_L P_L, \beta_R P_R)$ where C_L is diagonal matrix and β_R is a constant
3. Use SBP property of $\{P_L, Q_L\}$ and $\{P_R, Q_R\}$ to show that

$$\frac{\mathrm{d}}{\mathrm{d}t}||\vec{w}||_H^2 = \vec{z}^T K \vec{z}$$

   where K is a small square matrix (size depends on order)

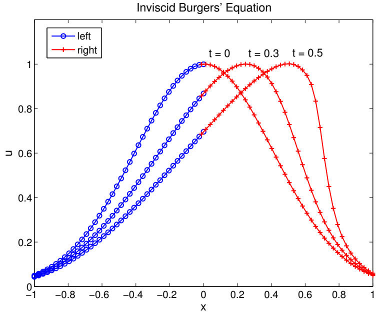
4. Ensure $K \le 0$ by determinant test

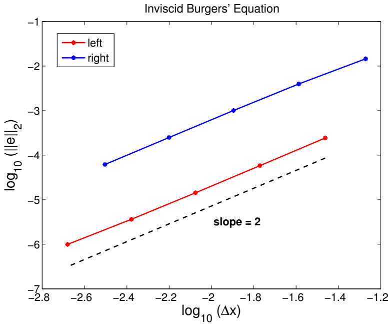
Overset scheme—stability and proof



- A range of parameters are possible to ensure stability (x and y are linear functions of the SAT paramters)
- Requiring conservation across the interface sets the values to be $\tau_R = 1$ and $\gamma = \alpha_L^2/[2(2-\alpha_L^2)]$ where $0 \leq \alpha_L \leq 1$ is the percentage of overlap.

Overset scheme—results for 2nd order system





Overset scheme—next steps

- Currently extending to systems of hyperbolic systems (easy)
- Incorporating diffusion (difficult, but we have preliminary results)

Outline

1. Background on basic flow solver
2. Stable methods for overset grids
3. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

4. Dual consistent, discrete adjoint

Variable coefficient 2nd derivatives

- The CNS equations contain several terms of the form

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

  that arise from
  - Viscous dissipation
  - Heat transfer
  - Coordinate mappings
- When using LES, then $a(x,t)$ could be a model coefficient
- When using wall modeled LES, then $a(x,t)$ also arises with a Neumann boundary condition (wall stress) on u

Variable coefficient 2nd derivatives

- The CNS equations contain several terms of the form

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

  that arise from
  - Viscous dissipation
  - Heat transfer
  - Coordinate mappings
- When using LES, then $a(x,t)$ could be a model coefficient
- When using wall modeled LES, then $a(x,t)$ also arises with a Neumann boundary condition (wall stress) on u

Need methods that ensure these terms are *stable*!

Variable coefficient 2nd derivatives—Dirichlet

We want to solve the variable-coefficient diffusion equation

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(a(x)\frac{\partial u}{\partial x}\right)$$

on the domain $x \in [0, 2\pi]$ with $u(0,t) = f_1(t)$ and $u(2\pi, t) = f_N(t)$. The semi-discrete formulation is

$$\frac{\mathrm{d}\vec{u}}{\mathrm{d}t} = P^{-1}M(a)\vec{u} - \frac{\tau}{h}P^{-1}A\vec{e}_1(u_1 - f_1) - \frac{\tau}{h}P^{-1}Ae_N(u_N - f_N)$$

where

$$P = h\,\mathrm{diag}(1/2, 1, \ldots, 1, 1/2), \quad A = \mathrm{diag}(a_1, a_2, \ldots, a_N)$$

and $\tau > 1/2$ for stability. The *magic* is in M.

Variable coefficient 2nd derivatives—Dirichlet

The matrix $M(a)$ has the following properties:

1. $$M(a)\vec{u} \approx P\left(\vec{a}.*\frac{\mathrm{d}^2\vec{u}}{\mathrm{d}x^2} + \frac{\mathrm{d}\vec{a}}{\mathrm{d}x}.*\frac{\mathrm{d}\vec{u}}{\mathrm{d}x}\right)$$
2. The boundary closures telecope:
   $$(M(a)\vec{u})_j = F_{j+1/2} - F_{j-1/2}$$
   where
   $$F_{j+1/2} = a\frac{\mathrm{d}\vec{u}}{\mathrm{d}x}\bigg|_{j+1/2} + \mathcal{O}(h^p)$$
3. $M(a)$ is definite to ensure
   $$\frac{\mathrm{d}}{\mathrm{d}t}||\vec{u}||_P^2 < 0$$
4. $M(a)$ preserves the SBP property
   $$\frac{\mathrm{d}}{\mathrm{d}t}\int u^2\,\mathrm{d}x = -\left[2au\frac{\partial u}{\partial x}\right]_{x=x_\ell}^{x_r} - 2\int a\left(\frac{\partial u}{\partial x}\right)^2\,\mathrm{d}x.$$

Variable coefficient 2nd derivatives—Dirichlet

Applying conditions 1–4 yields the matrix

$$M(a) = \frac{1}{h}\begin{bmatrix} \frac{1}{2}(a_1 - a_2) & \frac{1}{2}(a_2 - a_1) & 0 & 0 & \cdots & 0 \\ \frac{1}{2}(a_1 + a_2) & -\frac{1}{2}(a_1 + 2a_2 + a_3) & \frac{1}{2}(a_2 + a_3) & 0 & \cdots & 0 \\ & & & \ddots & & \\ 0 & \cdots & 0 & \frac{1}{2}(a_N + a_{N-1}) & -\frac{1}{2}(a_N + 2a_{N-1} + a_{N-2}) & \frac{1}{2}(a_{N-1} + a_{N-2}) \\ 0 & \cdots & 0 & 0 & \frac{1}{2}(a_{N-1} - a_N) & \frac{1}{2}(a_N - a_{N-1}) \end{bmatrix}$$

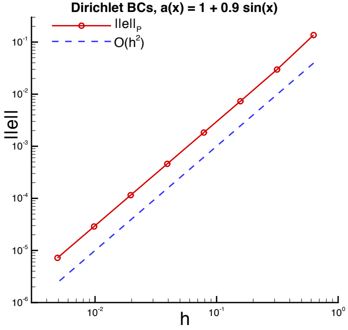
When $a \equiv$ constant then $M(a)$ reverts to the narrow stencil second derivative operator, with zeroes on the first and last rows.

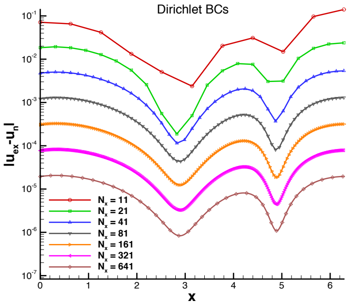
Variable coefficient 2nd derivatives—Dirichlet

Exact solution to

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(a(x)\frac{\partial u}{\partial x}\right) + F(x,t), \quad a(x) = 1 + \epsilon \sin x, \quad \epsilon = 0.9$$

with $F(x,t) = \epsilon \exp\{-t\} \sin 2x$ is $u(x,t) = \exp\{-t\} \cos x$.





Variable coefficient 2nd derivatives—Neumann

For Neumann boundary conditions, the method-of-lines discretization is

$$\frac{\mathrm{d}\vec{u}}{\mathrm{d}t} = P^{-1}M(a)\vec{u} - \tau P^{-1}A\vec{e}_1((Su)_1 - f_1) + \tau P^{-1}Ae_N((Su)_N - f_N)$$

with

$$(Su)_1 = -\frac{1}{3}u_1 + \frac{1}{3}u_2, \quad \text{and} \quad (Su)_N = -\frac{1}{3}u_{N-3} + \frac{1}{3}u_N$$

and $\tau \geq 3/2$ for stability. The matrix $M(a)$ is the same.

Outline

1. Background on basic flow solver
2. Stable methods for overset grids
3. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

4. Dual consistent, discrete adjoint

Adjoint Introduction—Optimization

To

$$\min_f \mathcal{J}(Q, f), \quad \text{subject to} \quad \mathcal{N}(Q) = f,$$

you can either *brute force* compute the variation

$$\delta\mathcal{J} = \left(\frac{\partial \mathcal{J}}{\partial Q}\right) \delta Q + \left(\frac{\partial \mathcal{J}}{\partial f}\right) \delta f$$

Adjoint Introduction—Optimization

To

$$\min_f \mathcal{J}(Q, f), \quad \text{subject to} \quad \mathcal{N}(Q) = f,$$

you can either *brute force* compute the variation

$$\delta\mathcal{J} = \left(\frac{\partial\mathcal{J}}{\partial Q}\right)\delta Q + \left(\frac{\partial\mathcal{J}}{\partial f}\right)\delta f$$

or you can use a calculus of variation result that the variation

$$\delta\mathcal{J} = \left[\left(\frac{\partial\mathcal{J}}{\partial f}\right) - Q^\dagger\left(\frac{\partial\mathcal{M}}{\partial f}\right)\right]\delta f$$

where

$$\mathcal{M}(Q, f) = \mathcal{N}(Q) - f \equiv 0$$

and where $Q^\dagger$ is the adjoint. Note that δQ is not needed in the second form. $Q^\dagger$ satisfies a CNS-like PDE with (x, t) dependent coefficients.

Why A Discrete Adjoint?

The efficiency of the optimization depends on the accuracy of $\delta\mathcal{J}$, which depends on $Q^\dagger$.

Why A Discrete Adjoint?

The efficiency of the optimization depends on the accuracy of $\delta\mathcal{J}$, which depends on $Q^\dagger$.

There are main paths to computing $Q^\dagger$:

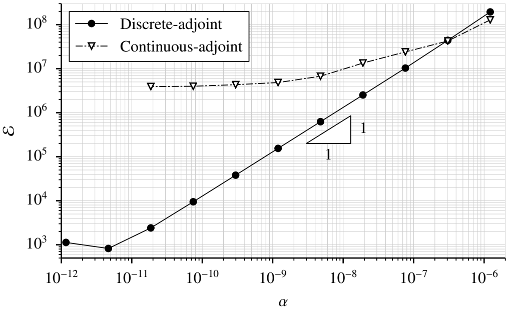
- *continuous adjoint*: PDE $\rightarrow$ Adjoint PDE $\rightarrow$ discretize
- *discrete adjoint*: PDE $\rightarrow$ discretize $\rightarrow$ transpose

Why A Discrete Adjoint?

The efficiency of the optimization depends on the accuracy of $\delta\mathcal{J}$, which depends on $Q^\dagger$.

There are main paths to computing $Q^\dagger$:

- *continuous adjoint*: PDE $\rightarrow$ Adjoint PDE $\rightarrow$ discretize
- *discrete adjoint*: PDE $\rightarrow$ discretize $\rightarrow$ transpose

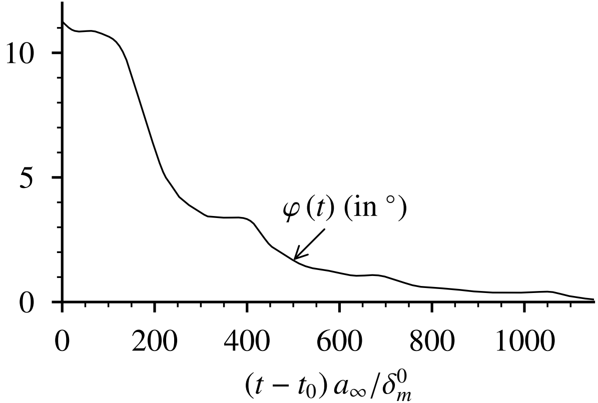


$$\mathcal{E} = \left| \frac{\mathcal{J}[Q, f + \delta f] - \mathcal{J}[Q, f]}{\alpha} + ||\mathcal{G}[Q^\dagger]||^2 \right|$$

Errors in Gradient Can Find Different Minima

The direction between $Q^\dagger_{\text{disc}}$ and $Q^\dagger_{\text{cont}}$ is different:

$$\varphi(t) = \cos^{-1}\left(\frac{\langle Q^\dagger_{\text{disc}}, Q^\dagger_{\text{cont}}\rangle}{||Q^\dagger_{\text{disc}}||\,||Q^\dagger_{\text{cont}}||}\right)$$



Discrete Adjoint for SBP-SAT

We have developed a discrete adjoint for the *fully discretized* CNS using SBP-SAT and RK4, focusing on the efficiency of the implementation. For example, the forward and adjoint RK4 work out to be

$$
\begin{aligned}
\vec{\mathsf{M}}^{n,1} &= \frac{2\vec{Q}^{n,1} - 2\vec{Q}^{n-1,4}}{\Delta t} - \vec{\mathsf{R}}^{n-1,4} - \vec{F}_{\Gamma}^{n,1} \\
\vec{\mathsf{M}}^{n,2} &= \frac{2\vec{Q}^{n,2} - 2\vec{Q}^{n-1,4}}{\Delta t} - \vec{\mathsf{R}}^{n,1} - \vec{F}_{\Gamma}^{n,2} \\
\vec{\mathsf{M}}^{n,3} &= \frac{\vec{Q}^{n,3} - \vec{Q}^{n-1,4}}{\Delta t} - \vec{\mathsf{R}}^{n,2} - \vec{F}_{\Gamma}^{n,3} \\
\vec{\mathsf{M}}^{n,4} &= \frac{6\vec{Q}^{n,4} + 2\vec{Q}^{n-1,4} - 2\vec{Q}^{n,1} - 4\vec{Q}^{n,2} - 2\vec{Q}^{n,3}}{\Delta t} - \vec{\mathsf{R}}^{n,3} - \vec{F}_{\Gamma}^{n,4},
\end{aligned}
$$

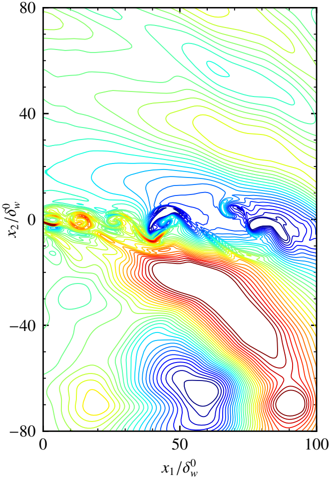
Forward

$$
\begin{aligned}
\vec{\mathsf{M}}^{\dagger n,4} &= \frac{6\beta^{n,4}\vec{Q}^{\dagger n,4} + 2\beta^{n+1,4}\vec{Q}^{\dagger n+1,4} - \beta^{n+1,3}\vec{Q}^{\dagger n+1,3} - 2\beta^{n+1,2}\vec{Q}^{\dagger n+1,2} - 2\beta^{n+1,1}\vec{Q}^{\dagger n+1,1}}{\beta^{n,4}\Delta t} \\
&\qquad - \frac{\beta^{n+1,1}}{\beta^{n,4}}\mathbf{P}^{-1}\mathbf{T}^{n,4}\mathbf{P}\vec{Q}^{\dagger n+1,1} - \vec{G}_{\Omega}^{\dagger n,4} \\
\vec{\mathsf{M}}^{\dagger n,3} &= \frac{\beta^{n,3}\vec{Q}^{\dagger n,3} - 2\beta^{n,4}\vec{Q}^{\dagger n,4}}{\beta^{n,3}\Delta t} - \frac{\beta^{n,4}}{\beta^{n,3}}\mathbf{P}^{-1}\mathbf{T}^{n,3}\mathbf{P}\vec{Q}^{\dagger n,4} - \vec{G}_{\Omega}^{\dagger n,3} \\
\vec{\mathsf{M}}^{\dagger n,2} &= \frac{2\beta^{n,2}\vec{Q}^{\dagger n,2} - 4\beta^{n,4}\vec{Q}^{\dagger n,4}}{\beta^{n,2}\Delta t} - \frac{\beta^{n,3}}{\beta^{n,2}}\mathbf{P}^{-1}\mathbf{T}^{n,2}\mathbf{P}\vec{Q}^{\dagger n,3} - \vec{G}_{\Omega}^{\dagger n,2} \\
\vec{\mathsf{M}}^{\dagger n,1} &= \frac{2\beta^{n,1}\vec{Q}^{\dagger n,1} - 2\beta^{n,4}\vec{Q}^{\dagger n,4}}{\beta^{n,1}\Delta t} - \frac{\beta^{n,2}}{\beta^{n,1}}\mathbf{P}^{-1}\mathbf{T}^{n,1}\mathbf{P}\vec{Q}^{\dagger n,2} - \vec{G}_{\Omega}^{\dagger n,1}
\end{aligned}
$$

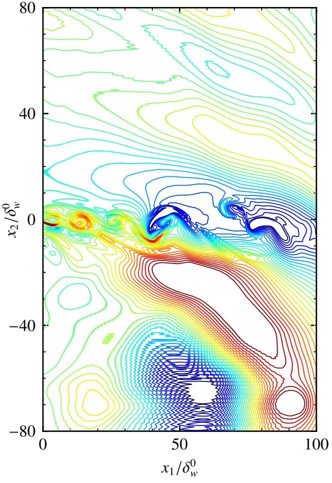
Adjoint

Note: a semi-discrete adjoint which uses RK4 for backward time advancement has (x, t) coefficients that are evaluated at different times than the fully discrete, and different initial conditions.

Discrete Adjoint for SBP-SAT



(a)

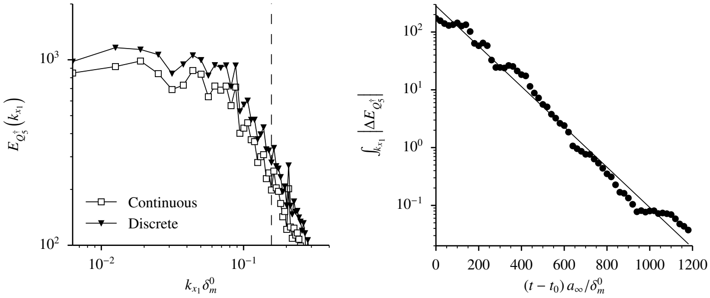


(b)

continuous

discrete

Implications



(a)

(b)

- The unsteady coefficients in the adjoint PDE cause *scattering* of the errors to *low frequency* components $\Longrightarrow$ conclusions based on continuous adjoint may differ from those based on discrete adjoint.
- The error grows exponentially in time for chaotic systems (e.g., turbulent)

Discrete Dual-Consistent Adjoint for SBP-SAT

The discrete adjoint *is exact* but it is unsatisfactory (inconsistent) because, in general, it is a low-order approximate discretization of the continuous adjoint PDE.

Discrete Dual-Consistent Adjoint for SBP-SAT

The discrete adjoint *is exact* but it is unsatisfactory (inconsistent) because, in general, it is a low-order approximate discretization of the continuous adjoint PDE.

Why?

Discrete Dual-Consistent Adjoint for SBP-SAT

The discrete adjoint *is exact* but it is unsatisfactory (inconsistent) because, in general, it is a low-order approximate discretization of the continuous adjoint PDE.

Why?

A primary cause for the inconsistency is:

$$(D\vec{u})_j = \left.\frac{\partial u}{\partial x}\right|_j + \mathcal{O}(h^q) \;\not\Rightarrow\; (D^T\vec{u})_j = \left.\frac{\partial u}{\partial x}\right|_j + \mathcal{O}(h^q),$$

which occurs *at the boundaries* for explicit finite difference schemes.

Discrete Dual-Consistent Adjoint for SBP-SAT

Hicken & Zingg (2013) worked out a dual-consistent adjoint for the Euler equations. We have extended this to the CNS equations with temperature-dependent viscosities. The key ingredients are:

1. Discretize all spatial derivatives using standard SBP operators ($D_{2x} \approx D_x^2$)
2. Form the bilinear concomitant

$$\langle \mathcal{L}[Q]\delta Q, Q^\dagger \rangle = \langle \delta Q, \mathcal{L}^\dagger[Q] Q^\dagger \rangle + b(Q, \delta Q, Q^\dagger)$$

   where b contains all of the boundary terms from $D^T \neq D$ and SAT
3. Require $b \equiv 0$ by defining the splitting b into forward and adjoint SAT BCs
4. Check well-posedness of resulting BCs (can show that if SAT BCs are used for the forward equations, then resulting BCs for adjoint are well-posed)

Discrete Dual-Consistent Adjoint for SBP-SAT

Start with the governing equation:

$$\frac{d\vec{Q}}{dt} + \mathbf{J}\mathbf{D}_i\vec{F}_i + \vec{\mathsf{N}}_{\mathrm{SAT}}\left[\vec{Q}\right] + \vec{\mathsf{N}}_s\left[\vec{Q}\right] = -\sigma_d\mathbf{J}\mathbf{P}^{-1}\mathbf{D}_i^{(m)\mathrm{T}}\mathbf{D}_i^{(m)}\vec{Q},$$

Linearize:

$$\frac{d}{dt}\left(\delta\vec{Q}\right) + \mathbf{J}\mathbf{D}_i\left\{\mathbf{A}_i\left[\vec{Q}\right]\delta\vec{Q} - \mathbf{B}_{ij}\left[\vec{Q}\right]\mathbf{D}_j\left(\mathbf{C}\left[\vec{Q}\right]\delta\vec{Q}\right)\right\} + \mathbf{T}_{\mathrm{SAT}}\left[\vec{Q}\right]\delta\vec{Q} + \delta\vec{\mathsf{N}}_s\left[\vec{Q},\delta\vec{Q}\right] = -\sigma_d\mathbf{J}\mathbf{P}^{-1}\mathbf{D}_i^{(m)\mathrm{T}}\mathbf{D}_i^{(m)}\delta\vec{Q},$$

Define adjoint inner product:

$$\int_{t_0}^{t_1}\left\langle\vec{Q}^\dagger,\delta\vec{\mathsf{N}}\right\rangle_{\mathbf{P}}dt = \int_{t_0}^{t_1}\left\langle\vec{\mathsf{N}}^\dagger,\delta\vec{Q}\right\rangle_{\mathbf{P}}dt,$$

Identify adjoint

$$\vec{\mathsf{N}}^\dagger\left[\vec{Q},\vec{Q}^\dagger\right] = -\frac{d\vec{Q}^\dagger}{dt} - \mathbf{J}\left(\mathbf{A}_i^{\mathrm{T}}\left[\vec{Q}\right] + \mathbf{C}^{\mathrm{T}}\left[\vec{Q}\right]\mathbf{D}_j\mathbf{B}_{ij}^{\mathrm{T}}\left[\vec{Q}\right]\right)\mathbf{D}_i\vec{Q}^\dagger + \vec{\mathsf{N}}_{\mathrm{SAT}}^\dagger\left[\vec{Q},\vec{Q}^\dagger\right] + \vec{\mathsf{N}}_s^\dagger\left[\vec{Q},\vec{Q}^\dagger\right] + \sigma_d\mathbf{J}\mathbf{P}^{-1}\mathbf{D}_i^{(m)\mathrm{T}}\mathbf{D}_i^{(m)}\vec{Q}^\dagger,$$

where

$$\vec{\mathsf{N}}_{\mathrm{SAT}}^\dagger\left[\vec{Q},\vec{Q}^\dagger\right] = \mathbf{J}\mathbf{P}^{-1}\mathbf{T}_{\mathrm{SAT}}^\dagger\left[\vec{Q}\right]\vec{Q}^\dagger$$

and

$$\mathbf{T}_{\mathrm{SAT}}\left[\vec{Q}\right] + \mathbf{\Delta}_i\mathbf{P}\mathbf{A}_i\left[\vec{Q}\right] - \mathbf{\Delta}_i\mathbf{P}\mathbf{B}_{ij}\left[\vec{Q}\right]\mathbf{D}_j\mathbf{C}\left[\vec{Q}\right] = \left(\mathbf{T}_{\mathrm{SAT}}^\dagger\left[\vec{Q}\right] - \mathbf{\Delta}_j\mathbf{P}\mathbf{C}^{\mathrm{T}}\left[\vec{Q}\right]\mathbf{B}_{ij}^{\mathrm{T}}\left[\vec{Q}\right]\mathbf{D}_i\right)^{\mathrm{T}}.$$

The last equation defines the adjoint SAT in terms of the forward SAT required for dual consistency.

Discrete Dual-Consistent Adjoint for SBP-SAT

We have yet to:

1. Demonstrate expected accuracy of dual consistent adjoint
2. Demonstrate super-convergence of functionals

Summary

We have discussed the following three developments:

1. Stable methods for overset grids
2. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

3. Dual consistent, discrete adjoint

Summary

We have discussed the following three developments:

1. Stable methods for overset grids
2. Provably definite methods for

$$\frac{\partial}{\partial x_i}\left(a(x,t)\frac{\partial u}{\partial x_i}\right)$$

3. Dual consistent, discrete adjoint

We are currently assembling all three into a single, provably stable, overset grid flow solver capable of prediction and control of turbulent flows with discrete, dual consistent adjoint-based gradients.

Coordinates

Professor Daniel J. Bodony
Blue Waters Associate Professor
Department of Aerospace Engineering
University of Illinois at Urbana-Champaign
bodony@illinois.edu
http://acoustics.ae.illinois.edu

Thank you for your kind attention!