Challenges in Adjoint-Based Aerodynamic Design for Unsteady Flows

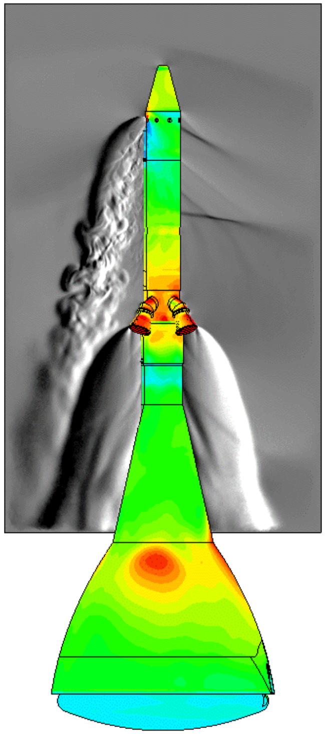
Eric J. Nielsen
FUN3D Development Team
NASA Langley Research Center
USA

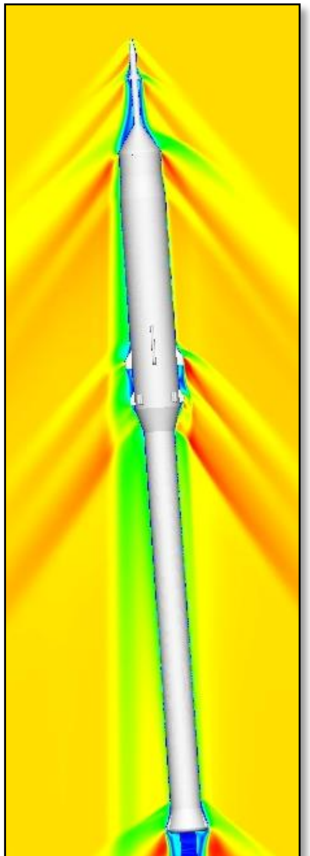
http://fun3d.larc.nasa.gov

Langley Research Center

It's A Multidisciplinary World

Ablation
Acoustics
Aerodynamics
Flight Dynamics
Materials
Optics
Performance
Propulsion
Radiation
Structures

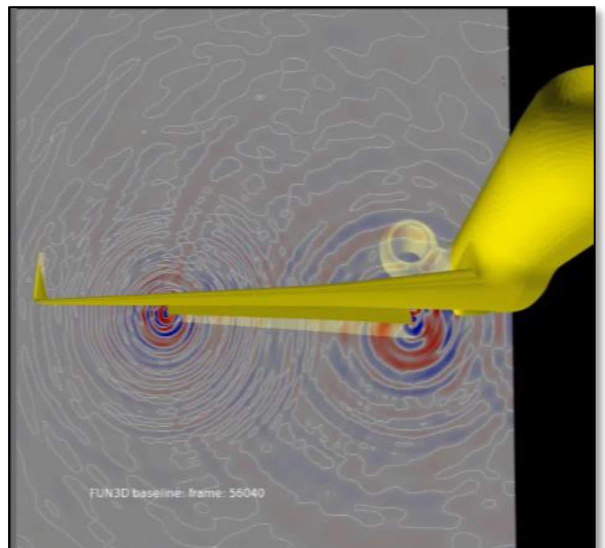


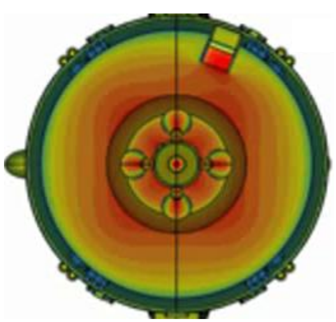


Ares 1-X

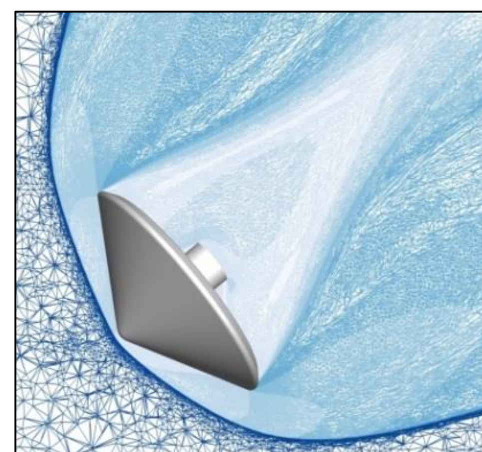
Mars InSight Lander

Gulfstream G550





NASA/Boeing Truss-Braced Wing



Inflatable Decelerators

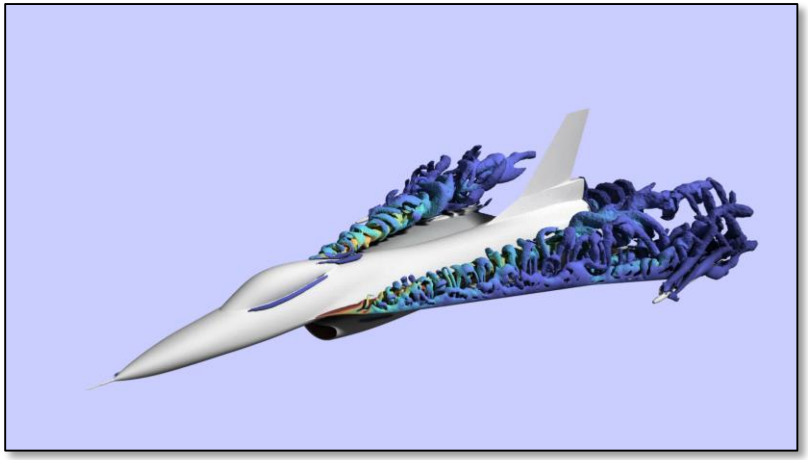
Physics-Based Aerodynamic Simulation

Euler Equations (inviscid)	Reynolds-averaged Navier-Stokes (turbulence modeled)	Large-Eddy Simulations (large scales resolved)	Direct Numerical Simulations (all scales resolved)

Increasing physics, increasing cost →

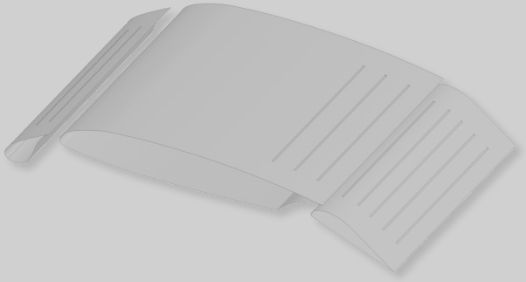
- Today's design approaches typically rely on Euler and RANS simulations, each requiring O(10^2)-O(10^3) CPU hours on moderate HPC resources
- Current projection is full aircraft LES as a grand challenge problem in the 2045 timeframe using an entire leadership-class machine, with DNS following in 2080*
- Hybrid RANS-LES simulations are the current state of the art and may require O(10^7) CPU hours on large HPC resources

Accurate predictions for many aerospace concepts require at least hybrid RANS-LES:
Pushing these methods into the design cycle is critical



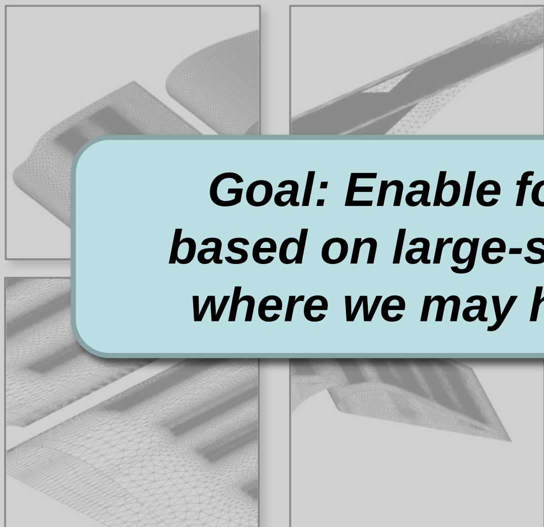
* Spalart, P., "Strategies for Turbulence Modeling and Simulations," *Int. J. Heat and Fluid Flow*, 21(3), 2000, pp. 252-263.

Physics-Based Aerodynamic Design

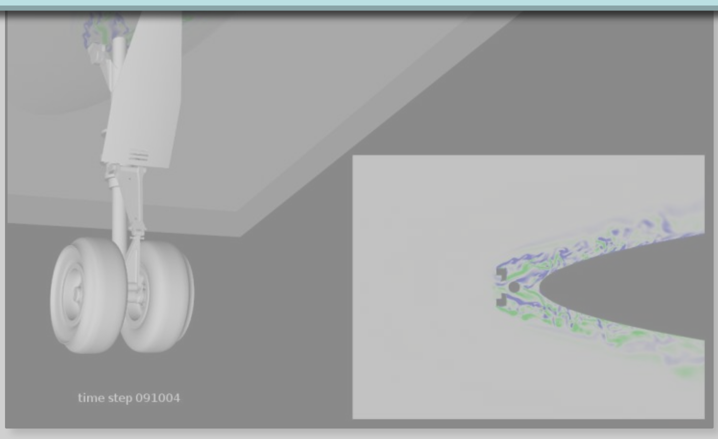


Biologically-Inspired UAV's

What are the optimal kinematics?
What is the optimal shape?



Goal: Enable formal, physics-based design optimization based on large-scale computational simulations of vehicles where we may have no a priori knowledge nor experience



Landing Gear Configuration

How can we minimize noise on approach?

Active Flow Control

Where should control jets be located?
At what orientation?
What should the unsteady blowing profiles look like?
What is the optimal phase difference between jets?
How should the outer mold line be altered?

…and what should my grid look like?

Design Approach

- Systematic design of a complete vehicle may involve thousands of design variables
- The number of function evaluations required by zeroth-order (e.g., sampling) optimization techniques increases dramatically with only a few design variables

→ **Gradient-based methods are the only feasible approach**

Bear in mind that we have not even touched on:

- ***Robust design optimization***
- ***Multidisciplinary optimization***
- ***Uncertainty quantification***
- ***…***

We are only at the tip of the iceberg!

Forward-Mode Sensitivity Analysis

- Conventional sensitivity analysis techniques such as finite differencing or direct differentiation consider a perturbation to a single input parameter
- This effect is then propagated through the simulation to ultimately determine a single element of the desired gradient vector
- This class of methods is referred to as *forward-mode differentiation*
- These methods can effectively provide sensitivities of many outputs with respect to a single input
- However, the cost of these approaches scales linearly with the number of design variables
  - E.g., for a problem with 1,000 variables, central differencing will require 2,000 (very accurate!) simulations just to obtain a single gradient vector

→ ***These approaches are prohibitively expensive in our context***

So how can we efficiently compute sensitivity information for thousands of simulation parameters?

Motivation for Adjoint Methods

- The adjoint approach flips the entire sensitivity analysis upside down by solving an auxiliary PDE and instead pushing the dependence on the number of design variables to the very end of the process
- In this manner, everything is done backwards; hence, adjoint methods are often referred to as *reverse-mode differentiation*

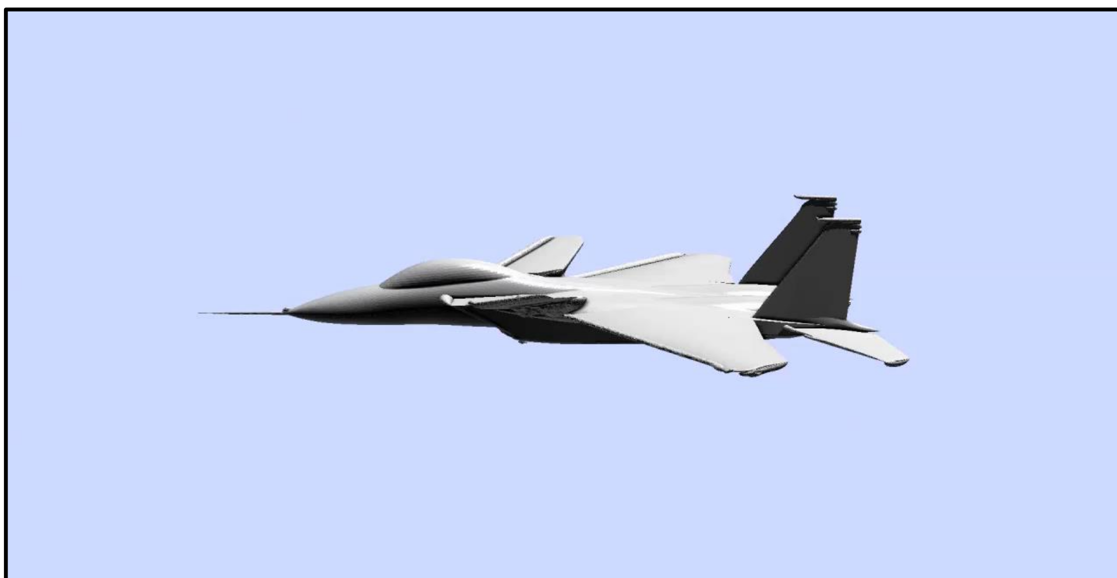
Adjoints can provide sensitivities of an output function for virtually unlimited numbers of input parameters
at the cost of a <u>single additional simulation</u>

Adjoint Solution Example

F-15 Configuration

Forward Solution

- Transonic turbulent flow over modified F-15 configuration
- Propulsion effects included as well as simulated aeroelastic deformations of canard/wing/h-tail
- Objective is lift-to-drag ratio
- Adjoint solution indicates where objective is sensitive to perturbations in both space and time



Reverse Solution

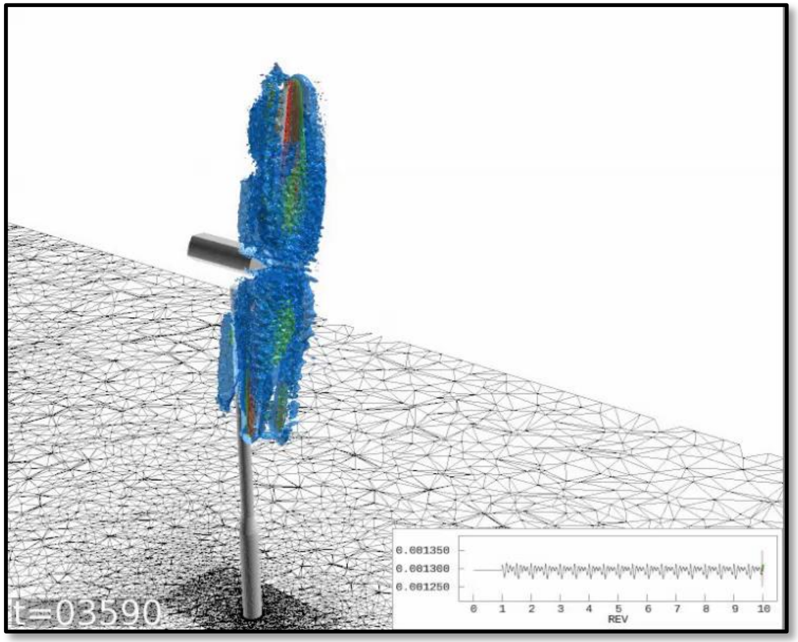
Adjoint Solution Example

Wind Turbine Configuration

- Incompressible turbulent flow over NREL Phase VI wind turbine
- Overset grids used to model rotating blade system
- Objective function is based on the torque

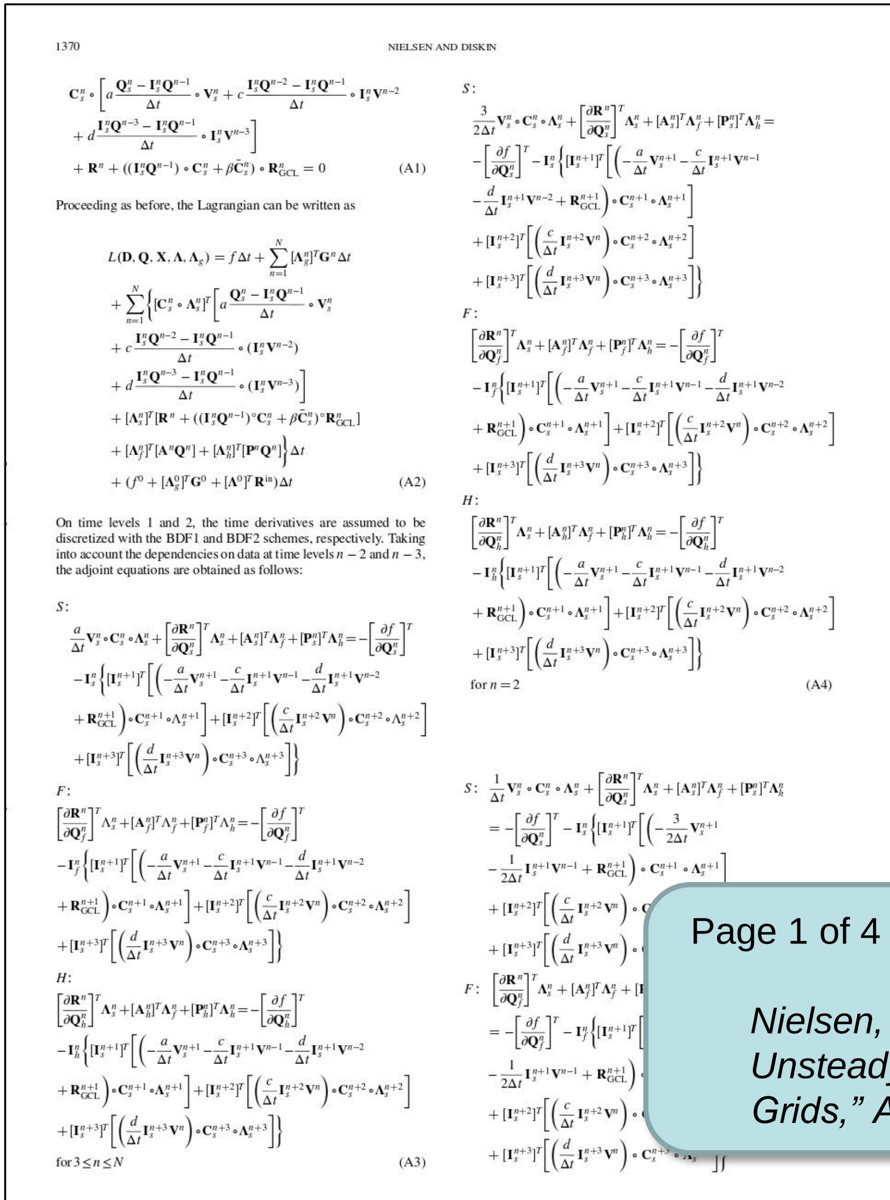
Forward Solution

Reverse Solution



Some Challenges

The Unsteady Adjoint Equations



$$\mathbf{C}_s^n \circ \left[a\frac{\mathbf{Q}_s^n - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ \mathbf{V}_s^n + c\frac{\mathbf{I}_s^n\mathbf{Q}^{n-2} - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ \mathbf{I}_s^n\mathbf{V}^{n-2} + d\frac{\mathbf{I}_s^n\mathbf{Q}^{n-3} - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ \mathbf{I}_s^n\mathbf{V}^{n-3} \right] + \mathbf{R}^n + ((\mathbf{I}_s^n\mathbf{Q}^{n-1}) \circ \mathbf{C}_s^n + \beta\bar{\mathbf{C}}_s^n) \circ \mathbf{R}_{\mathrm{GCL}}^n = 0 \tag{A1}$$

Proceeding as before, the Lagrangian can be written as

$$\begin{aligned} L(\mathbf{D},\mathbf{Q},\mathbf{X},\mathbf{\Lambda},\mathbf{\Lambda}_g) &= f\Delta t + \sum_{n=1}^{N}[\mathbf{\Lambda}_g^n]^T\mathbf{G}^n\Delta t \\ &+ \sum_{n=1}^{N}\Big\{[\mathbf{C}_s^n \circ \mathbf{\Lambda}_s^n]^T\Big[a\frac{\mathbf{Q}_s^n - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ \mathbf{V}_s^n \\ &+ c\frac{\mathbf{I}_s^n\mathbf{Q}^{n-2} - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ (\mathbf{I}_s^n\mathbf{V}^{n-2}) \\ &+ d\frac{\mathbf{I}_s^n\mathbf{Q}^{n-3} - \mathbf{I}_s^n\mathbf{Q}^{n-1}}{\Delta t} \circ (\mathbf{I}_s^n\mathbf{V}^{n-3})\Big] \\ &+ [\mathbf{\Lambda}_s^n]^T[\mathbf{R}^n + ((\mathbf{I}_s^n\mathbf{Q}^{n-1})\circ\mathbf{C}_s^n + \beta\bar{\mathbf{C}}_s^n)\circ\mathbf{R}_{\mathrm{GCL}}^n] \\ &+ [\mathbf{\Lambda}_f^n]^T[\mathbf{A}^n\mathbf{Q}^n] + [\mathbf{\Lambda}_h^n]^T[\mathbf{P}^n\mathbf{Q}^n]\Big\}\Delta t \\ &+ (f^0 + [\mathbf{\Lambda}_g^0]^T\mathbf{G}^0 + [\mathbf{\Lambda}^0]^T\mathbf{R}^{\mathrm{in}})\Delta t \end{aligned} \tag{A2}$$

On time levels 1 and 2, the time derivatives are assumed to be discretized with the BDF1 and BDF2 schemes, respectively. Taking into account the dependencies on data at time levels $n-2$ and $n-3$, the adjoint equations are obtained as follows:

$S:$

$$\begin{aligned} &\frac{a}{\Delta t}\mathbf{V}_s^n \circ \mathbf{C}_s^n \circ \mathbf{\Lambda}_s^n + \left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_s^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_s^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_s^n]^T\mathbf{\Lambda}_h^n = -\left[\frac{\partial f}{\partial\mathbf{Q}_s^n}\right]^T \\ &-\mathbf{I}_s^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} - \frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} \\ &+ \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] + [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

$F:$

$$\begin{aligned} &\left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_f^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_f^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_f^n]^T\mathbf{\Lambda}_h^n = -\left[\frac{\partial f}{\partial\mathbf{Q}_f^n}\right]^T \\ &-\mathbf{I}_f^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} - \frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} \\ &+ \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] + [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

$H:$

$$\begin{aligned} &\left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_h^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_h^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_h^n]^T\mathbf{\Lambda}_h^n = -\left[\frac{\partial f}{\partial\mathbf{Q}_h^n}\right]^T \\ &-\mathbf{I}_h^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} - \frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} \\ &+ \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] + [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

for $3 \le n \le N$ (A3)

$S:$

$$\begin{aligned} &\frac{3}{2\Delta t}\mathbf{V}_s^n \circ \mathbf{C}_s^n \circ \mathbf{\Lambda}_s^n + \left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_s^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_s^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_s^n]^T\mathbf{\Lambda}_h^n = \\ &-\left[\frac{\partial f}{\partial\mathbf{Q}_s^n}\right]^T - \mathbf{I}_s^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} \\ &-\frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} + \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] \\ &+ [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

$F:$

$$\begin{aligned} &\left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_f^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_f^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_f^n]^T\mathbf{\Lambda}_h^n = -\left[\frac{\partial f}{\partial\mathbf{Q}_f^n}\right]^T \\ &-\mathbf{I}_f^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} - \frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} \\ &+ \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] + [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

$H:$

$$\begin{aligned} &\left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_h^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_h^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_h^n]^T\mathbf{\Lambda}_h^n = -\left[\frac{\partial f}{\partial\mathbf{Q}_h^n}\right]^T \\ &-\mathbf{I}_h^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{a}{\Delta t}\mathbf{V}_s^{n+1} - \frac{c}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} - \frac{d}{\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-2} \\ &+ \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] + [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+2}\circ\mathbf{\Lambda}_s^{n+2}\Big] \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3}\circ\mathbf{\Lambda}_s^{n+3}\Big]\Big\} \end{aligned}$$

for $n = 2$ (A4)

$$\begin{aligned} S:\ &\frac{1}{\Delta t}\mathbf{V}_s^n \circ \mathbf{C}_s^n \circ \mathbf{\Lambda}_s^n + \left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_s^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_s^n]^T\mathbf{\Lambda}_f^n + [\mathbf{P}_s^n]^T\mathbf{\Lambda}_h^n \\ &= -\left[\frac{\partial f}{\partial\mathbf{Q}_s^n}\right]^T - \mathbf{I}_s^n\Big\{[\mathbf{I}_s^{n+1}]^T\Big[\Big(-\frac{3}{2\Delta t}\mathbf{V}_s^{n+1} \\ &-\frac{1}{2\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} + \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big)\circ\mathbf{C}_s^{n+1}\circ\mathbf{\Lambda}_s^{n+1}\Big] \\ &+ [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ \ldots \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ \ldots \\ F:\ &\left[\frac{\partial\mathbf{R}^n}{\partial\mathbf{Q}_f^n}\right]^T\mathbf{\Lambda}_s^n + [\mathbf{A}_f^n]^T\mathbf{\Lambda}_f^n + \ldots \\ &= -\left[\frac{\partial f}{\partial\mathbf{Q}_f^n}\right]^T - \mathbf{I}_f^n\Big\{[\mathbf{I}_s^{n+1}]^T \ldots \\ &-\frac{1}{2\Delta t}\mathbf{I}_s^{n+1}\mathbf{V}^{n-1} + \mathbf{R}_{\mathrm{GCL}}^{n+1}\Big) \ldots \\ &+ [\mathbf{I}_s^{n+2}]^T\Big[\Big(\frac{c}{\Delta t}\mathbf{I}_s^{n+2}\mathbf{V}^n\Big)\circ \ldots \\ &+ [\mathbf{I}_s^{n+3}]^T\Big[\Big(\frac{d}{\Delta t}\mathbf{I}_s^{n+3}\mathbf{V}^n\Big)\circ\mathbf{C}_s^{n+3} \ldots \end{aligned}$$

Complexity

- Considerably more involved than the Navier-Stokes equations
- Every line of the baseline code must be differentiated with respect to flow solution, grid coordinates, and design variables
- Tremendous amount of software infrastructure required
- Implemented by hand and verified using complex variables

Sheer Expense

- Full linearizations must be evaluated at every time step

Page 1 of 4 of the adjoint equations derived and implemented in:

Nielsen, E.J., and Diskin, B., "Discrete Adjoint-Based Design for Unsteady Turbulent Flows on Dynamic Overset Unstructured Grids," AIAA Journal, Vol. 51, No. 6, June 2013.

Big Data

Since the adjoint equations must be integrated backwards in time, we must have the forward solution available at every time plane

Possible Approaches

- *Brute force*: Store the entire forward solution
- *Recompute*: Store the forward solution periodically and recompute intermediate time steps as needed
- *Approximate*: Store the forward solution periodically and interpolate intermediate time planes somehow

Big Data

In FUN3D, we store all of the forward data to disk

- The amount of data adds up fast – consider a small example:
  - 50,000,000 grid points and 10,000 physical time steps
  - Assume a 1-equation turbulence model (5+1 unknowns per grid point)
  - Dynamic grids (3 additional unknowns per grid point)

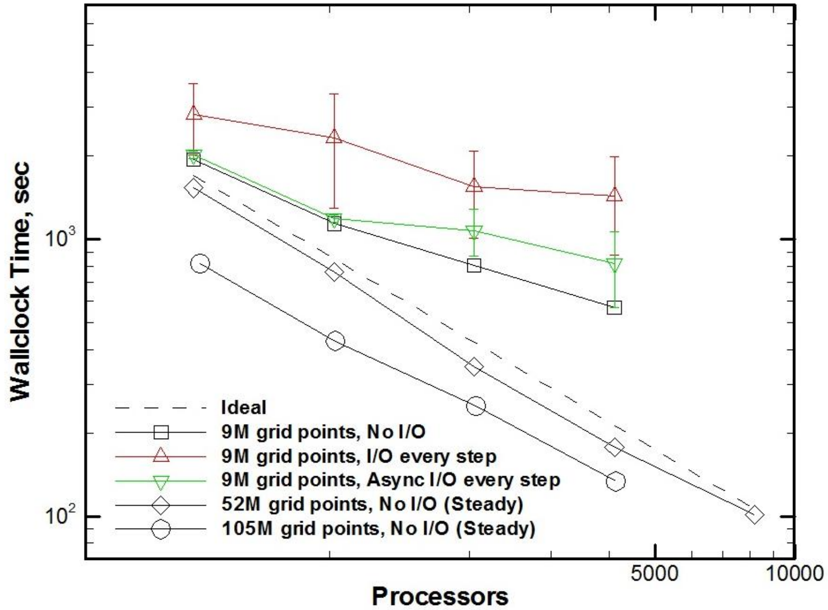
    → 50,000,000 x 10,000 x (6+3) x 8 bytes = ***36 Terabytes***

- This amount of data is not prohibitively large, but we need to run much bigger problems, say 10^9 grid points with 10^6 time steps

- So far, the challenge has been efficiently getting the data to/from the disk at every time step

Big Data

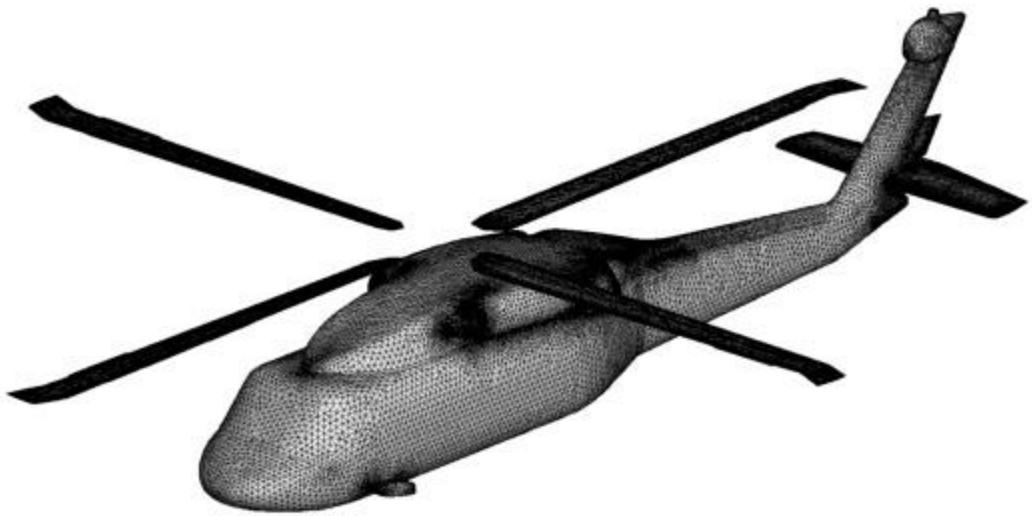
- Approaches used to write conventional checkpoint files are prohibitively expensive
- FUN3D uses parallel, asynchronous, direct access read/writes from every rank
  - Flow solver is writing the previous time plane while the current time step is computing
  - Adjoint solver is pre-fetching earlier time planes while the current time step is computing
- This strategy has performed well so far, but is not infinitely scalable



Application Examples

Example: UH-60A Blackhawk

Overview



- Such simulations are tremendously complex; here we are only doing aero
- Overset grid system consists of 9,262,941 nodes / 54,642,499 tetrahedra
- Compressible RANS: M_{tip}=0.64, Re_{tip}=7.3M, μ=0.37, α=0.0º
- Blade pitch has child motion governed by pilot collective and cyclic controls:

$$\theta = \theta_c + \theta_{1c}\cos\psi + \theta_{1s}\sin\psi$$

Blade pitch

Collective

Lateral cyclic

Longitudinal cyclic

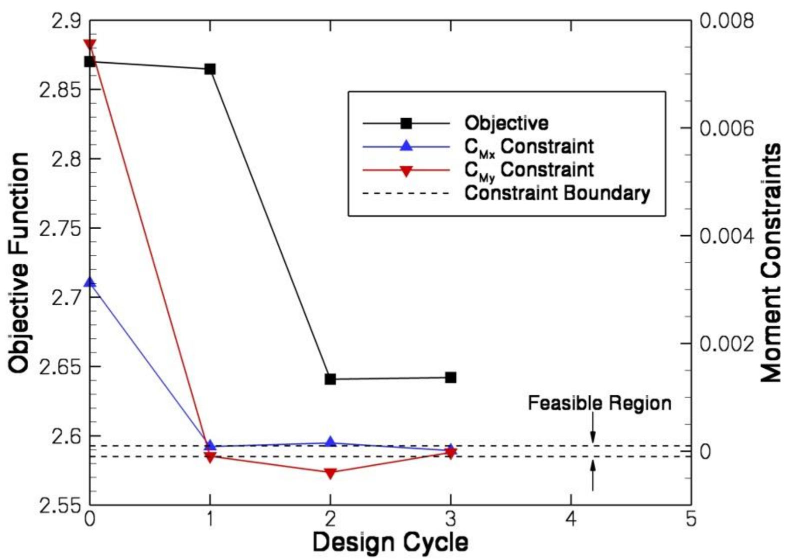
Example: UH-60A Blackhawk

Problem Definition and Results

- Objective is to maximize time-averaged lift while satisfying trim constraints:

$$\min f = \left[\left(\frac{1}{360} \sum_{n=361}^{720} C_L^n \right) - 2.0 \right]^2 \Delta t \quad \text{such that} \quad \begin{aligned} g_1 &= \frac{1}{360} \sum_{n=361}^{720} C_{M_x}^n \Delta t = 0 \\ g_2 &= \frac{1}{360} \sum_{n=361}^{720} C_{M_y}^n \Delta t = 0 \end{aligned}$$

- Separate adjoint solutions required for all three functions
- 67 design variables include 64 thickness and camber variables across the blade planform, plus collective and cyclic control inputs



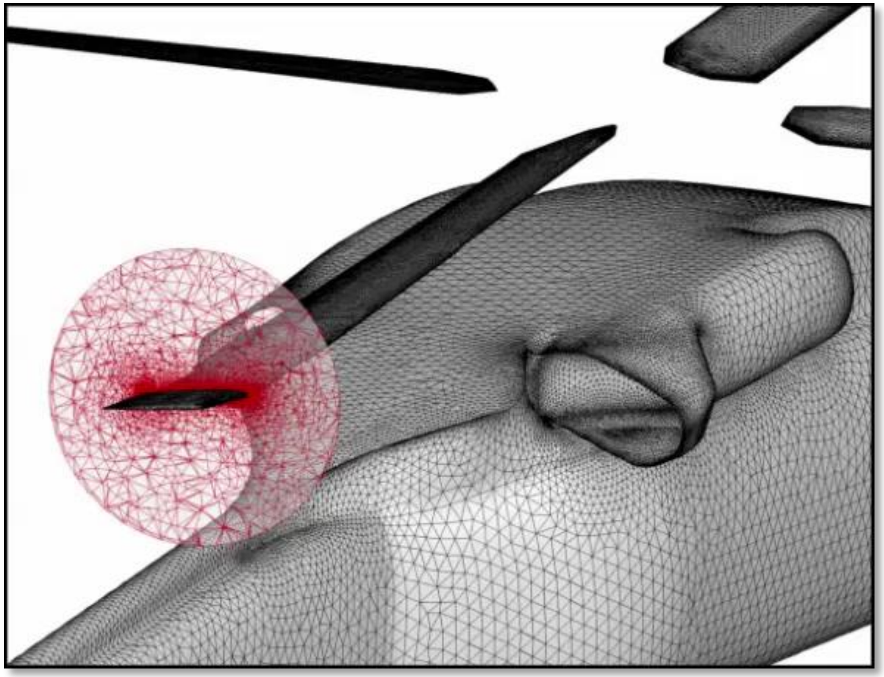
	$\bar{C}_L$	Flow Solves (2 hrs)	Adjoint Solves (3 hrs)	Total Time
Baseline	0.023	-	-	-
Design	0.103	4	4	0.8 days (38,400 CPU hrs)

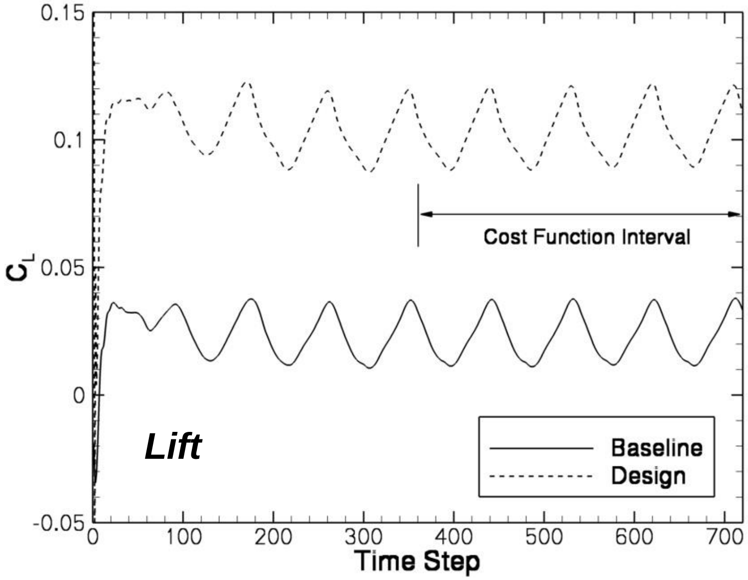
- Feasible region is quickly located
- Both moment constraints are satisfied within tolerance at the optimal solution
- Final controls: θ_c=6.71º, θ_{1c}=2.58º, θ_{1s}=-7.00º

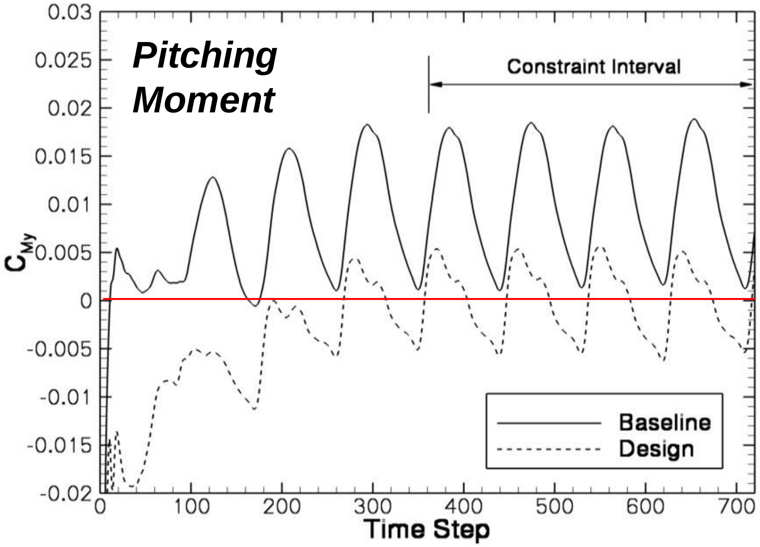
Example: UH-60A Blackhawk

Results

Lift has gone up significantly; vehicle is trimmed in both pitch and roll

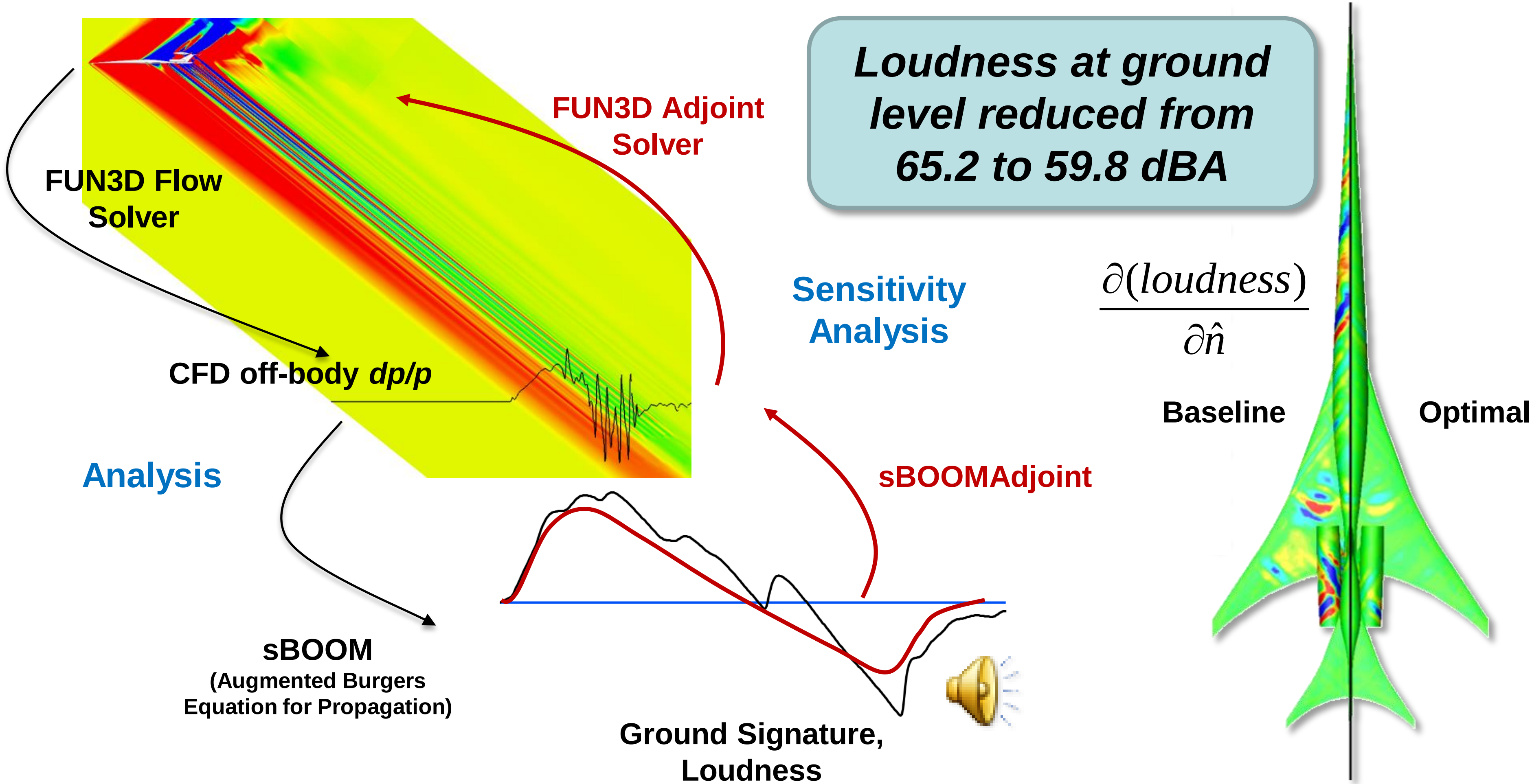






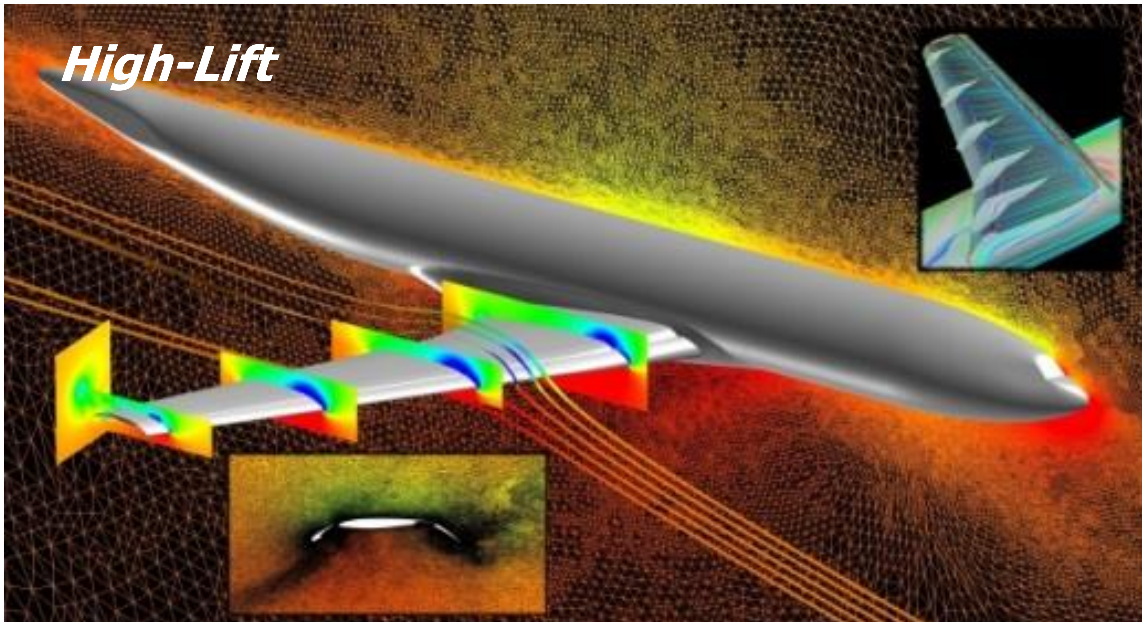
Multidisciplinary Example
Sonic Boom Mitigation

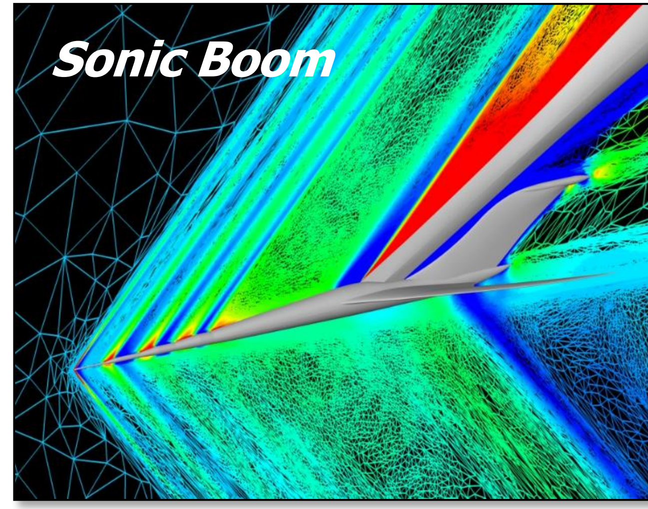
- Multidisciplinary adjoint has been very successful for sonic boom mitigation – discrete sensitivities of ground-based metrics to aircraft geometry
- Recently extended to include atmospheric UQ
- Many other disciplines being considered / pursued

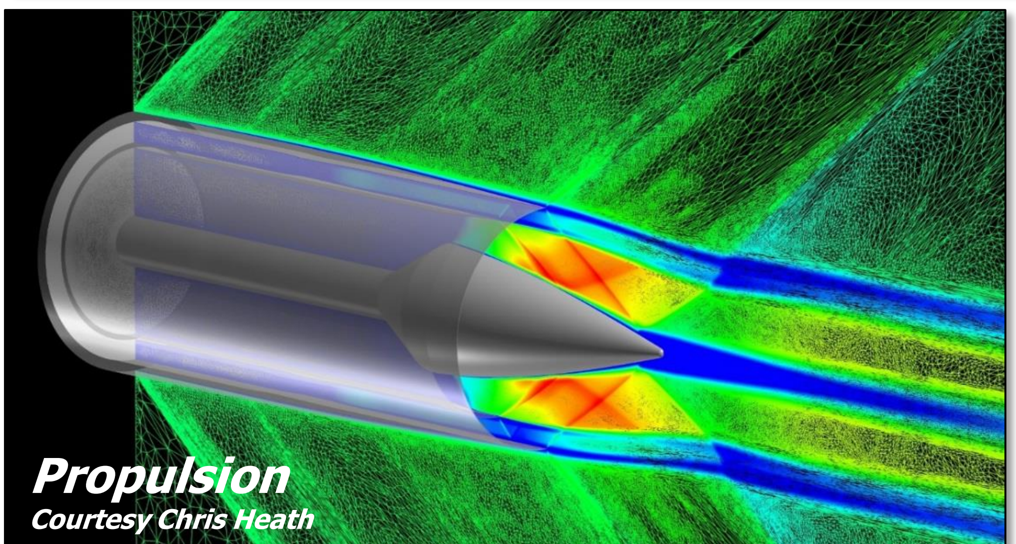


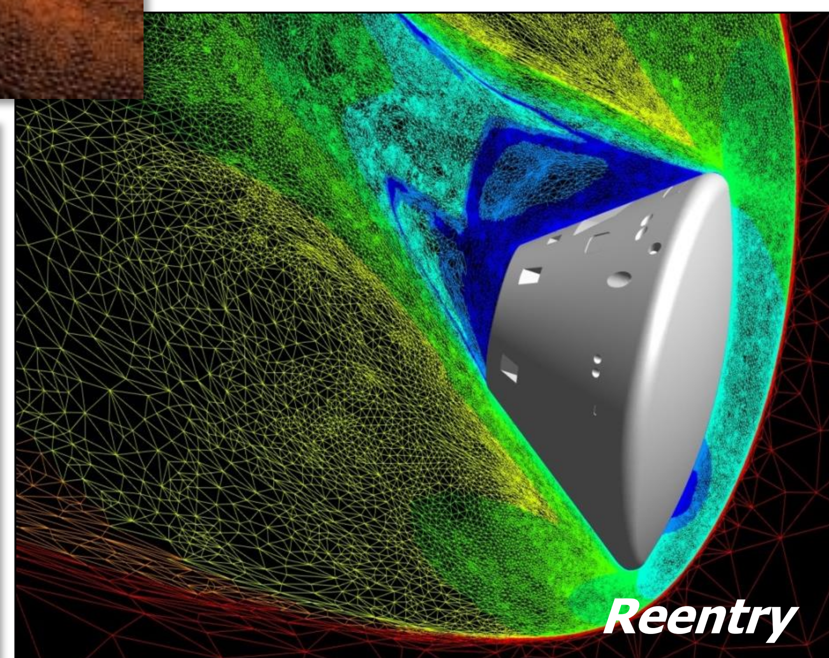
Langley Research Center

Mesh Adaptation Examples









A Remaining Challenge: Chaos

Wish to compute sensitivities of infinite time averages for chaotic flows (Hybrid RANS-LES, LES, DNS)

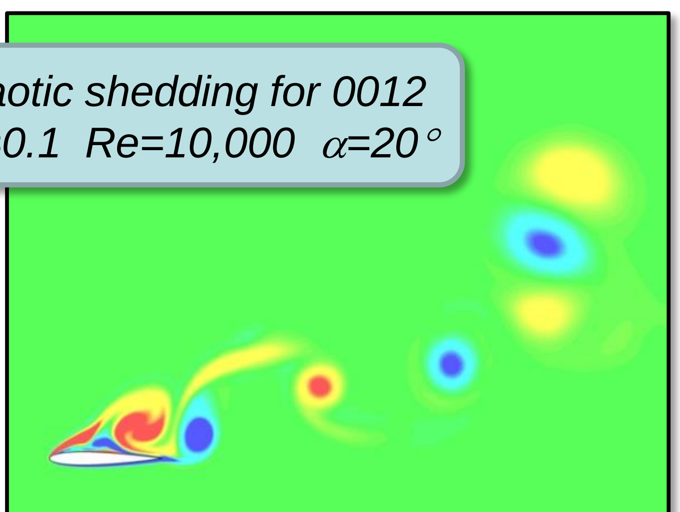
- Theory states these sensitivities are well-defined and bounded

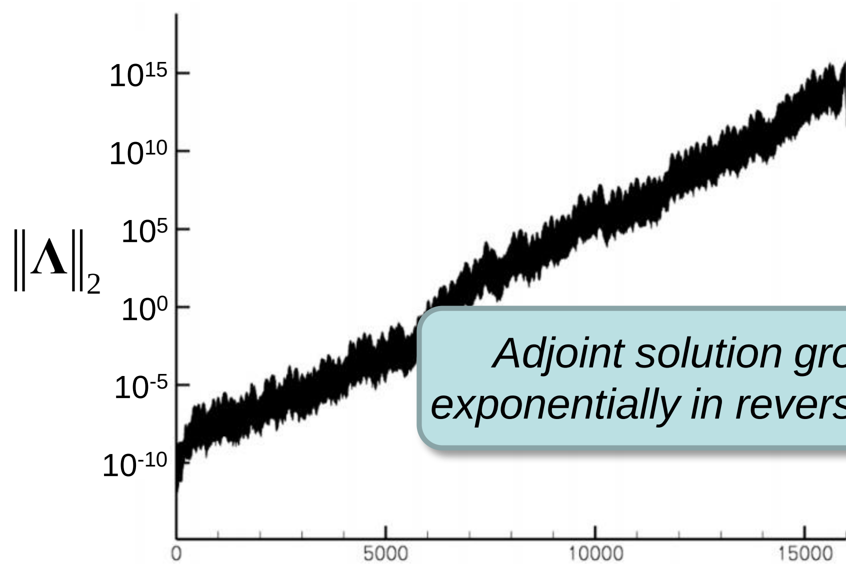
Why does conventional approach not work?

For chaotic flows,

- The finite time average approaches the infinite time average
- The sensitivity for a finite time average does not approach the sensitivity for the infinite time average

Chaotic shedding for 0012
M_∞=0.1 Re=10,000 α=20°





Adjoint solution grows exponentially in reverse time

A Remaining Challenge: Chaos

Least-Squares Shadowing (MIT)

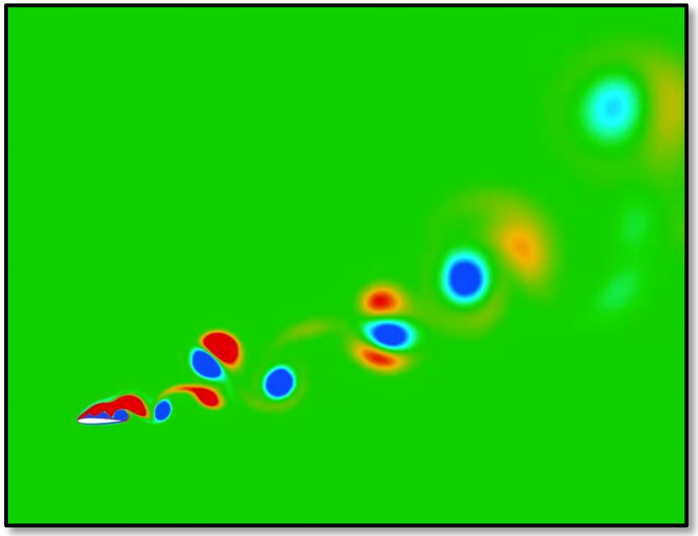
- Least-Squares Shadowing (LSS) method proposed by Wang and Blonigan (MIT)
  - Key assumption is ergodicity of the simulation: long time averages are essentially independent of the initial conditions
  - Also assumes existence of a shadowing trajectory
- The LSS formulation involves a linearly-constrained least squares optimization problem which results in a set of optimality equations
- **The LSS adjoint equations are a globally coupled system in space-time**
- To date, work at MIT has focused on solutions of this system for academic dynamical systems containing O(1) state variables
- Langley and MIT are collaborating to explore the extension to CFD systems: *enormous* computational challenge for even the smallest of problems

A Remaining Challenge: Chaos

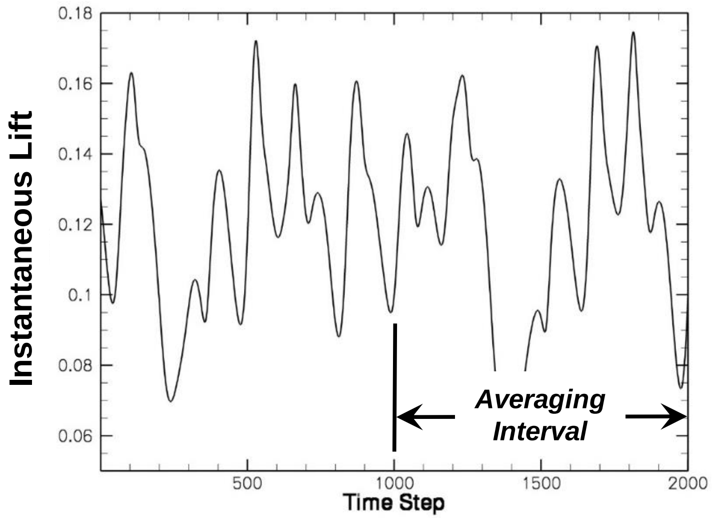
Least-Squares Shadowing (MIT)

Shedding NACA 0012
M_∞=0.1 Re=10,000 α=20°
102,940 grid points

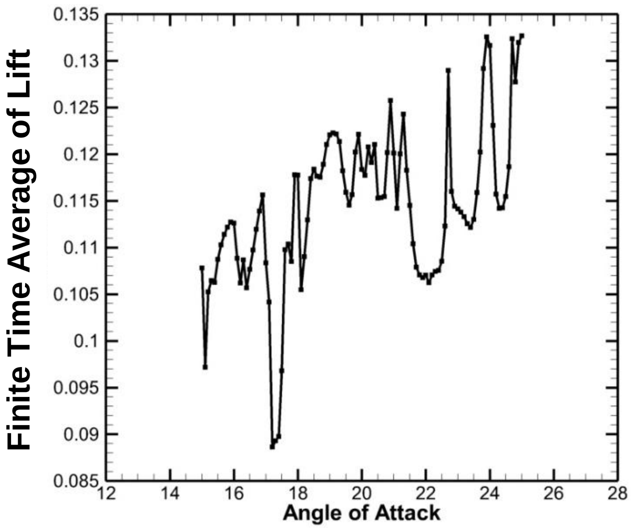
- Goal is to compute an AOA sensitivity that would allow us to maximize the time-averaged lift over final 1,000 time steps



Instantaneous Lift vs Time



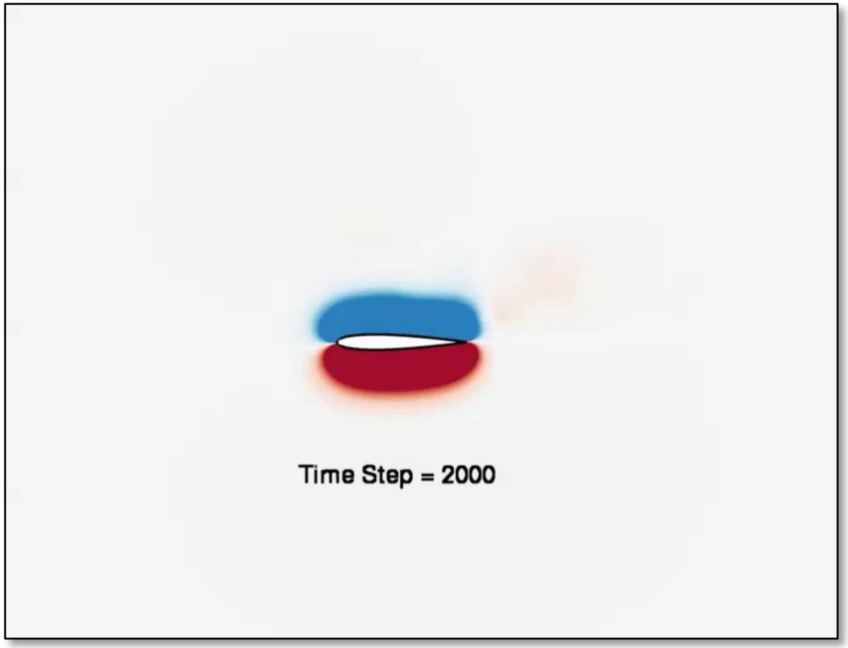
Finite Time Average of Lift vs Alpha



A Remaining Challenge: Chaos

Least-Squares Shadowing (MIT)

- FUN3D used to output data for use in LSS solver
  - Nonlinear residual vectors; Jacobians of residual, objective function
  - For this tiny problem, this is 1.1 TB of raw data
- Dimension of the resulting LSS matrix problem:
  102,940 grid points x 5 DOFs
  x 2,000 time planes = 1.03 billion
- Stand-alone LSS solver has been developed where decomposition is performed in time with a single time plane per core
- Global GMRES solver used with a local ILU(0) preconditioner for each time plane – has proven vastly inadequate
- Required ~10 hours on 2,000 cores



This is a toy problem – target simulations are 10^6 larger!
Desired matrix dimension = 10^9 x 10^6 = 10^{15}

"If you build it, we will come…"

Thank you to the organizers for having me!