Time-Dependent Adjoint Methods for Single and Multi-Disciplinary Problems

Dimitri J. Mavriplis

Max Castagne Professor

Department of Mechanical Engineering

University of Wyoming

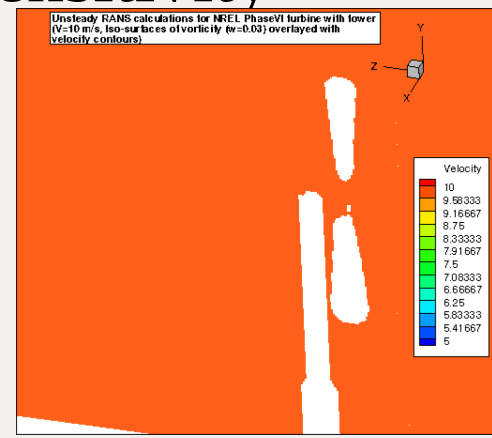
Laramie, WY USA

Motivation

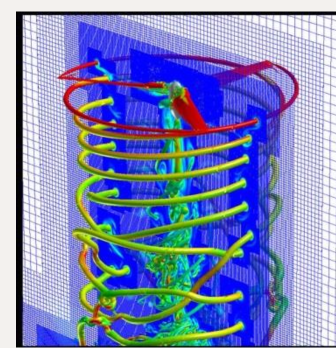
- Computational fluid dynamics analysis capabilities commonplace today
- In addition to analysis capability, sensitivity capability is highly desirable
  - Design optimization
  - Error estimation
  - Parameter sensitivity
- Sensitivities may be obtained by:
  - Perturb input, rerun analysis code (Finite difference)
  - Linearizing analysis code (tangent method)
    - Good for 1 input, many outputs
  - Adjoint method (Pironneau, Jameson, many others…)
    - Good for many inputs, one output

Objective

- Demonstrate methodical approach for formulating and implementing discrete adjoint to increasingly complex problems
- Progressively more complex simulation sensitivity formulations
  - – Steady-state aerodynamics
  - – Time-dependent aerodynamics (2D)
  - – Time-dependent coupled aero-elastic (3D)
- Focus
  - – Adjoint formulation
    - Hand coded (occasional use of AD)
    - Same data structures/solution techniques as analysis
    - Verification
  - – Exact full sensitivities in all cases
  - – Optimization examples are mostly illustrative



✓ **Time-dependent**
✓ **Aeroelastic**
✓ **Overset meshes**
✓ **Adaptive meshes**



Outline

- General unsteady tangent/adjoint formulations
  - Backward time integration
- Specific Case
  - Time-dependent aerodynamic shape optimization
  - 2D pitching airfoil optimization
- Generalized multidisciplinary formulation
  - Time-dependent aerodynamic shape optimization
  - Time-dependent aeroelastic shape optimization
- Aeroelastic rotor optimization example
- Conclusions

Adjoint Sensitivity Formulation

- Continuous vs. Discrete Adjoint Approaches
  - Continuous: Linearize then discretize
  - Discrete: Discretize then Linearize

- Continuous Approach:
  - More flexible adjoint discretizations
  - Framework for non-differentiable tasks (limiters)
  - Often invoked using flow solution as constraint using Lagrange multipliers

- Discrete Approach:
  - Reproduces exact sensitivities of code
    - Verifiable through finite differences
  - Relatively simple implementation (but tedious)
    - Chain rule differentiation of analysis code
    - Transpose these derivatives
      - (transpose and reverse order)
    - Includes boundary conditions
    - Automation possible (but use judiciously for efficiency reasons)

Adjoint Sensitivity Formulation

- Continuous vs. Discrete Adjoint Approaches
  - Continuous: Linearize then discretize
  - Discrete: Discretize then Linearize

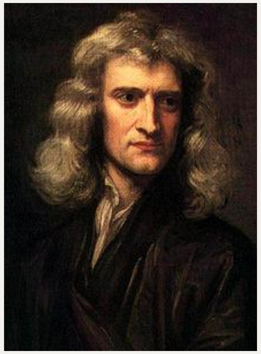
- Continuous Approach:
  - More flexible adjoint discretizations
  - Framework for non-differentiable tasks (limiters)
  - Often invoked using flow solution as constraint using Lagrange multipliers

- Discrete Approach:
  - Reproduces exact sensitivities of code
    - Verifiable through finite differences
  - Relatively simple implementation (but tedious)
    - Chain rule differentiation of analysis code
    - Transpose these derivatives
      - (transpose and reverse order)
    - Includes boundary conditions
    - Automation possible (but use judiciously for efficiency reasons)

Adjoint Sensitivity Formulation

- Continuous vs. Discrete Adjoint Approaches
  - Continuous: Linearize then discretize
  - Discrete: Discretize then Linearize

- Continuous Approach:
  - More flexible adjoint discretizations
  - Framework for non-differentiable tasks (limiters)
  - Often invoked using flow solution as constraint using Lagrange multipliers



- Discrete Approach:
  - Reproduces exact sensitivities of code
    - Verifiable through finite differences
  - Relatively simple implementation (but tedious)
    - Chain rule differentiation of analysis code
    - Transpose these derivatives
      - (transpose and reverse order)
    - Includes boundary conditions
    - Automation possible (but use judiciously for efficiency reasons)

Adjoint Formulation (chain rule approach)

- Objective: $L = L(D, u(D))$
- Subject to: $R(u(D), D) = 0$
  - R = 0 : flow solution converged
  - u : flow variables (solution)
  - D: design parameters (shape parameters)
- Sensitivities: $$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \frac{\partial L}{\partial u}\frac{\partial u}{\partial D}$$

easy

hard

- Constraint sensitivity eqn: $$\left[\frac{\partial R}{\partial u}\right]\frac{\partial u}{\partial D} + \frac{\partial R}{\partial D} = 0 \qquad \left[\frac{\partial R}{\partial u}\right]\frac{\partial u}{\partial D} = -\frac{\partial R}{\partial D}$$
- Final form: $$\frac{dL}{dD} = \frac{\partial L}{\partial D} - \frac{\partial L}{\partial u}\left[\frac{\partial R}{\partial u}\right]^{-1}\frac{\partial R}{\partial D}$$

Adjoint Formulation

- Sensitivity equation: $$\frac{dL}{dD} = \frac{\partial L}{\partial D} - \frac{\partial L}{\partial u}\left[\frac{\partial R}{\partial u}\right]^{-1}\frac{\partial R}{\partial D}$$

Evaluate first, define as Λ^T

- Adjoint equation: $$\left[\frac{\partial R}{\partial u}\right]^T \Lambda = -\left[\frac{\partial L}{\partial u}\right]^T$$
  - No dependence on D
  - Dependence on L

- Final form: $$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \Lambda^T \frac{\partial R}{\partial D}$$
- Cost is independent of number of D's
- dL/dD are then used by a gradient based optimizer to find next best shape

KKT/Lagrange Multiplier Adjoint Formulation

$$J(D, \mathbf{U}, \mathbf{\Lambda}) = L(\mathbf{U}(D), D) + \mathbf{\Lambda}^T \mathbf{R}(\mathbf{U}(D), D)$$

$$\begin{aligned} J_{min} &= \min_{D \in I} J(D) \\ subject\ to & \quad \mathbf{R}(\mathbf{U}(\mathbf{D}), \mathbf{D}) = 0 \end{aligned}$$

$$\frac{\partial J}{\partial D} = \frac{\partial L}{\partial D} + \mathbf{\Lambda}^T \frac{\partial \mathbf{R}}{\partial D} = 0$$

Sensitivity equation

$$\frac{\partial J}{\partial \mathbf{U}} = \frac{\partial L}{\partial \mathbf{U}} + \mathbf{\Lambda}^T \frac{\partial \mathbf{R}}{\partial \mathbf{U}} = 0$$

Adjoint equation

$$\frac{\partial J}{\partial \mathbf{\Lambda}} = \mathbf{R}(\mathbf{U}) = 0$$

Constraint equation

Optimization for Time Dependent Problems

- Using chain rule linearization
  - New time-step values depend on previous time step values
  - Integrate linearized equation in time (tangent problem)
  - Transpose all to get adjoint
    (and reverse order of matrix multiplication) $$\left[[A][B]\right]^T = [B]^T [A]^T$$
    - Integrate backwards in time
    - Requires storing entire time history (to disk)
      - 10 8-Byte variables per grid point per time step
      - Advantage of using large time steps with good implicit solver
      - Use local node disks on parallel computer (>1TB each node)
- Simple derivation of time-dependent adjoint
  - Chain rule linearization
  - Lagrange multipliers

Time Dependent Problems: Forward Sensitivity

- Time integrated objective functional

$$L = L(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), \mathbf{u}^{n-2}(D), ..., \mathbf{u}^0(D), D)$$

- Unsteady Residual *(BDF1 for simplicity)*

$$\mathbf{R}^n(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), D) = 0$$

- Sensitivity of objective *(chain rule)*

$$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^n}\frac{\partial \mathbf{u}^n}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^{n-1}}\frac{\partial \mathbf{u}^{n-1}}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^{n-2}}\frac{\partial \mathbf{u}^{n-2}}{\partial D} + ... + \frac{\partial L}{\partial \mathbf{u}^0}\frac{\partial \mathbf{u}^0}{\partial D}$$

Time Dependent Problems: Forward Sensitivity

$$\mathbf{R}^n(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), D) = 0$$

Obtain flow sensitivity by differentiating residual

$$\left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^n}\right] \frac{\partial \mathbf{u}^n}{\partial D} = -\left(\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^{n-1}} \frac{\partial \mathbf{u}^{n-1}}{\partial D} + \frac{\partial \mathbf{R}^n}{\partial D}\right)$$

$$\vdots \quad \vdots \quad \vdots$$

$$\left[\frac{\partial \mathbf{R}^1}{\partial \mathbf{u}^1}\right] \frac{\partial \mathbf{u}^1}{\partial D} = -\frac{\partial \mathbf{R}^1}{\partial D}$$

Recursion relation:

Forward integration in time

(requires u at each time level

Substitute into objective sensitivity equation

$$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^n} \frac{\partial \mathbf{u}^n}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^{n-1}} \frac{\partial \mathbf{u}^{n-1}}{\partial D} + \frac{\partial L}{\partial \mathbf{u}^{n-2}} \frac{\partial \mathbf{u}^{n-2}}{\partial D} + \ldots + \frac{\partial L}{\partial \mathbf{u}^0} \frac{\partial \mathbf{u}^0}{\partial D}$$

Time Dependent Problems: Adjoint Formulation

$$L = L(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), \mathbf{u}^{n-2}(D), ..., \mathbf{u}^0(D), D)$$

$$\mathbf{R}^n(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), D) = 0$$

- **n constraints (1 per time step)**
- **n Lagrange multipliers**

$$\begin{aligned}
J(D, \mathbf{u}^n, \mathbf{u}^{n-1}, ..., \mathbf{\Lambda}^n, \mathbf{\Lambda}^{n-1}, ...) &= L(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), \mathbf{u}^{n-2}(D), ..., \mathbf{u}^0(D), D) \\
&+ \mathbf{\Lambda}^{n^T}\mathbf{R}^n(\mathbf{u}^n(D), \mathbf{u}^{n-1}(D), D) \\
&+ \mathbf{\Lambda}^{n-1^T}\mathbf{R}^{n-1}(\mathbf{u}^{n-1}(D), \mathbf{u}^{n-2}(D), D) \\
&+ ... \\
&+ \mathbf{\Lambda}^{1^T}\mathbf{R}^n(\mathbf{u}^1(D), \mathbf{u}^0(D), D)
\end{aligned}$$

Time Dependent Problems: Adjoint Formulation

- Constraint equation(s)

$$\begin{aligned}
\frac{\partial J}{\partial \boldsymbol{\Lambda}^n} &= \mathbf{R}^n(\mathbf{u}^n, \mathbf{u}^{n-1}, D) = 0 \\
\frac{\partial J}{\partial \boldsymbol{\Lambda}^{n-1}} &= \mathbf{R}^{n-1}(\mathbf{u}^{n-1}, \mathbf{u}^{n-2}, D) = 0 \\
&\dots \\
\frac{\partial J}{\partial \boldsymbol{\Lambda}^1} &= \mathbf{R}^1(\mathbf{u}^1, \mathbf{u}^0, D) = 0
\end{aligned}$$

- Sensitivity equation(s)

$$\begin{aligned}
\frac{\partial J}{\partial D} = \frac{\partial L}{\partial D} &+ \boldsymbol{\Lambda}^{nT}\frac{\partial \mathbf{R}^n}{\partial D} \\
&+ \boldsymbol{\Lambda}^{n-1T}\frac{\partial \mathbf{R}^{n-1}}{\partial D} \\
&+ \boldsymbol{\Lambda}^{n-2T}\frac{\partial \mathbf{R}^{n-2}}{\partial D} \\
&+ \dots = 0
\end{aligned}$$

Time Dependent Problems: Adjoint Formulation

- Adjoint equations

$$\begin{aligned}
\frac{\partial J}{\partial \mathbf{u}^n} &= \frac{\partial L}{\partial \mathbf{u}^n} + \mathbf{\Lambda}^{nT}\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^n} = 0 \\
\frac{\partial J}{\partial \mathbf{u}^{n-1}} &= \frac{\partial L}{\partial \mathbf{u}^{n-1}} + \mathbf{\Lambda}^{nT}\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^{n-1}} + \mathbf{\Lambda}^{n-1^T}\frac{\partial R^{n-1}}{\partial \mathbf{u}^{n-1}} = 0 \\
\frac{\partial J}{\partial \mathbf{u}^{n-2}} &= \frac{\partial L}{\partial \mathbf{u}^{n-2}} + \mathbf{\Lambda}^{n-1^T}\frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{u}^{n-2}} + \mathbf{\Lambda}^{n-2^T}\frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{u}^{n-2}} = 0 \\
&\dots \\
\frac{\partial J}{\partial \mathbf{u}^1} &= \frac{\partial L}{\partial \mathbf{u}^1} + \mathbf{\Lambda}^{2^T}\frac{\partial \mathbf{R}^2}{\partial \mathbf{u}^1} + \mathbf{\Lambda}^{1^T}\frac{\partial \mathbf{R}^1}{\partial \mathbf{u}^1} = 0
\end{aligned}$$

$$\left[\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^n}\right]^T \mathbf{\Lambda}^n = -\frac{\partial L}{\partial \mathbf{u}^n}^T$$

$$\left[\frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{u}^{n-1}}\right]^T \mathbf{\Lambda}^{n-1} = -\frac{\partial L}{\partial \mathbf{u}^{n-1}}^T - \mathbf{\Lambda}^{nT}\frac{\partial \mathbf{R}^n}{\partial \mathbf{u}^{n-1}}$$

- Reverse recurrence relation
- Solved by back-substitution
  - Requires u at each time level

Generalized Discrete Sensitivities

- Consider a multi-phase analysis code:

$$\mathbf{L}(\mathbf{D}) = \mathbf{L}(F_{n-1}(F_{n-2}(....F_2(F_1(\mathbf{D}))....))))$$

  - L = Objective(s)
  - D = Design variable(s)
- Sensitivity Analysis
  - Using chain rule:

$$\delta\mathbf{L} = \frac{d\mathbf{L}}{d\mathbf{D}}\delta\mathbf{D}$$

$$\frac{d\mathbf{L}}{d\mathbf{D}} = \frac{\partial\mathbf{L}}{\partial F_{n-1}} \cdot \frac{\partial F_{n-1}}{\partial F_{n-2}} \cdots\cdots \frac{\partial F_2}{\partial F_1} \cdot \frac{\partial F_1}{\partial \mathbf{D}}$$

Tangent Model

- Special Case:
  - 1 Design variable D, many objectives L
- Precompute all stuff depending on single D
- Construct dL/dD elements as:

$$\frac{d\mathbf{L}}{d\mathbf{D}} = \frac{\partial \mathbf{L}}{\partial F_{n-1}} \cdot \left[\frac{\partial F_{n-1}}{\partial F_{n-2}} \left[\cdots\cdots \left[\frac{\partial F_2}{\partial F_1} \left[\frac{\partial F_1}{\partial \mathbf{D}} \right] \right] \right] \right]$$

Adjoint Model

- Special Case:
  - 1 Objective L, Many Design Variables D
  - Would like to precompute all left terms

$$\frac{d\mathbf{L}}{d\mathbf{D}} = \frac{\partial \mathbf{L}}{\partial F_{n-1}} \cdot \frac{\partial F_{n-1}}{\partial F_{n-2}} \cdots\cdots \frac{\partial F_2}{\partial F_1} \cdot \frac{\partial F_1}{\partial \mathbf{D}}$$

  - Transpose entire equation:

$$\frac{d\mathbf{L}}{d\mathbf{D}}^T = \frac{\partial F_1}{\partial \mathbf{D}}^T \cdot \frac{\partial F_2}{\partial F_1}^T \cdot \frac{\partial F_{n-1}}{\partial F_{n-2}}^T \cdots\cdots \frac{\partial \mathbf{L}}{\partial F_{n-1}}^T$$

Adjoint Model

- Special Case:
  - 1 Objective L, Many Design Variables D
  - Would like to precompute all left terms

$$\frac{d\mathbf{L}}{d\mathbf{D}} = \frac{\partial \mathbf{L}}{\partial F_{n-1}} \cdot \frac{\partial F_{n-1}}{\partial F_{n-2}} \cdot \cdots \cdot \frac{\partial F_2}{\partial F_1} \cdot \frac{\partial F_1}{\partial \mathbf{D}}$$

  - Transpose entire equation: precompute as:

$$\frac{d\mathbf{L}}{d\mathbf{D}}^T = \frac{\partial F_1}{\partial \mathbf{D}}^T \cdot \left[\frac{\partial F_2}{\partial F_1}^T \cdot \left[\frac{\partial F_{n-1}}{\partial F_{n-2}}^T \cdot \left[\cdots \left[\frac{\partial \mathbf{L}}{\partial F_{n-1}}^T \right] \right] \right] \right]$$

Steady-State Shape Optimization Problem

- Multi-phase process:

$$\mathbf{L}(\mathbf{D}) \quad = \quad \mathbf{L}(F_3(F_2(F_1(\mathbf{D}))))$$

$$\mathbf{x}_{surf} \quad = \quad F_1(\mathbf{D})$$

$$\mathbf{x}_{int} \quad = \quad F_2(\mathbf{x}_{surf})$$

$$\mathbf{w} \quad = \quad F_3(\mathbf{x}_{int})$$

$$L \quad = \quad L(\mathbf{w}, \mathbf{x}_{int})$$

Tangent Problem

- 1: Surface mesh sensitivity: $$\frac{\partial \mathbf{x}_{surf}}{\partial \mathbf{D}}$$
- 2: Interior mesh sensitivity: $$[K]\frac{\partial \mathbf{x}_{int}}{\partial \mathbf{D}} = \frac{\partial \mathbf{x}_{surf}}{\partial \mathbf{D}}$$
- 3: Residual sensitivity: $$\frac{\partial \mathbf{R}}{\partial \mathbf{D}} = \frac{\partial \mathbf{R}}{\partial \mathbf{x}_{int}} \cdot \frac{\partial \mathbf{x}_{int}}{\partial \mathbf{D}}$$
- 4: Flow variable sensitivity: $$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]\frac{\partial \mathbf{w}}{\partial \mathbf{D}} = -\frac{\partial \mathbf{R}}{\partial \mathbf{D}}$$
- 5: Final sensitivity $$\frac{d\mathbf{L}}{d\mathbf{D}} = \frac{\partial \mathbf{L}}{\partial \mathbf{w}} \cdot \frac{\partial \mathbf{w}}{\partial \mathbf{D}} + \frac{\partial \mathbf{L}}{\partial \mathbf{x}_{int}} \cdot \frac{\partial \mathbf{x}_{int}}{\partial \mathbf{D}}$$

Adjoint Problem

- 1: Objective flow sensitivity: $$\frac{\partial \mathbf{L}}{\partial \mathbf{w}}$$
- 2: Flow adjoint: $$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^T \mathbf{\Lambda}_w = \frac{\partial \mathbf{L}}{\partial \mathbf{w}}^T$$
- 3:Objective sens. wrt mesh: $$\frac{d\mathbf{L}}{d\mathbf{x}^*_{\mathbf{int}}}^T = \frac{\partial \mathbf{L}}{\partial \mathbf{x}_{int}}^T - \frac{\partial \mathbf{R}}{\partial \mathbf{x}_{int}}^T \mathbf{\Lambda}_w$$
- 4: Mesh adjoint: $$[K]^T \mathbf{\Lambda}_x = \frac{d\mathbf{L}}{d\mathbf{x}^*_{int}}^T$$
- 5: Final sensitivity: $$\frac{d\mathbf{L}}{d\mathbf{D}}^T = \frac{\partial \mathbf{x}_{surf}}{\partial \mathbf{D}}^T \mathbf{\Lambda}_x$$

General Approach

- Linearize each subroutine/process individually in analysis code (tangent or forward model)
  - Check linearization by finite difference/complex variables/dual numbers/AD
  - Transpose to get adjoint, and check duality relation
    - Should reproduce same sensitivities to machine precision
- Build up larger components
  - Check linearization, duality relation
- Check entire process for FD/Complex and duality
- Use single modular solver for all phases
- Maintaining forward linearization has advantages
  - Cases with few design variables, many objectives
  - Debugging adjoint code
  - Enables exact Jacobian/vector products for Krylov solve
  - Facilitates Hessian calculations (later…)

Verification: Complex Step

- Finite difference approach is plagued by round-off error (small ε) versus non-linear error (large ε)
  - Must find range of ε that gives accurate sensitivities
  - Sometimes no such range exists
- Complex variable approach:
  - Replace $f(x)$ with complex function $f(x+i\varepsilon)$
  - Then $df/dx = \mathrm{Im}(f(x+i\varepsilon))/\varepsilon$
  - Can take ε=1.e-100 (no roundoff error)
  - Very accurate gradients (machine precision)

General Duality Relation

•Analysis Routine: $f = f(x)$

•Tangent Model: $\delta f_1 = \frac{\partial f}{\partial x} \delta x_1$

•Adjoint Model: $\delta x_2 = \frac{\partial f}{\partial x}^T \partial f_2$

•Duality Relation: $\partial f_2^T . \partial f_1 = \partial x_2^T . \partial x_1$

- Necessary but not sufficient test
  - Check using series of arbitrary input vectors $\partial x_1 \; \partial f_2$

General Duality Relation

•Analysis Routine: $$f = f(x)$$

•Tangent Model: $$\delta f_1 = \frac{\partial f}{\partial x} \delta x_1$$

•Adjoint Model: $$\delta x_2 = \frac{\partial f}{\partial x}^T \partial f_2$$

•Duality Relation: $$\partial f_2^T . \partial f_1 = \partial x_2^T . \partial x_1$$

- Necessary but not sufficient test
  - Check using series of arbitrary input vectors $\partial x_1 \; \partial f_2$

Time Dependent & Moving Mesh Shape Optimization Problem

- Flow equations written in ALE form
- Functional dependence of residual (BDF2)
- Functional dependence of mesh motion equations (same as before)

$$R^n(U^n, U^{n-1}, U^{n-2}, x^n, x^{n-1}, x^{n-2}) = 0$$

$$G(x^n, x^n_{surf}(D)) = 0 \Longrightarrow [K]\delta x = \delta x_{surf}$$

Time Dependent & Moving Mesh Shape Optimization Problem

- Flow equations written in ALE form
- Using space-time notation
  - U = solution over all space and time
  - R = residuals over all space and time

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right] = \begin{bmatrix} \ddots & 0 & 0 & 0 & 0 \\ \ddots & \ddots & 0 & 0 & 0 \\ \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{U}^{n-4}} & \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{U}^{n-3}} & \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{U}^{n-2}} & 0 & 0 \\ 0 & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{U}^{n-3}} & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{U}^{n-2}} & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{U}^{n-1}} & 0 \\ 0 & 0 & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{U}^{n-2}} & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{U}^{n-1}} & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{U}^{n}} \end{bmatrix}$$

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{x}}\right] = \begin{bmatrix} \ddots & 0 & 0 & 0 & 0 \\ \ddots & \ddots & 0 & 0 & 0 \\ \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{x}^{n-4}} & \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{x}^{n-3}} & \frac{\partial \mathbf{R}^{n-2}}{\partial \mathbf{x}^{n-2}} & 0 & 0 \\ 0 & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{x}^{n-3}} & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{x}^{n-2}} & \frac{\partial \mathbf{R}^{n-1}}{\partial \mathbf{x}^{n-1}} & 0 \\ 0 & 0 & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{x}^{n-2}} & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{x}^{n-1}} & \frac{\partial \mathbf{R}^{n}}{\partial \mathbf{x}^{n}} \end{bmatrix}$$

Time Dependent & Moving Mesh Shape Optimization Problem

- Flow adjoint solved by backsubstitution
  - Using 2 previous levels in time (BDF2)

$$\left[\frac{\partial R^n}{\partial U^n}\right]^T \Lambda_u^n = -\left[\frac{\partial L}{\partial U^n}\right]^T + \left[\frac{\partial R^{n+1}}{\partial U^n}\right]^T \Lambda_u^{n+1} + \left[\frac{\partial R^{n+2}}{\partial U^n}\right]\Lambda_u^{n+2}$$

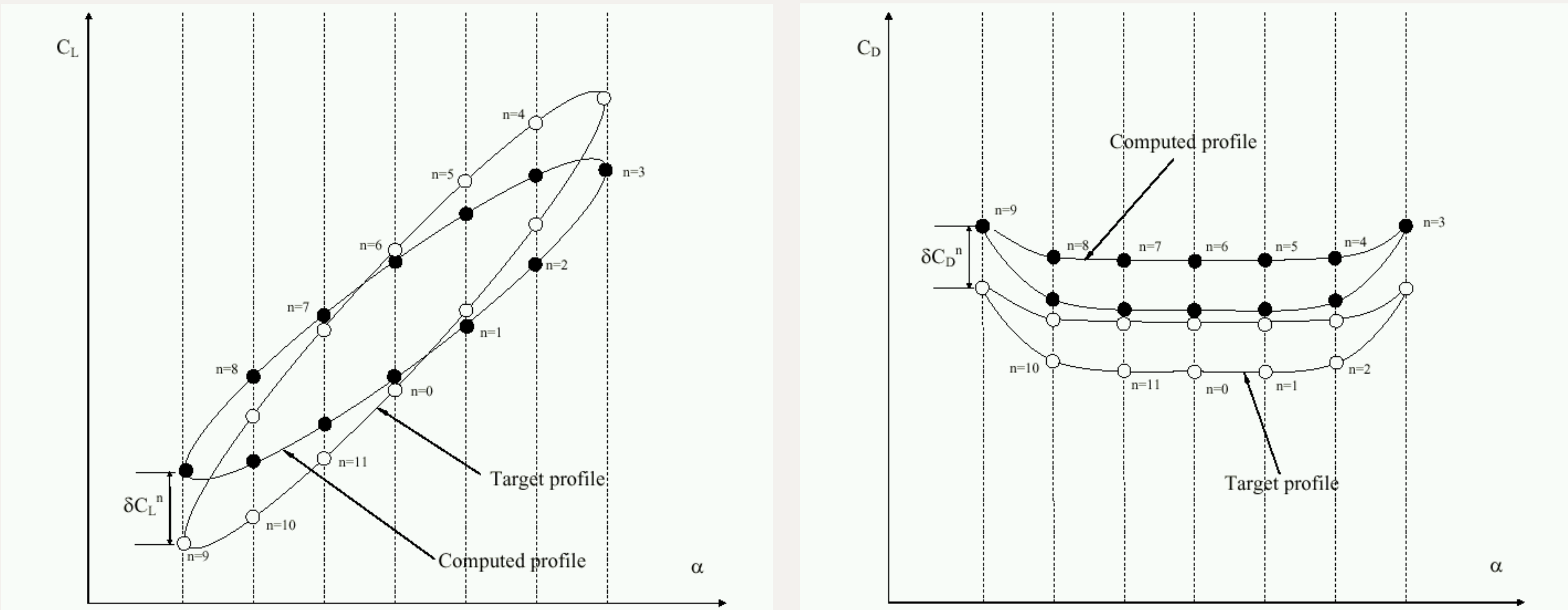
- Interior mesh sensitivities : dR/dX inner product (2 levels back in time)

$$[K]^T \Lambda_x^n = \left[\frac{\partial L}{\partial x^n}\right]^T + \left[\frac{\partial R^n}{\partial x^n}\right]^T \Lambda_u^n + \left[\frac{\partial R^{n+1}}{\partial x^n}\right]^T \Lambda_u^{n+1} + \left[\frac{\partial R^{n+2}}{\partial x^n}\right]\Lambda_u^{n+2}$$

- Final (surface) sensitivity
  - Single mesh adjoint problem

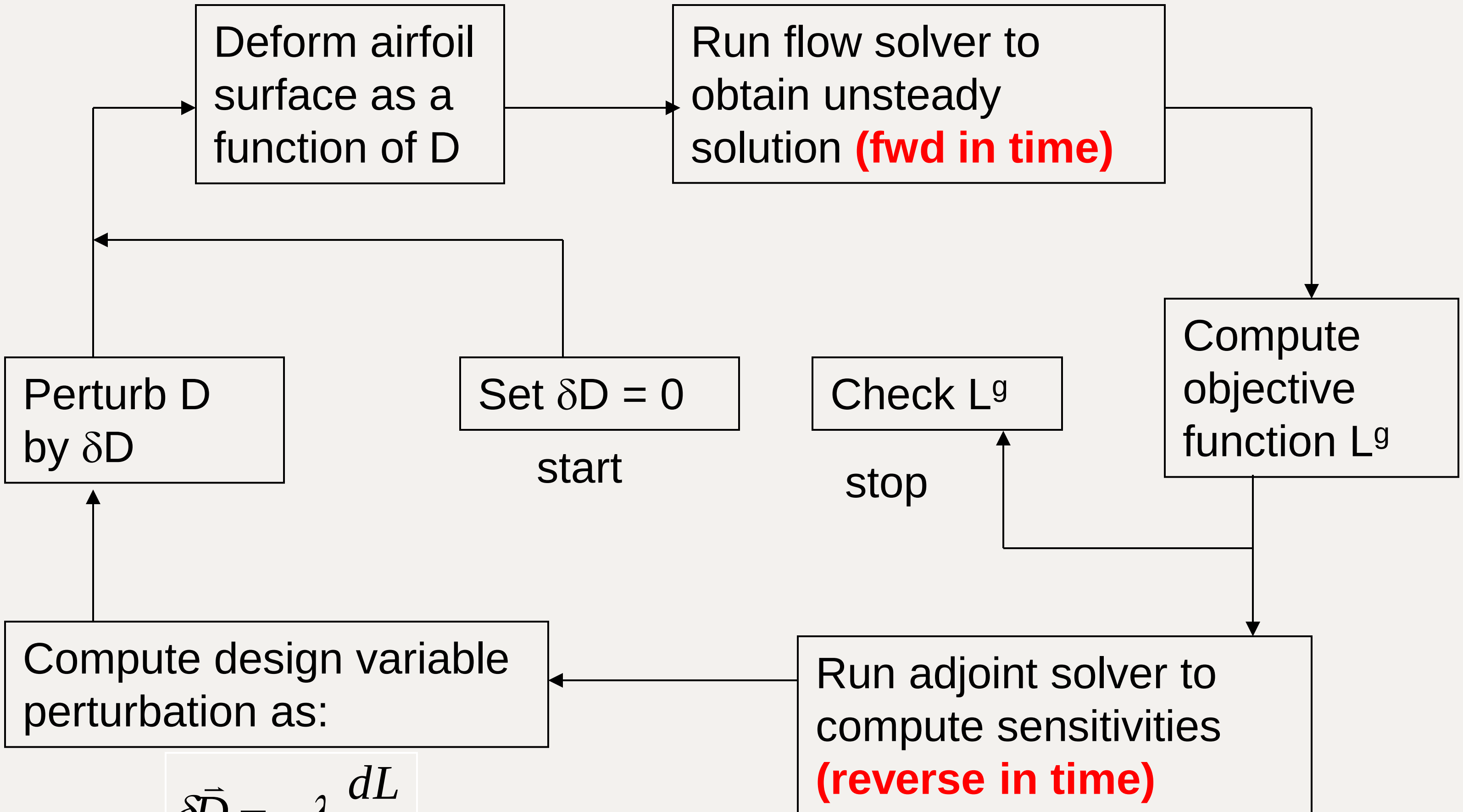
$$\frac{dL}{dD} = \Lambda_x^n \frac{\partial x_{surf}}{\partial D} + \Lambda_x^{n-1} \frac{\partial x_{surf}}{\partial D} + \ldots$$

Time-Integrated Objective Formulation



Pitching airfoil time histories of CL and CD

Optimization Procedure



$$\delta\vec{D} = -\lambda \frac{dL}{d\vec{D}}$$

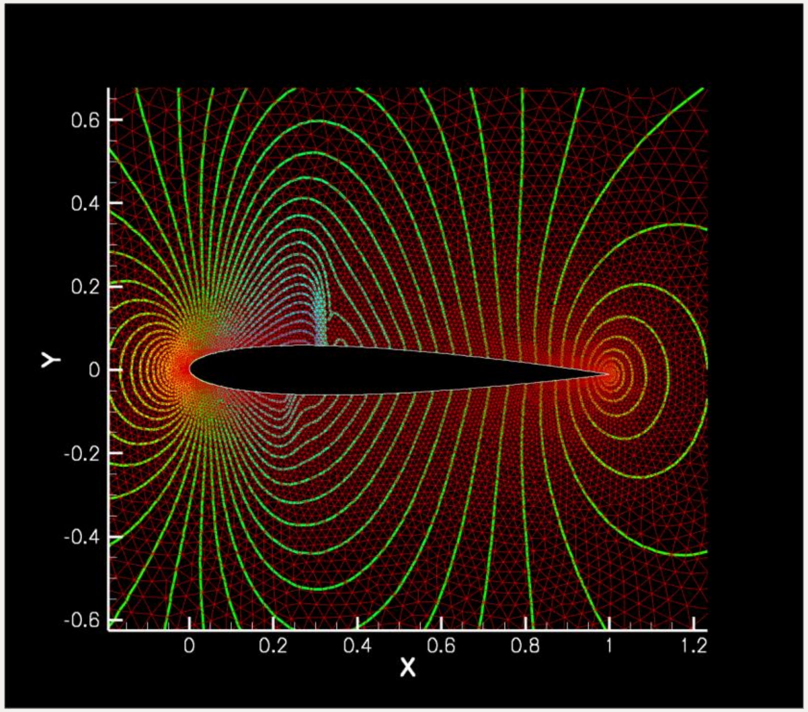
(or provide gradient to optimizer)

Unsteady Flow Solution

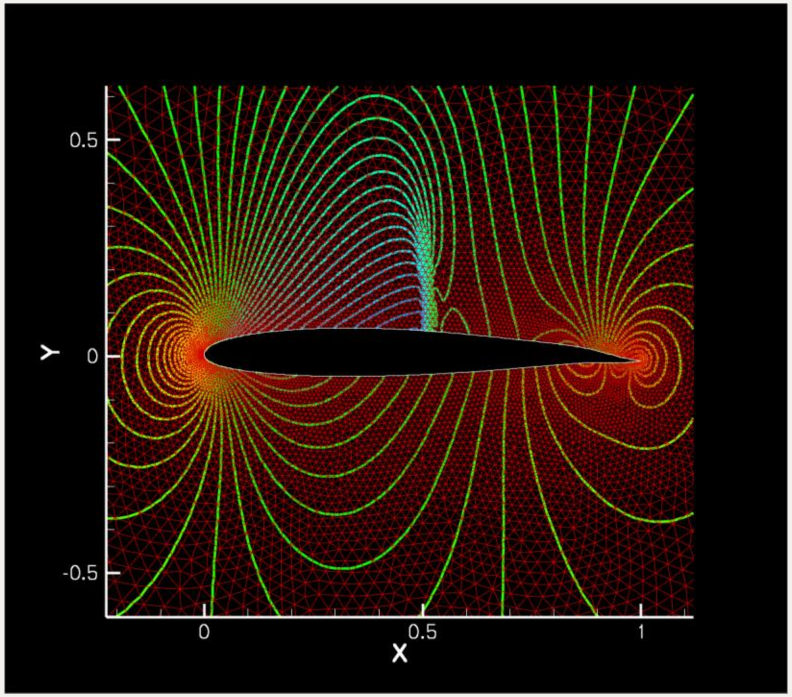
Pressure Contours for Pitching Airfoils

M_{inf} = 0.755, α_0 = 0.016°, α_{max} = 2.51°, ω = 0.1628, t=0 to 54

27 time-steps with dt=2.0

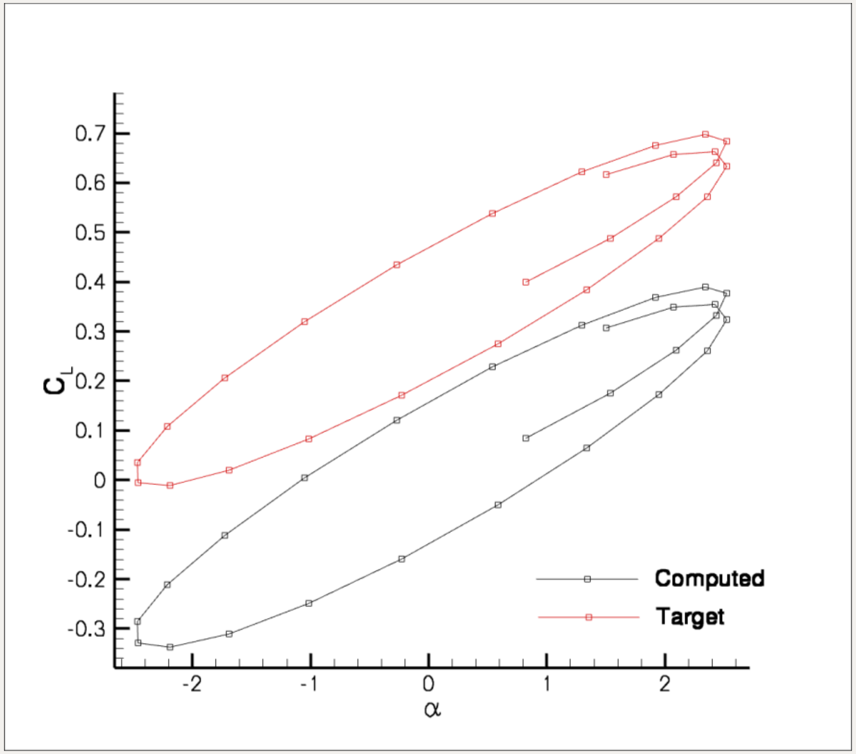


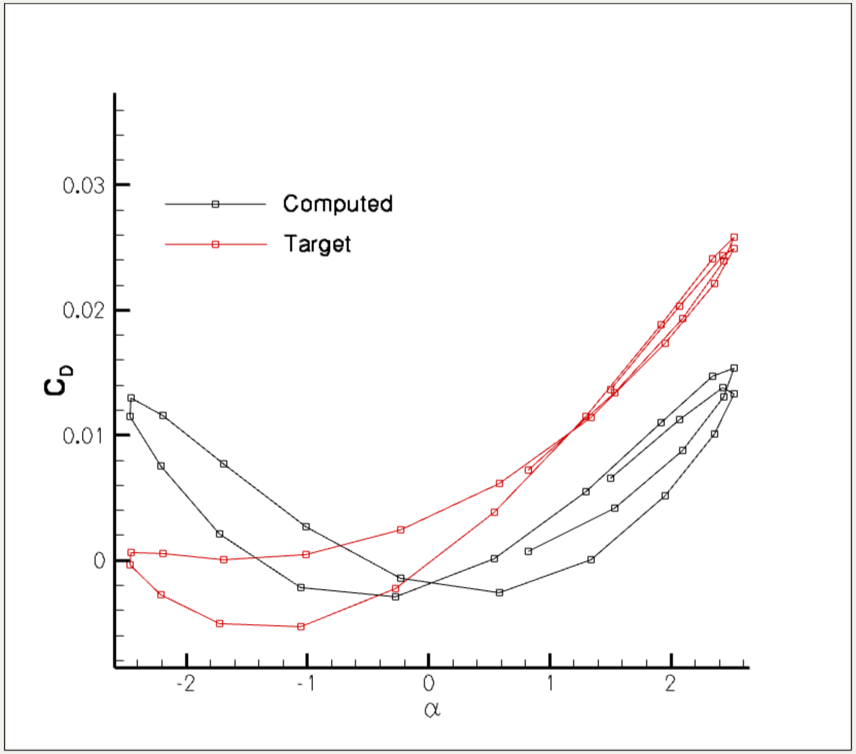
NACA0012 Baseline Airfoil



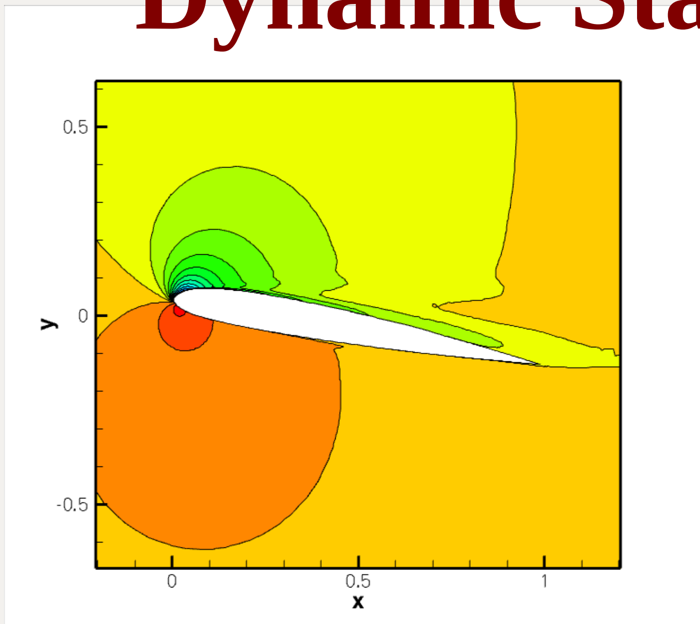
Optimized Airfoil

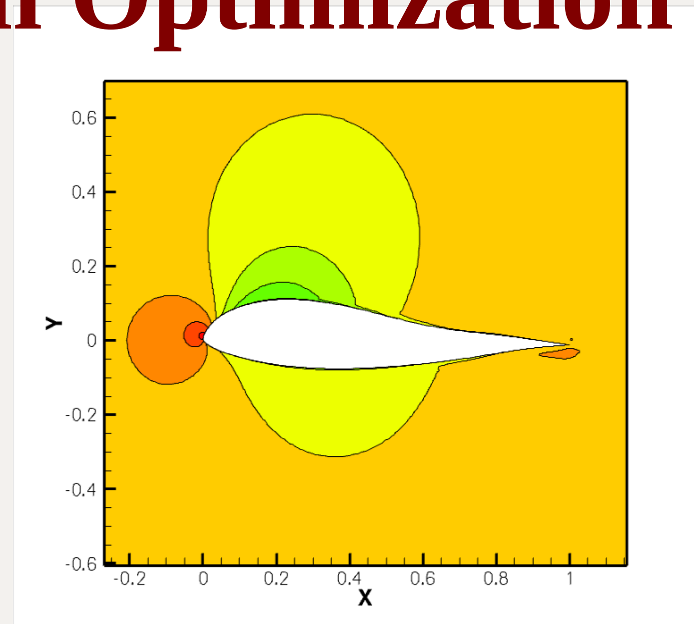
Time-Dependent Load Convergence



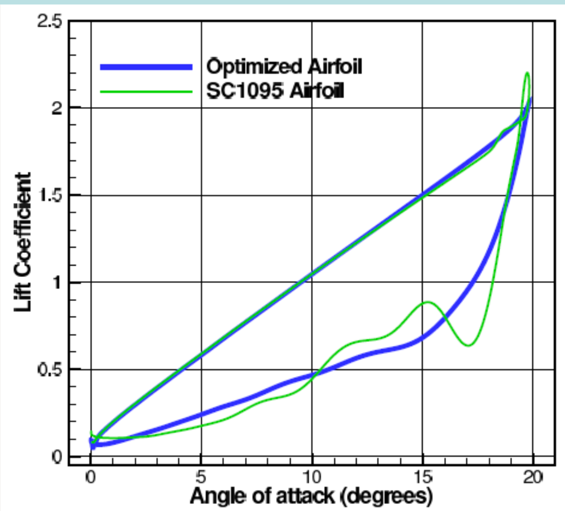


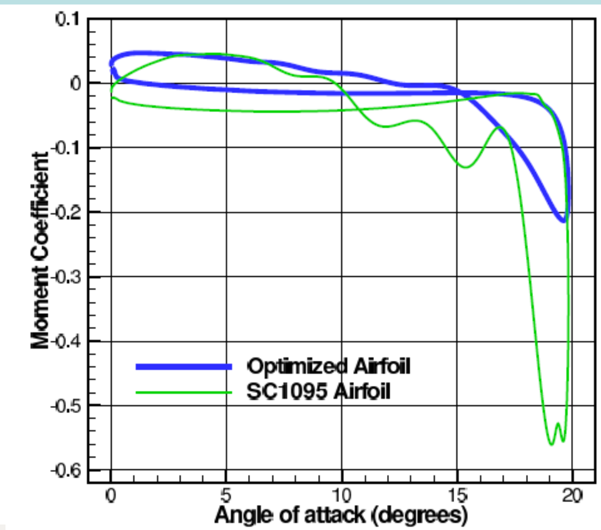
Dynamic Stall Optimization

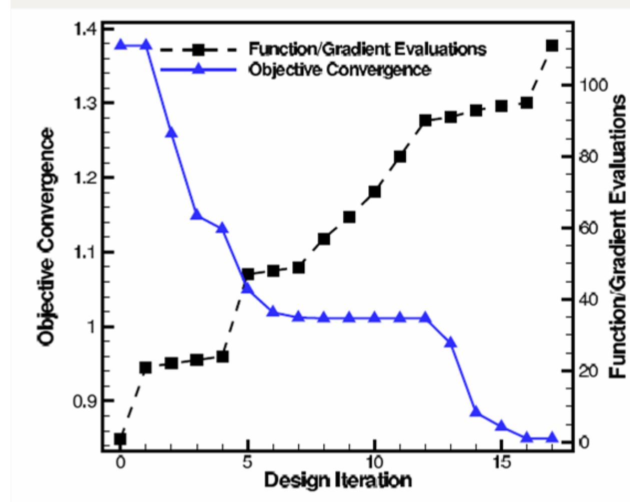




Optimized airfoil performance: Objective=Minimize moment excursions at constant lift







$$L = \| C_M(t) \|^{100} + \omega \| C_L(t) - C_{L_{TARGET}}(t) \|^2$$

Generalized Form for Multidisciplinary Unsteady Coupled Equations

m coupled disciplines=> L is a functional computed using multidisciplinary solution set:

$$L = L(D, \mathscr{U}_1, \mathscr{U}_2, \cdots, \mathscr{U}_m)$$

U over all space and time

m^{th} discipline

Linearize using chain rule with respect to **D**:

$$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \frac{\partial L}{\partial \mathscr{U}_1}\frac{\partial \mathscr{U}_1}{\partial D} + \frac{\partial L}{\partial \mathscr{U}_2}\frac{\partial \mathscr{U}_2}{\partial D} + \cdots + \frac{\partial L}{\partial \mathscr{U}_m}\frac{\partial \mathscr{U}_m}{\partial D}$$

Inner-product form:

$$\frac{dL}{dD} = \frac{\partial L}{\partial D} + \begin{bmatrix} \frac{\partial L}{\partial \mathscr{U}_1} & \frac{\partial L}{\partial \mathscr{U}_2} & \cdots & \frac{\partial L}{\partial \mathscr{U}_m} \end{bmatrix} \begin{bmatrix} \frac{\partial \mathscr{U}_1}{\partial D} \\ \frac{\partial \mathscr{U}_2}{\partial D} \\ \vdots \\ \frac{\partial \mathscr{U}_m}{\partial D} \end{bmatrix}$$

Tranpose for adjoint total sensitivity:

$$\frac{dL}{dD}^T = \frac{\partial L}{\partial D}^T + \begin{bmatrix} \frac{\partial \mathscr{U}_1}{\partial D}^T & \frac{\partial \mathscr{U}_2}{\partial D}^T & \cdots & \frac{\partial \mathscr{U}_m}{\partial D}^T \end{bmatrix} \begin{bmatrix} \frac{\partial L}{\partial \mathscr{U}_1}^T \\ \frac{\partial L}{\partial \mathscr{U}_2}^T \\ \vdots \\ \frac{\partial L}{\partial \mathscr{U}_m}^T \end{bmatrix}$$

Inner products over all space and time

Generalized Form for Multidisciplinary Unsteady Coupled Equations

Residual equations for *m* disciplines:

$$
\begin{aligned}
\mathscr{R}_1(D, \mathscr{U}_1, \mathscr{U}_2, \cdots, \mathscr{U}_m) &= 0 \\
\mathscr{R}_2(D, \mathscr{U}_1, \mathscr{U}_2, \cdots, \mathscr{U}_m) &= 0 \\
&\vdots \\
\mathscr{R}_m(D, \mathscr{U}_1, \mathscr{U}_2, \cdots, \mathscr{U}_m) &= 0
\end{aligned}
$$

Linearize with respect to **D:**

$$
\begin{aligned}
\frac{\partial \mathscr{R}_1}{\partial D} + \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_1}\frac{\partial \mathscr{U}_1}{\partial D} + \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_2}\frac{\partial \mathscr{U}_2}{\partial D} + \cdots + \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_m}\frac{\partial \mathscr{U}_m}{\partial D} &= 0 \\
\frac{\partial \mathscr{R}_2}{\partial D} + \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_1}\frac{\partial \mathscr{U}_1}{\partial D} + \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_2}\frac{\partial \mathscr{U}_2}{\partial D} + \cdots + \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_m}\frac{\partial \mathscr{U}_m}{\partial D} &= 0 \\
&\vdots \\
\frac{\partial \mathscr{R}_m}{\partial D} + \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_1}\frac{\partial \mathscr{U}_1}{\partial D} + \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_2}\frac{\partial \mathscr{U}_2}{\partial D} + \cdots + \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_m}\frac{\partial \mathscr{U}_m}{\partial D} &= 0
\end{aligned}
$$

Write in combined matrix form:

$$
\begin{bmatrix}
\frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_1} & \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_2} & \cdots & \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_m} \\
\frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_1} & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_2} & \cdots & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_m} \\
\vdots & & & \\
\frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_1} & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_2} & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_m}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial \mathscr{U}_1}{\partial D} \\
\frac{\partial \mathscr{U}_2}{\partial D} \\
\vdots \\
\frac{\partial \mathscr{U}_m}{\partial D}
\end{bmatrix}
=
\begin{bmatrix}
-\frac{\partial \mathscr{R}_1}{\partial D} \\
-\frac{\partial \mathscr{R}_2}{\partial D} \\
\vdots \\
-\frac{\partial \mathscr{R}_m}{\partial D}
\end{bmatrix}
$$

Generalized Form for Multidisciplinary Unsteady Coupled Equations

Transpose and rearrange for vector of state sensitivity matrices:

$$\begin{bmatrix} \dfrac{\partial \mathscr{U}_1}{\partial D}^T & \dfrac{\partial \mathscr{U}_2}{\partial D}^T & \cdots & \dfrac{\partial \mathscr{U}_m}{\partial D}^T \end{bmatrix} =$$

$$- \begin{bmatrix} \dfrac{\partial \mathscr{R}_1}{\partial D}^T & \dfrac{\partial \mathscr{R}_2}{\partial D}^T & \cdots & \dfrac{\partial \mathscr{R}_m}{\partial D}^T \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathscr{R}_1}{\partial \mathscr{U}_1}^T & \dfrac{\partial \mathscr{R}_2}{\partial \mathscr{U}_1}^T & \cdots & \dfrac{\partial \mathscr{R}_m}{\partial \mathscr{U}_1}^T \\ \dfrac{\partial \mathscr{R}_1}{\partial \mathscr{U}_2}^T & \dfrac{\partial \mathscr{R}_2}{\partial \mathscr{U}_2}^T & \cdots & \dfrac{\partial \mathscr{R}_m}{\partial \mathscr{U}_2}^T \\ \vdots & & & \\ \dfrac{\partial \mathscr{R}_1}{\partial \mathscr{U}_m}^T & \dfrac{\partial \mathscr{R}_2}{\partial \mathscr{U}_m}^T & \cdots & \dfrac{\partial \mathscr{R}_m}{\partial \mathscr{U}_m}^T \end{bmatrix}^{-1}$$

Generalized Form for Multidisciplinary Unsteady Coupled Equations

Substitute into adjoint total sensitivity equation:

$$\frac{dL}{dD}^T = \frac{\partial L}{\partial D}^T - \left[\frac{\partial \mathcal{R}_1}{\partial D}^T \quad \frac{\partial \mathcal{R}_2}{\partial D}^T \quad \dots \quad \frac{\partial \mathcal{R}_m}{\partial D}^T \right] \begin{bmatrix} \frac{\partial \mathcal{R}_1}{\partial \mathcal{U}_1}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{U}_1}^T & \dots & \frac{\partial \mathcal{R}_m}{\partial \mathcal{U}_1}^T \\ \frac{\partial \mathcal{R}_1}{\partial \mathcal{U}_2}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{U}_2}^T & \dots & \frac{\partial \mathcal{R}_m}{\partial \mathcal{U}_2}^T \\ \vdots & & & \\ \frac{\partial \mathcal{R}_1}{\partial \mathcal{U}_m}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{U}_m}^T & \dots & \frac{\partial \mathcal{R}_m}{\partial \mathcal{U}_m}^T \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial L}{\partial \mathcal{U}_1}^T \\ \frac{\partial L}{\partial \mathcal{U}_2}^T \\ \vdots \\ \frac{\partial L}{\partial \mathcal{U}_m}^T \end{bmatrix}$$

Generalized Form for Multidisciplinary Unsteady Coupled Equations

Define vector of disciplinary adjoint variables:

$$\begin{bmatrix} \Lambda_{\mathscr{U}_1} \\ \Lambda_{\mathscr{U}_2} \\ \vdots \\ \Lambda_{\mathscr{U}_m} \end{bmatrix} = - \begin{bmatrix} \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_1}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_1}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_1}^T \\ \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_2}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_2}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_2}^T \\ \vdots & & & \\ \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_m}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_m}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_m}^T \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial L}{\partial \mathscr{U}_1}^T \\ \frac{\partial L}{\partial \mathscr{U}_2}^T \\ \vdots \\ \frac{\partial L}{\partial \mathscr{U}_m}^T \end{bmatrix}$$

Rearrange to recover linear adjoint system:

$$\begin{bmatrix} \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_1}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_1}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_1}^T \\ \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_2}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_2}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_2}^T \\ \vdots & & & \\ \frac{\partial \mathscr{R}_1}{\partial \mathscr{U}_m}^T & \frac{\partial \mathscr{R}_2}{\partial \mathscr{U}_m}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial \mathscr{U}_m}^T \end{bmatrix} \begin{bmatrix} \Lambda_{\mathscr{U}_1} \\ \Lambda_{\mathscr{U}_2} \\ \vdots \\ \Lambda_{\mathscr{U}_m} \end{bmatrix} = - \begin{bmatrix} \frac{\partial L}{\partial \mathscr{U}_1}^T \\ \frac{\partial L}{\partial \mathscr{U}_2}^T \\ \vdots \\ \frac{\partial L}{\partial \mathscr{U}_m}^T \end{bmatrix}$$

Generalized Form for Multidisciplinary Unsteady Coupled Equations

General Jacobian matrix when expanded discretely in time is lower triangular due to hyperbolic nature of time:

$$\frac{\partial \mathscr{R}}{\partial \mathscr{U}} = \begin{bmatrix} \ddots & 0 & 0 & 0 \\ \cdots & \dfrac{\partial \mathscr{R}^{n-2}}{\partial \mathscr{U}^{n-2}} & 0 & 0 \\ \cdots & \dfrac{\partial \mathscr{R}^{n-1}}{\partial \mathscr{U}^{n-2}} & \dfrac{\partial \mathscr{R}^{n-1}}{\partial \mathscr{U}^{n-1}} & 0 \\ \cdots & \dfrac{\partial \mathscr{R}^{n}}{\partial \mathscr{U}^{n-2}} & \dfrac{\partial \mathscr{R}^{n}}{\partial \mathscr{U}^{n-1}} & \dfrac{\partial \mathscr{R}^{n}}{\partial \mathscr{U}^{n}} \end{bmatrix}$$

$$\left[\begin{array}{cc|cc} \dfrac{\partial \mathscr{R}_1^{n-1}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n-1}}{\partial \mathscr{U}_2^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_2^{n-1}}^T \\ 0 & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_1^{n}}^T & 0 & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_2^{n}}^T \\ \hline \dfrac{\partial \mathscr{R}_2^{n-1}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n-1}}{\partial \mathscr{U}_2^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_2^{n-1}}^T \\ 0 & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_1^{n}}^T & 0 & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_2^{n}}^T \end{array}\right] \left[\begin{array}{c} \Lambda_{\mathscr{U}_1}^{n-1} \\ \Lambda_{\mathscr{U}_1}^{n} \\ \hline \Lambda_{\mathscr{U}_2}^{n-1} \\ \Lambda_{\mathscr{U}_2}^{n} \end{array}\right] = - \left[\begin{array}{c} \dfrac{\partial L}{\partial \mathscr{U}_1^{n-1}}^T \\ \dfrac{\partial L}{\partial \mathscr{U}_1^{n}}^T \\ \hline \dfrac{\partial L}{\partial \mathscr{U}_2^{n-1}}^T \\ \dfrac{\partial L}{\partial \mathscr{U}_2^{n}}^T \end{array}\right]$$

$$\left[\begin{array}{cc|cc} \dfrac{\partial \mathscr{R}_1^{n-1}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n-1}}{\partial \mathscr{U}_2^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_2^{n-1}}^T \\ \dfrac{\partial \mathscr{R}_2^{n-1}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n-1}}{\partial \mathscr{U}_2^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_1^{n-1}}^T & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_2^{n-1}}^T \\ \hline 0 & 0 & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_1^{n}}^T & \dfrac{\partial \mathscr{R}_1^{n}}{\partial \mathscr{U}_2^{n}}^T \\ 0 & 0 & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_1^{n}}^T & \dfrac{\partial \mathscr{R}_2^{n}}{\partial \mathscr{U}_2^{n}}^T \end{array}\right] \left[\begin{array}{c} \Lambda_{\mathscr{U}_1}^{n-1} \\ \Lambda_{\mathscr{U}_2}^{n-1} \\ \hline \Lambda_{\mathscr{U}_1}^{n} \\ \Lambda_{\mathscr{U}_2}^{n} \end{array}\right] = - \left[\begin{array}{c} \dfrac{\partial L}{\partial \mathscr{U}_1^{n-1}}^T \\ \dfrac{\partial L}{\partial \mathscr{U}_2^{n-1}}^T \\ \hline \dfrac{\partial L}{\partial \mathscr{U}_1^{n}}^T \\ \dfrac{\partial L}{\partial \mathscr{U}_2^{n}}^T \end{array}\right]$$

Swap rows and columns to obtain an upper triangular linear system that can be solved by back substitution

Generalized Form for Multidisciplinary Unsteady Coupled Equations

Finally, substitute vector of disciplinary adjoint variables into total sensitivity equation to obtain gradient

$$\frac{dL}{dD}^T = \frac{\partial L}{\partial D}^T + \begin{bmatrix} \frac{\partial \mathscr{R}_1}{\partial D}^T & \frac{\partial \mathscr{R}_2}{\partial D}^T & \cdots & \frac{\partial \mathscr{R}_m}{\partial D}^T \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_{\mathscr{U}_1} \\ \mathbf{\Lambda}_{\mathscr{U}_2} \\ \cdots \\ \mathbf{\Lambda}_{\mathscr{U}_m} \end{bmatrix}$$

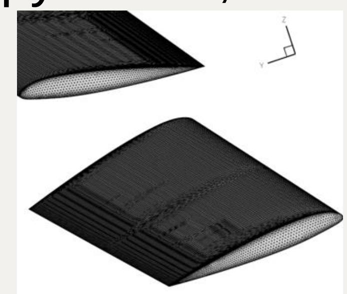
Extension to Multidisciplinary Problems: Time-Dependent Aeroelasticity

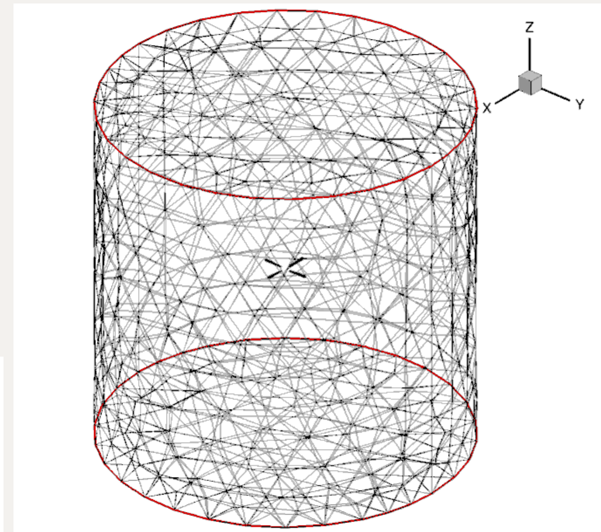
- Fully coupled problem involves 4 modules:
  - Flow solver
  - Structural Solver
  - Mesh deformation
  - Fluid-structure interface (FSI)
- Adjoint formulation leads to :
  - Disciplinary adjoints: Fluids, Structures, Mesh, FSI
  - Disciplinary adjoints are coupled at each time step
  - Coupled adjoint solver analogous (transpose) of coupled aeroelastic analysis solver

Forward Flight Flexible Rotor Optimization

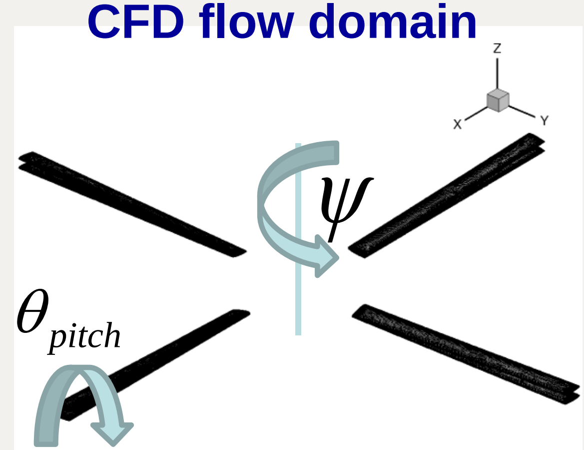
- 4 bladed Hart-II rotor in forward flight:
  - Rigid & Flexible; $M_{tip} = 0.64$; 1040 RPM; $\mu=0.15$ (M_{∞}~0.1); $\alpha=5.4^{O}$
- CFD/CSD specifications:
  - 2.32 million grid nodes (prisms, pyramids, tets)
  - 20 beam elements per blade
- Tight CFD/CSD coupling
  - 2 rotor revs
  - 2.32M : $\Delta t=2^{O}$
    - 6 coupling per time step, **10** CFD and 20 CSD non-linear iterations per coupling
    - ~40 min/rev with 1024 cores
  - Control Inputs:
    - Collective (θ_{O}) and Cyclics (θ_{1c}, θ_{1s})

$$\theta_{pitch} = \theta_{O} + \theta_{1c}\cos\psi + \theta_{1s}\sin\psi$$



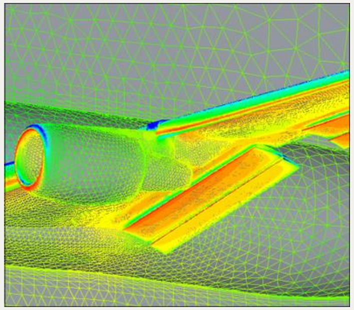


CFD flow domain

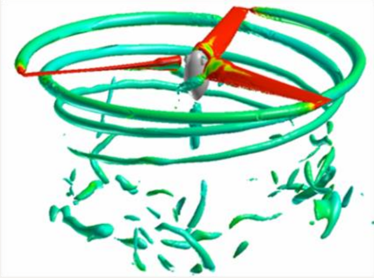


Rigid and Flexible blades

Aerodynamic Solver: NSU3D



- 3D unstructured mesh finite-volume RANS solver
- 2nd –order accurate in space and time.
- One equation Spalart-Allmaras turbulence model.
- Deforming mesh capability with GCL compliance
- Fully implicit discretization solved using Newton's method at each time-step as:



$$\mathbf{R}^n \;=\; A\frac{\partial \mathbf{U}}{\partial t} + \mathbf{S}(\mathbf{x}^n, \dot{\mathbf{x}}^n, \mathbf{U}^n) = 0$$

$$\left[\frac{\partial \mathbf{R}(\mathbf{U}^k)}{\partial \mathbf{U}^k}\right]\delta\mathbf{U}^k = -\mathbf{R}(\mathbf{U}^k)$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \delta\mathbf{U}^k$$

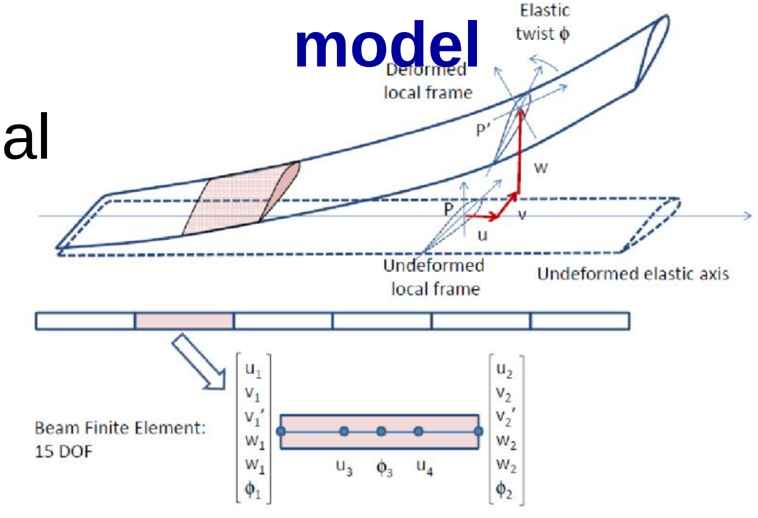
- Preconditioned GMRES used for linear system
  - Forward linearization used for exact Jacobian-vector products
- Linear agglomeration multigrid for preconditioner
- Line implicit solver as smoother for linear multigrid

Structural Analysis: Beam Model

- Hodges-Dowell type finite element based solver
- 15 degrees of freedom (flap, lag, axial and torsion)
- First order system: $\mathbf{Q} = [\mathbf{q}, \dot{\mathbf{q}}]^T$

  where, $\mathbf{J} = [I]\,\dot{\mathbf{Q}} + [A]\,\mathbf{Q} - \mathbf{F} = 0$
- **J** = Residual of structural equation
- **q** = blade dof (displacements)
- **F** = beam (aero) forcing
- **Solved via direct inversion**

Beam FEM model



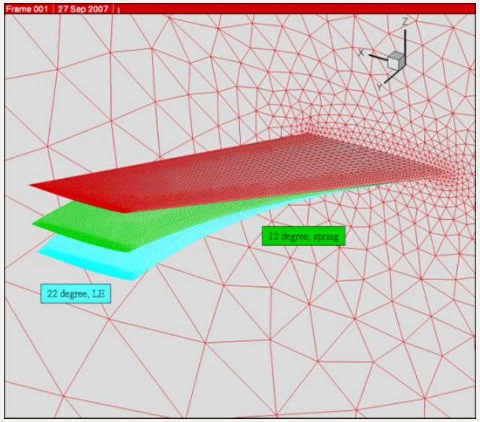
Comparison of Hart-II Natural Frequencies

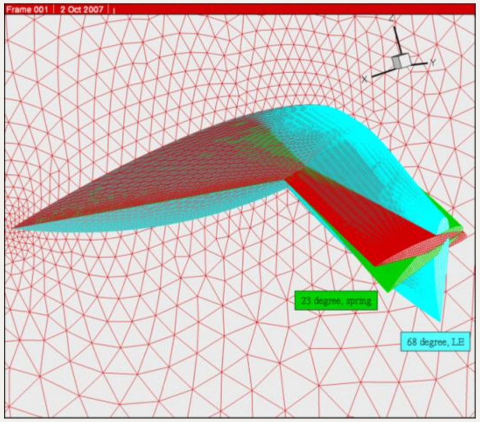
Modes	Present Model	UMARC	DLR
Flap 1	1.104	1.112	1.125
Flap 2	2.802	2.843	2.835
Flap 3	5.010	5.189	5.168
Torsion 1	3.878	3.844	3.845

Mesh Deformation

- Propagates surface displacements to interior mesh
  - Deflections from structural model at each time step (x^n)
  - Design shape changes (D)
- Based on linear elasticity analogy
  - (more robust than spring analogy)
- Solved using line-implicit agglomeration multigrid (analogous to flow solver)

$$G(x^n, x^n_{surf}, D) = 0$$





Fluid-Structure Interface (FSI)

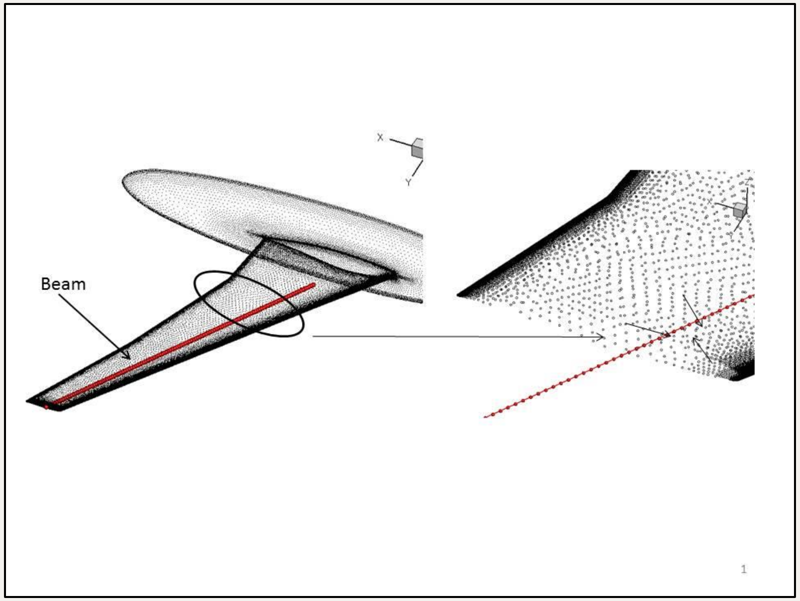
- Cloud of surface points associated with beam element
- Forces projected onto beam element shape functions

$$F_{beam} = [T(Q)]F_{cfd}(x,u) \qquad S(F_{beam}, Q, F_{cfd}(x,u)) = 0$$

- Displacements projected back to CFD surface points using transpose

$$x_{surf} = [T(Q)]^T Q$$

$$S'(x_{surf}, Q) = 0$$

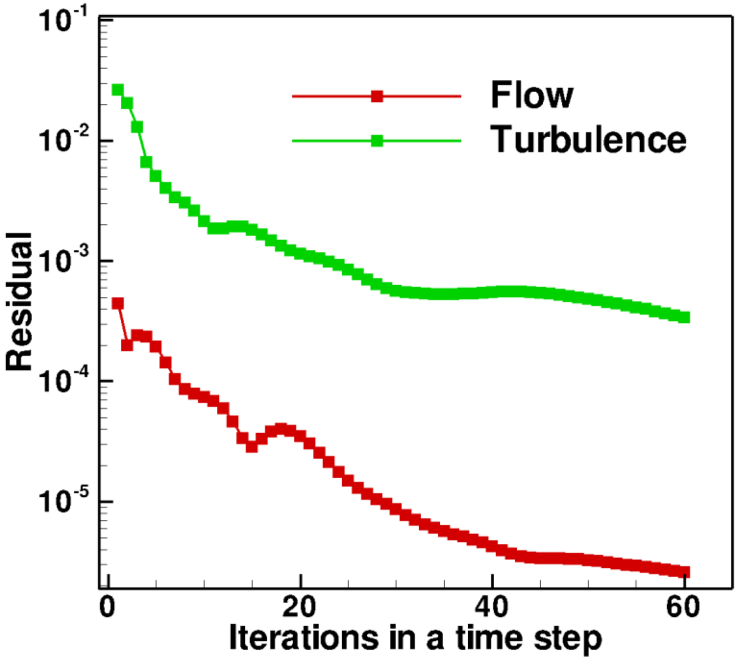


CFD/CSD Coupling Time Integration Methodology

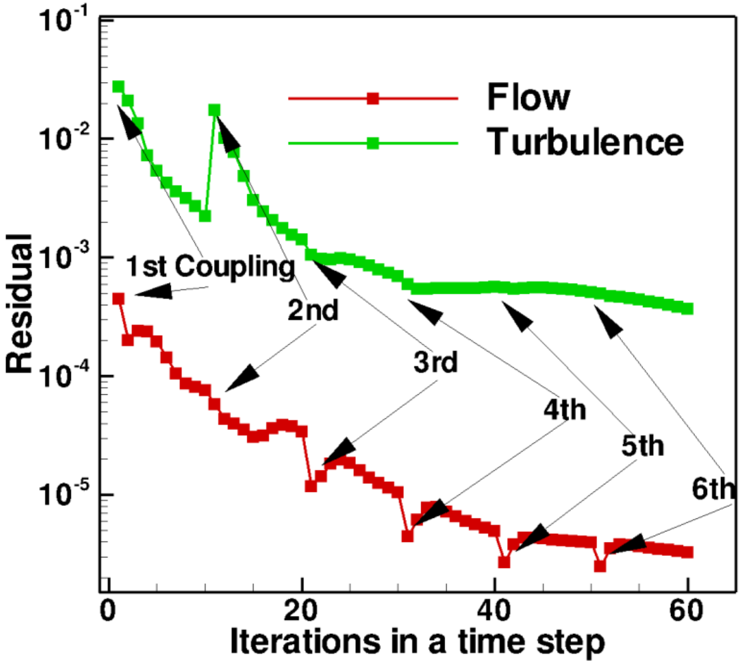
- **Outer loop over physical time steps**
  - **Coupling iterations per time step :**
    - Mesh:
      - Line implicit multigrid
    - Flow:
      - Implicit BDF2 Newton iterations (GMRES)
      - Linear agglomeration multi-grid
    - FSI (Fluid to structure)
      - Explicit assignment
    - Structure:
      - Implicit BDF2 newton iteration (direct inversion)
    - FSI (Structure to fluid)
      - Explicit assignment

Analysis Convergence

Rigid blade convergence



Flexible blade convergence

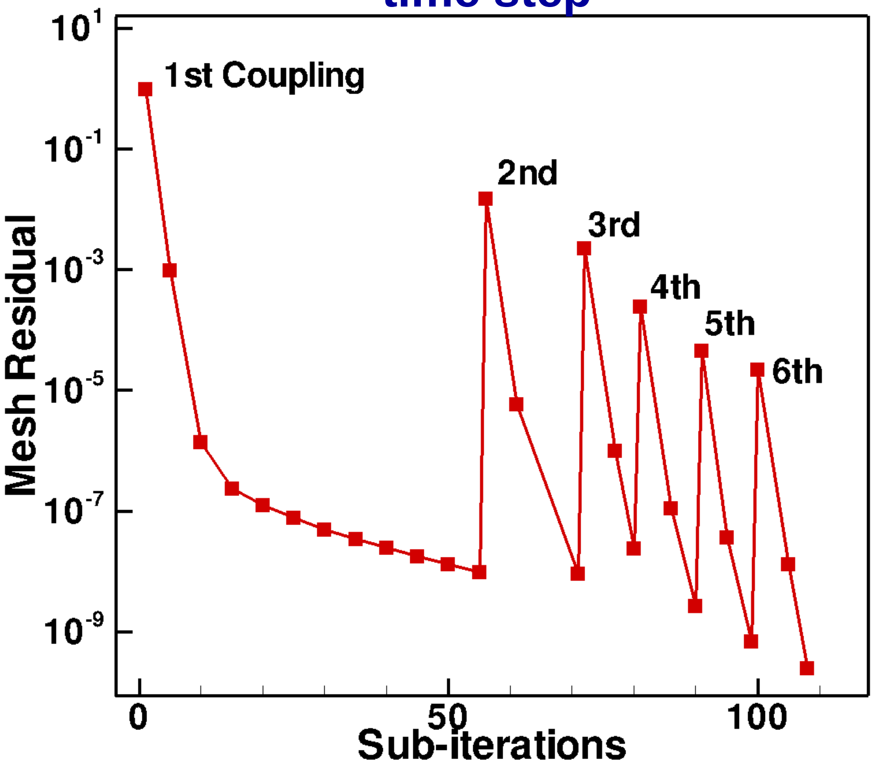


- 60 non-linear iterations per time step
- 3 multigrid cycles/iteration
- Convergence by 2 orders of magnitude

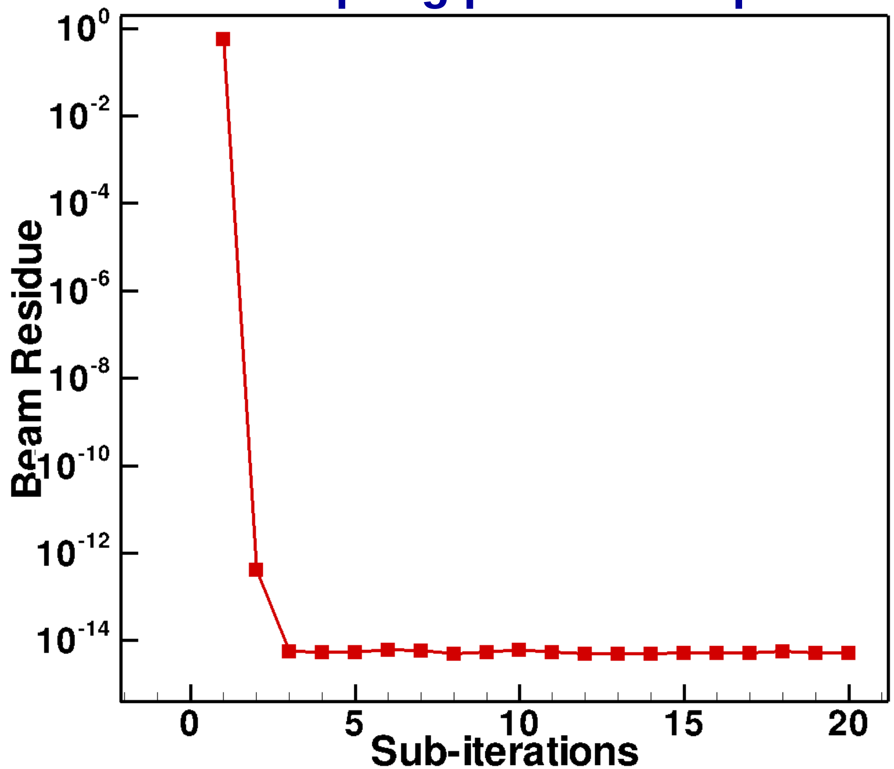
- 6 coupling cycles per time step
- 10 non-linear iterations/coupling with 3 multi-grid cycles/iteration
- Convergence by 2 orders of magnitude

Convergence continued…

Mesh convergence per time step

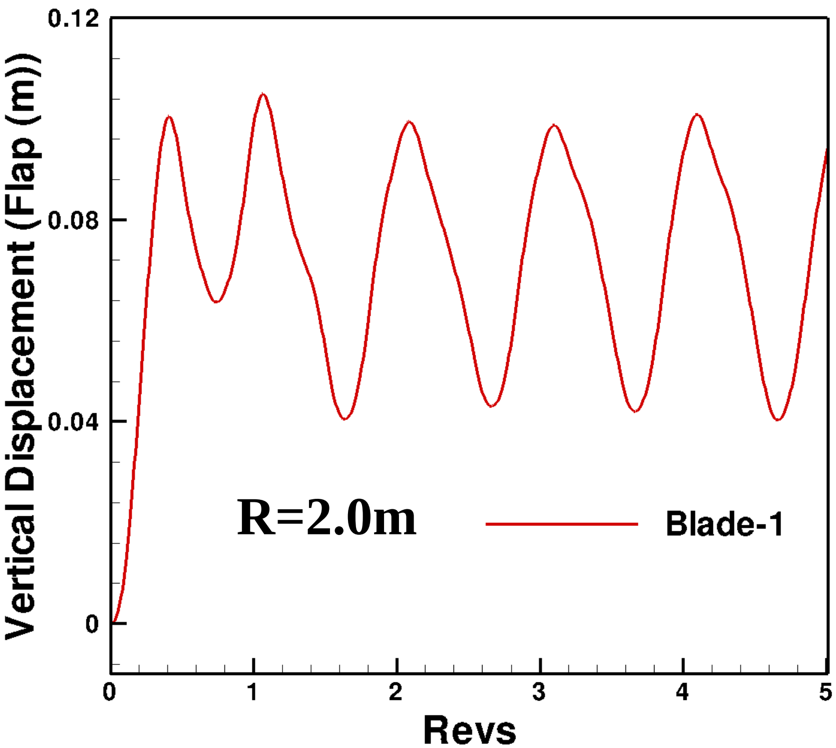


CSD convergence per coupling per time step

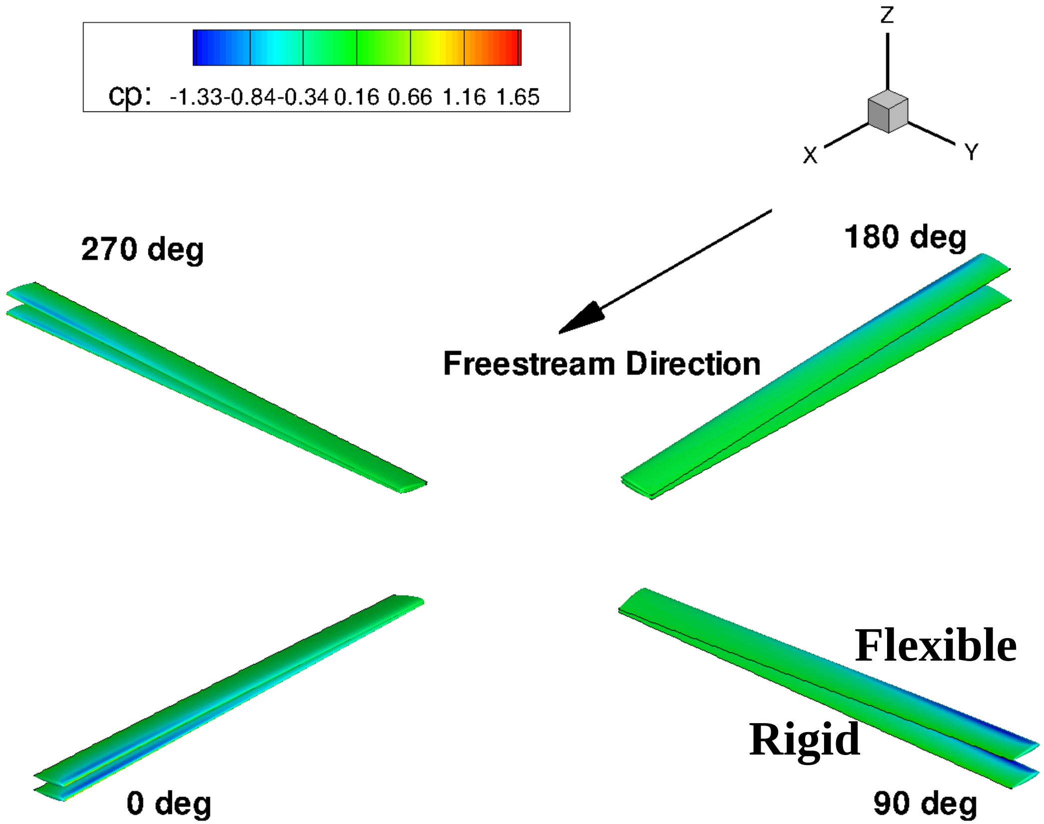


- Mesh solves upto 60 iterations or $1x10^{-9}$ (whichever earlier) per coupling
- Mesh convergence by 10 orders of magnitude per time step (6 coupling cycles)
- Beam convergence to machine precision (faster convergence)

Blade Tip Time History



Blade tip vs time



Surface Cp

- Blade flaps to high values, but converges to a lower value after 2 revs

Aerodynamic Sensitivity: Tangent

- Time-dependent objective function:

$$L = L(u^n, u^{n-1}, \ldots, u^0, x^n, x^{n-1}, \ldots, x^0)$$

- Linearizing w.r.t design variable '**D**':

$$\frac{dL}{dD} = \frac{\partial L}{\partial u^n}\frac{\partial u^n}{\partial D} + \frac{\partial L}{\partial u^{n-1}}\frac{\partial u^{n-1}}{\partial D} + \ldots + \frac{\partial L}{\partial x^n}\frac{\partial x^n}{\partial D} + \frac{\partial x^{n-1}}{\partial D} + \ldots.$$

- General expression for forward linearization of objective function w.r.t design variables:

$$\frac{dL}{d\mathbf{D}} = \begin{bmatrix} \dfrac{\partial L}{\partial \mathbf{x}} & \dfrac{\partial L}{\partial \mathbf{U}} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{x}}{\partial \mathbf{D}} \\ \dfrac{\partial \mathbf{U}}{\partial \mathbf{D}} \end{bmatrix}$$

Represents inner product over space and time

Aero Sensitivity: Tangent

Constraint equation to be satisfied: Implicit residual at each time-step.

$$G(x, D) = 0$$

$$R(U, x) = 0$$

Writing in generalized matrix form:

$$[K] \qquad \begin{bmatrix} \dfrac{\partial \mathbf{G}}{\partial \mathbf{x}} & 0 \\ \dfrac{\partial \mathbf{R}}{\partial \mathbf{x}} & \dfrac{\partial \mathbf{R}}{\partial \mathbf{U}} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{x}}{\partial \mathbf{D}} \\ \dfrac{\partial \mathbf{U}}{\partial \mathbf{D}} \end{bmatrix} = \begin{bmatrix} -\dfrac{\partial \mathbf{G}}{\partial \mathbf{D}} \\ 0 \end{bmatrix}$$

Equations dependent on **D**

Solution involves integrating forward over entire time domain

Aero Sensitivitiy: Adjoint

- Adjoint equations:

$$\begin{bmatrix} \frac{\partial \mathbf{G}}{\partial \mathbf{x}}^T & \frac{\partial \mathbf{R}}{\partial \mathbf{x}}^T \\ 0 & \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^T \end{bmatrix} \begin{bmatrix} \Lambda_\mathbf{x} \\ \Lambda_\mathbf{u} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial \mathbf{x}}^T \\ \frac{\partial L}{\partial \mathbf{U}}^T \end{bmatrix}$$

Solution involves integrating backward over entire time domain

Equations independent of **D**

- Solve adjoint system:

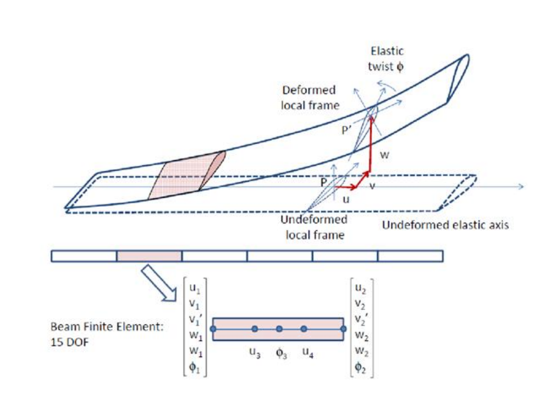
$$\begin{aligned} \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^T \Lambda_\mathbf{u} &= \frac{\partial L}{\partial \mathbf{U}}^T \\ \frac{\partial \mathbf{G}}{\partial \mathbf{x}}^T \Lambda_\mathbf{x} &= -\frac{\partial \mathbf{R}}{\partial \mathbf{x}}^T \Lambda_\mathbf{u} + \frac{\partial L}{\partial \mathbf{x}}^T \end{aligned}$$

- Finally objective sensitivity:

$$\frac{dL}{d\mathbf{D}}^T = \begin{bmatrix} \frac{\partial \mathbf{G}}{\partial \mathbf{D}}^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda_\mathbf{x} \\ \Lambda_\mathbf{u} \end{bmatrix}$$

Structural Sensitivity

- Objective function L, $\frac{dL}{d\mathbf{D}} = \frac{\partial L}{\partial \mathbf{D}} + \frac{\partial L}{\partial \mathbf{Q}} \frac{\partial \mathbf{Q}}{\partial \mathbf{D}}$
- Constraint: $\mathbf{J}(\mathbf{Q},\mathbf{F}) = \mathbf{0}$
- Tangent: $\left[\frac{\partial \mathbf{J}}{\partial \mathbf{Q}}\right] \frac{\partial \mathbf{Q}}{\partial \mathbf{D}} = -\frac{\partial \mathbf{J}}{\partial \mathbf{F}} \frac{\partial \mathbf{F}}{\partial \mathbf{D}} - \frac{\partial \mathbf{J}}{\partial \mathbf{D}}$
- Adjoint: $\left[\frac{\partial \mathbf{J}}{\partial \mathbf{Q}}\right]^T \Lambda_{\mathbf{Q}} = \frac{\partial \mathbf{L}}{\partial \mathbf{Q}}^{\mathbf{T}}$



- '**D**' can be structural or geometric shape parameters

Fully Coupled Fluid-Structure Analysis

General solution

Per coupling cycle

- Blade deformation transfer

$$\left[\frac{\partial J}{\partial Q}\right]\Delta Q$$

$$x_s = [T]^T Q^c$$

$$\left[\frac{\partial R}{\partial u}\right]\Delta u^c = -R(u^{c-1}, x^c)$$

sh: $G(x, x_s(Q)) = 0$

Flow: $R(u, x) = 0$

nsfer: $S(F_b, Q, F_{cfd}(x, u)) = 0$

Structure: $J(Q, F_b) = 0$

Blade deformation transfer: $S'(x_s, Q) = 0$

Fully Coupled Fluid-Structure Sensitivity: Tangent

- Functional sensitivity:

$$\frac{dL}{d\mathbf{D}} = \begin{bmatrix} \frac{\partial L}{\partial \mathbf{x}} & \frac{\partial L}{\partial \mathbf{U}} & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial D} \\ \frac{\partial \mathbf{U}}{\partial D} \\ \frac{\partial \mathbf{F}}{\partial D} \\ \frac{\partial \mathbf{F_B}}{\partial D} \\ \frac{\partial \mathbf{Q}}{\partial D} \\ \frac{\partial \mathbf{x_s}}{\partial D} \end{bmatrix}$$

- Solve:

$$\begin{bmatrix} \frac{\partial \mathbf{G}}{\partial \mathbf{x}} & 0 & 0 & 0 & 0 & \frac{\partial \mathbf{G}}{\partial \mathbf{x_s}} \\ \frac{\partial \mathbf{R}}{\partial \mathbf{x}} & \frac{\partial \mathbf{R}}{\partial \mathbf{u}} & 0 & 0 & 0 & 0 \\ -\frac{\partial \mathbf{F}}{\partial \mathbf{x}} & -\frac{\partial \mathbf{F}}{\partial \mathbf{u}} & I & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial \mathbf{S}}{\partial \mathbf{F}} & \frac{\partial \mathbf{S}}{\partial \mathbf{F_b}} & \frac{\partial \mathbf{S}}{\partial \mathbf{Q}} & 0 \\ 0 & 0 & 0 & \frac{\partial \mathbf{J}}{\partial \mathbf{F_b}} & \frac{\partial \mathbf{J}}{\partial \mathbf{Q}} & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial \mathbf{S'}}{\partial \mathbf{Q}} & \frac{\partial \mathbf{S'}}{\partial \mathbf{x_s}} \end{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \mathbf{D}} \\ \frac{\partial \mathbf{u}}{\partial \mathbf{D}} \\ \frac{\partial \mathbf{F}}{\partial \mathbf{D}} \\ \frac{\partial \mathbf{F_b}}{\partial \mathbf{D}} \\ \frac{\partial \mathbf{Q}}{\partial \mathbf{D}} \\ \frac{\partial \mathbf{x_s}}{\partial \mathbf{D}} \end{bmatrix} = \begin{bmatrix} -\frac{\partial \mathbf{G}}{\partial \mathbf{D}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Per coupling cycle

$$\left[\begin{bmatrix} \frac{\partial \mathbf{S}}{\partial \mathbf{F_b}} & \frac{\partial \mathbf{S}}{\partial \mathbf{Q}} \\ \frac{\partial \mathbf{J}}{\partial \mathbf{F_b}} & \frac{\partial \mathbf{J}}{\partial \mathbf{Q}} \end{bmatrix}\right] \begin{bmatrix} \frac{\partial \mathbf{F_b}}{\partial \mathbf{D}}^c \\ \frac{\partial \mathbf{Q}}{\partial \mathbf{D}}^c \end{bmatrix} = \begin{bmatrix} -\frac{\partial \mathbf{S}}{\partial \mathbf{F}}\frac{\partial \mathbf{F}}{\partial \mathbf{D}}^c \\ 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{x_s}}{\partial \mathbf{D}}^c = -\frac{\partial \mathbf{S'}}{\partial \mathbf{Q}}\frac{\partial \mathbf{Q}}{\partial \mathbf{D}}^c$$

Force
Surface
Mesh and
Transfer
Deformation
Force
Flow
and
transfer
sensitivity
Structure

Fully Coupled Fluid-Structure Sensitivity: Adjoint

- Solve:

$$
\begin{bmatrix}
\frac{\partial \mathbf{G}}{\partial \mathbf{x}}^T & \frac{\partial \mathbf{R}}{\partial \mathbf{x}}^T & -\frac{\partial \mathbf{F}}{\partial \mathbf{x}}^T & 0 & 0 & 0 \\
0 & \frac{\partial \mathbf{R}}{\partial \mathbf{u}}^T & -\frac{\partial \mathbf{F}}{\partial \mathbf{u}}^T & 0 & 0 & 0 \\
0 & 0 & I & \frac{\partial \mathbf{S}}{\partial \mathbf{F}}^T & 0 & 0 \\
0 & 0 & 0 & \frac{\partial \mathbf{S}}{\partial \mathbf{F_b}}^T & \frac{\partial \mathbf{J}}{\partial \mathbf{F_b}}^T & 0 \\
0 & 0 & 0 & \frac{\partial \mathbf{S}}{\partial \mathbf{Q}}^T & \frac{\partial \mathbf{J}}{\partial \mathbf{Q}}^T & \frac{\partial \mathbf{S}'}{\partial \mathbf{Q}}^T \\
\frac{\partial \mathbf{G}}{\partial \mathbf{x_s}}^T & 0 & 0 & 0 & 0 & \frac{\partial \mathbf{S}'}{\partial \mathbf{x_s}}^T
\end{bmatrix}
\begin{bmatrix}
\Lambda_{\mathbf{x}} \\ \Lambda_{\mathbf{u}} \\ \Lambda_{\mathbf{F}} \\ \Lambda_{\mathbf{F_b}} \\ \Lambda_{\mathbf{Q}} \\ \Lambda_{\mathbf{x_s}}
\end{bmatrix}
=
\begin{bmatrix}
\frac{\partial L}{\partial \mathbf{x}}^T \\ \frac{\partial L}{\partial \mathbf{u}}^T \\ 0 \\ 0 \\ 0 \\ 0
\end{bmatrix}
$$

Per coupling cycle

$$
\begin{bmatrix}
\frac{\partial \mathbf{G}}{\partial \mathbf{x}}^T & \frac{\partial \mathbf{R}}{\partial \mathbf{x}}^T \\
0 & \frac{\partial \mathbf{R}}{\partial \mathbf{u}}^T
\end{bmatrix}
\begin{bmatrix}
\Lambda_{\mathbf{x}}{}^c \\ \Lambda_{\mathbf{u}}{}^c
\end{bmatrix}
=
\begin{bmatrix}
\frac{\partial L}{\partial \mathbf{x}}^T + \frac{\partial \mathbf{F}}{\partial \mathbf{x}}^T \Lambda_{\mathbf{F}}{}^c \\
\frac{\partial L}{\partial \mathbf{u}}^T + \frac{\partial \mathbf{F}}{\partial \mathbf{u}}^T \Lambda_{\mathbf{F}}{}^c
\end{bmatrix}
$$

Fully Coupled Fluid-Structure Sensitivity: Adjoint

- Solve:

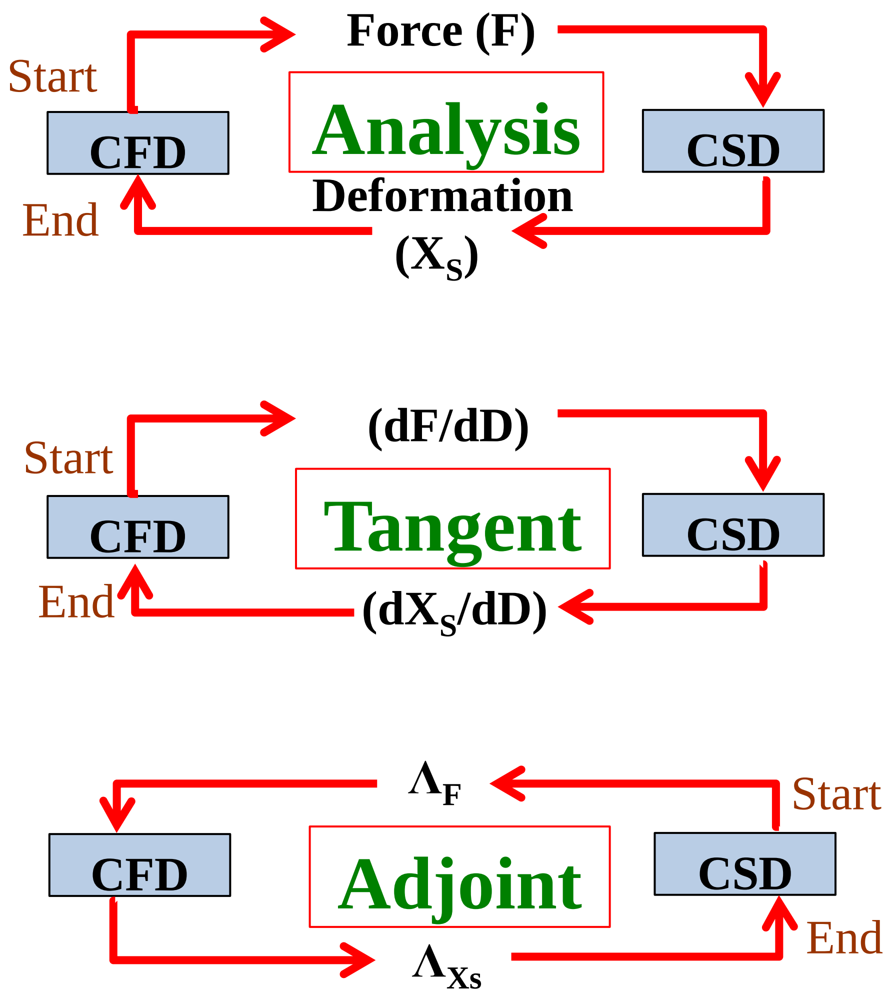
$$
\left[\begin{array}{c:c:c:c:c:c}
\frac{\partial \mathbf{G}}{\partial \mathbf{x}}^T & \frac{\partial \mathbf{R}}{\partial \mathbf{x}}^T & -\frac{\partial \mathbf{F}}{\partial \mathbf{x}}^T & 0 & 0 & 0 \\ \hdashline
0 & \frac{\partial \mathbf{R}}{\partial \mathbf{u}}^T & -\frac{\partial \mathbf{F}}{\partial \mathbf{u}}^T & 0 & 0 & 0 \\ \hdashline
0 & 0 & I & \frac{\partial \mathbf{S}}{\partial \mathbf{F}}^T & 0 & 0 \\ \hdashline
0 & 0 & 0 & \frac{\partial \mathbf{S}}{\partial \mathbf{F_b}}^T & \frac{\partial \mathbf{J}}{\partial \mathbf{F_b}}^T & 0 \\ \hdashline
0 & 0 & 0 & \frac{\partial \mathbf{S}}{\partial \mathbf{Q}}^T & \frac{\partial \mathbf{J}}{\partial \mathbf{Q}}^T & \frac{\partial \mathbf{S}'}{\partial \mathbf{Q}}^T \\ \hdashline
\frac{\partial \mathbf{G}}{\partial \mathbf{x_s}}^T & 0 & 0 & 0 & 0 & \frac{\partial \mathbf{S}'}{\partial \mathbf{x_s}}^T
\end{array}\right]
\begin{bmatrix} \Lambda_{\mathbf{x}} \\ \Lambda_{\mathbf{u}} \\ \Lambda_{\mathbf{F}} \\ \Lambda_{\mathbf{F_b}} \\ \Lambda_{\mathbf{Q}} \\ \Lambda_{\mathbf{x_s}} \end{bmatrix}
=
\begin{bmatrix} \frac{\partial L}{\partial \mathbf{x}}^T \\ \frac{\partial L}{\partial \mathbf{u}}^T \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

Per coupling cycle

$$
\frac{dL}{dD}^T = \begin{bmatrix} \frac{\partial \mathbf{G}}{\partial D}^T & 0 & 0 & 0 & \cancel{\frac{\partial \mathbf{J}}{\partial D}^T} & 0 \end{bmatrix}
\begin{bmatrix} \Lambda_{\mathbf{x}} \\ \Lambda_{\mathbf{u}} \\ \Lambda_{\mathbf{F}} \\ \Lambda_{\mathbf{F_B}} \\ \Lambda_{\mathbf{Q}} \\ \Lambda_{\mathbf{x_s}} \end{bmatrix}
$$

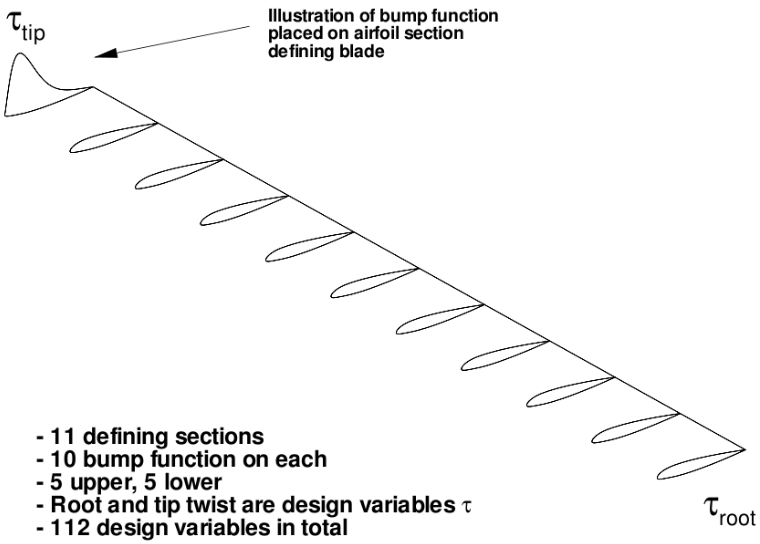
Coupling Schematic

Same data structures



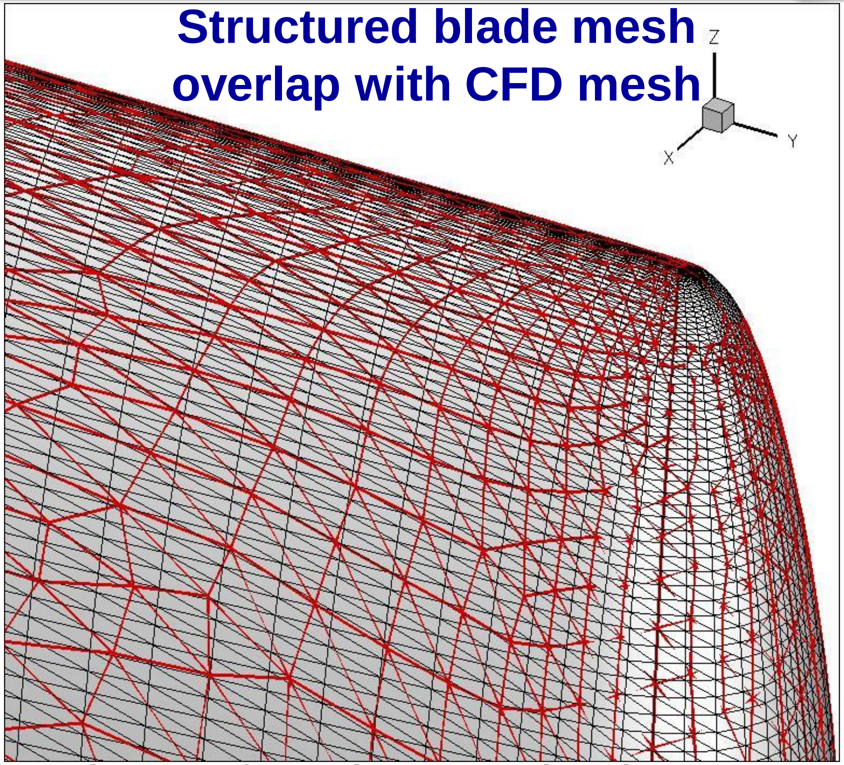
Same solution strategy

Blade Geometry Parametrization

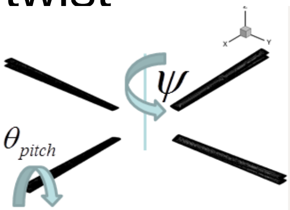


Hicks-Henne bump functions

Structured blade mesh overlap with CFD mesh



- Master blade shape defined by Hicks-Henne bump functions and twist
  - Defined by high-resolution structured mesh (in black)
  - Shape changes interpolated onto unstructured CFD surface mesh
- 115 design parameters
  - 10 Hicks-Henne bump fcts per blade section, 11 blade sections (110)
  - Twist at blade root and tip (2) and 3 pitch parameters



Unsteady Objective Function

$$\min L = L_{power} + \omega L_{trim}$$

$$L_{power} = \frac{1}{T}\sum_{n=1}^{n=N} \Delta t\left[\delta C_Q^n\right]^2$$

$$L_{trim} = \omega_1 \frac{1}{T}\sum_{n=1}^{n=N} \Delta t\left[\delta C_T^n\right]^2 + \omega_2\left[\delta \overline{C}_{Mx}\right]^2 + \omega_3\left[\delta \overline{C}_{My}\right]$$

$$\delta C_Q^n = (C_Q^n - C_{Q_{TARGET}}^n)$$

$$\delta C_T^n = (C_T^n - \overline{C}_{T_{TARGET}}^n)$$

$$\delta \overline{C}_{My} = \frac{1}{T}\sum_{n=1}^{n=N} \Delta t(C_{My}^n - C_{My_{TARGET}}^n)$$

$$\delta \overline{C}_{Mx} = \frac{1}{T}\sum_{n=1}^{n=N} \Delta t(C_{Mx}^n - C_{Mx_{TARGET}}^n)$$

- Trim
  - Target thrust = 4.4e-3
  - Target C_{Mx} C_{My}=0.0
- Performance
  - Target power = 0.0

Unsteady Strongly Coupled CFD/CSD Adjoint Sensitivity Verification

- Tangent and adjoint verification by perturbing collective pitch (i.e. $\mathbf{D}=\theta_O$)
- Complex perturbation of value $1\text{x}10^{-100}$
- $\frac{\partial L^n}{\partial \boldsymbol{D}}$ from **tangent** and **adjoint** verified with **complex step method** every time step
- Three sensitivity analysis formulations converged to machine zero every time step

Verification of Strongly Coupled CFD/CSD Adjoint Formulation

- Verified to 10 significant digits
- Verified over multiple time steps
- Accuracy preserved over multiple time-steps

Time step	Method	$\frac{\partial L^n}{\partial D}$
1	Complex	7.569817143673**123**E-005
	Tangent	7.569817143673**061**E-005
	Adjoint	7.569817143672**761**E-005
2	Complex	6.040142774935**852**E-005
	Tangent	6.040142774935**835**E-005
	Adjoint	6.040142774935**570**E-005
3	Complex	-4.959909870786**381**E-006
	Tangent	-4.959909870787**765**E-006
	Adjoint	-4.959909870785**228**E-006
5	Complex	-1.142069116982**308**E-004
	Tangent	-1.142069116982**308**E-004
	Adjoint	-1.142069116982**432**E-004
180	Complex	-5.176189427439**016**E-003
	Tangent	-5.176189427439**005**E-003
	Adjoint	-5.176189427434**507**E-003

Optimization Procedure

1:Trim rotor

- Minimize L_{trim} (drive to 0)
- Using only control inputs as design parameters

2: Perform optimization

- Minimze $L_{design}=L_{performance} + L_{trim}$
- Using shape parameters + control inputs (design)

3: Retrim shape-optimized rotor

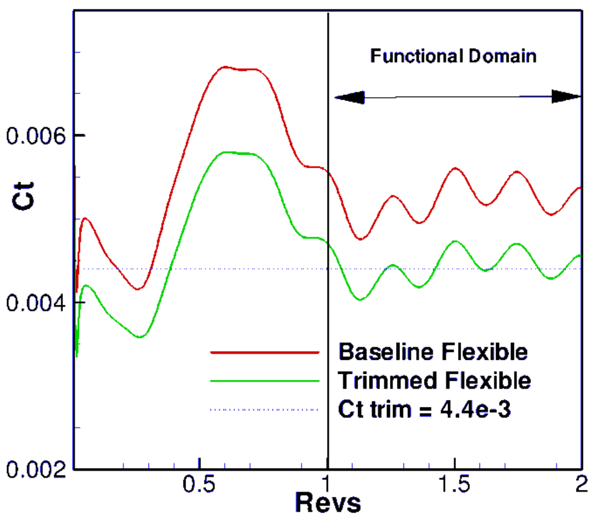
- $L_{trim}=0$ not maintained exactly in design process (implemented as penalty term)

- First perform for rigid rotor, then flexible rotor

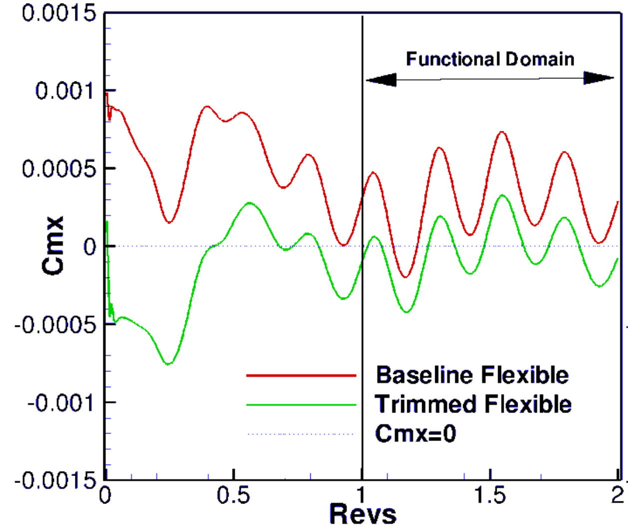
Strongly Coupled Optimization on Hart-II Rotor: 2.32M Grid

- Optimization (trim/shape) over 2 rev: $\Delta t = 2.0^{o}$
- Optimizer: L-BFGS-B bounded reduced Hessian
- Bounds for shape parameters:
  - $\pm 5\%$ chord on airfoil section
  - $\pm 1.0^{o}$ twist
- Bounds for control inputs
  - $\theta_o, \theta_{1c}, \theta_{1s} : \pm 5.0^{o}$
- 6 CFD-CSD coupling cycles/time-step, 10 Newton cycles/coupling
- Run in parallel using 1024 cores
  - 70 minutes per rotor revolution

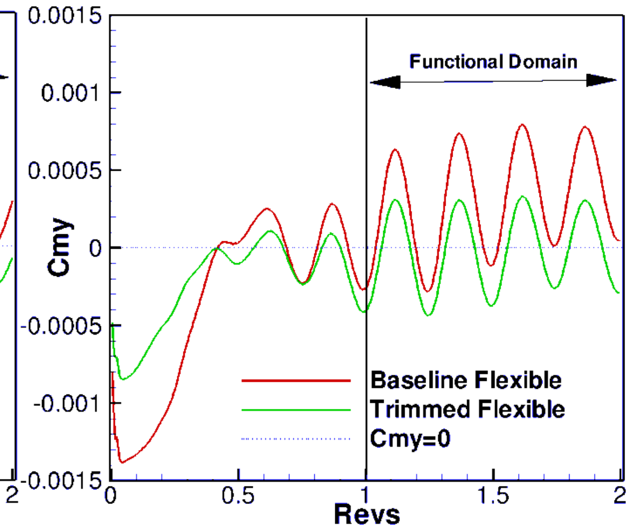
Flexible Hart-II Forward Flight Trim



Thrust vs time



Lateral Moment vs time

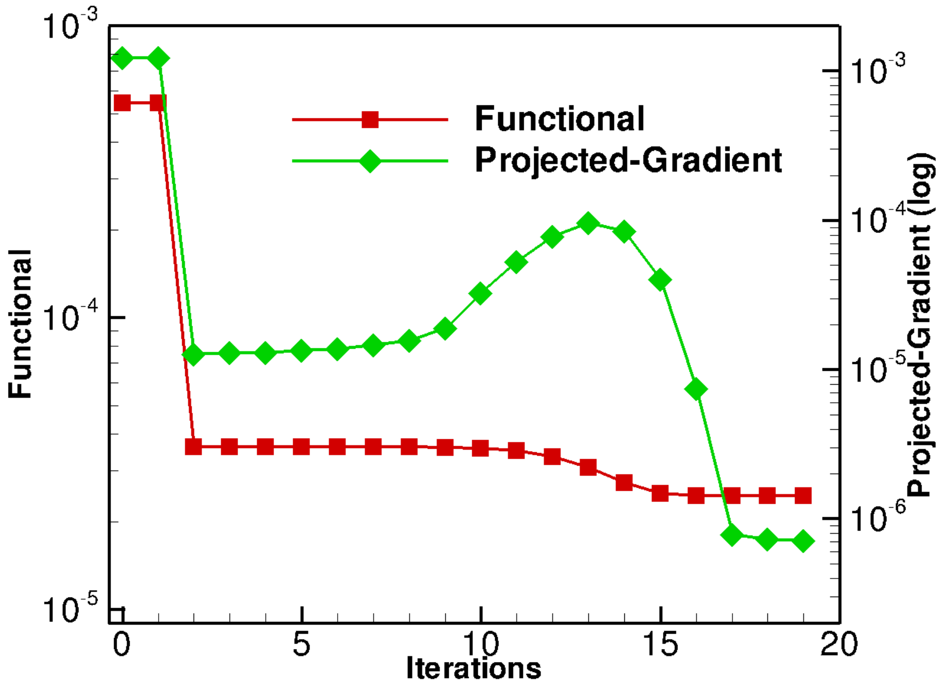


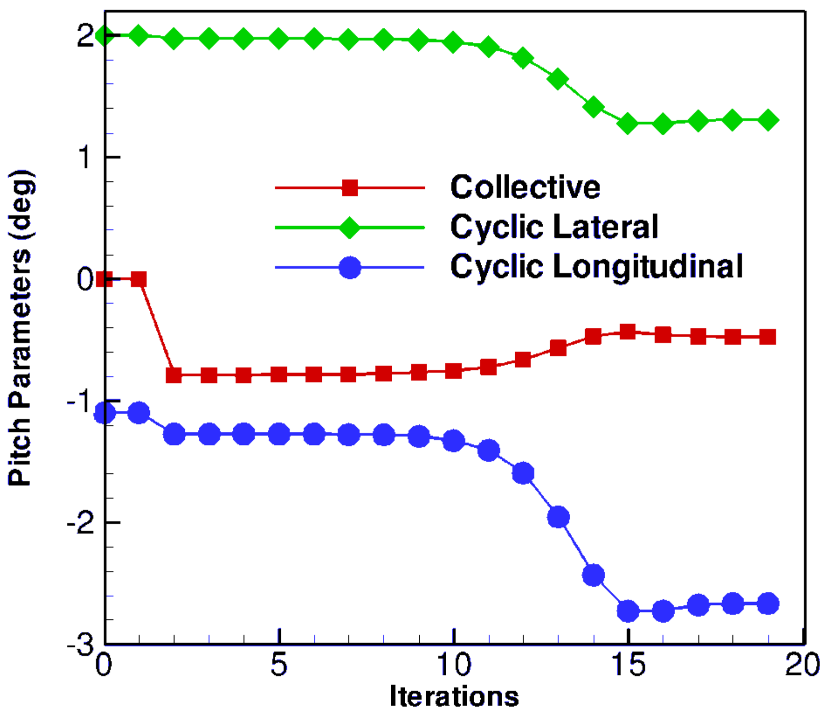
Long. Moment vs time

- Trimmed to target mean thrust (Ct=4.4e-3), zero moments (~1e-5)

Pitch (deg)	Experiment	HOST	Present (Flexible)
θ_O (Collective)	3.20	4.91	4.56
θ_{1c} (Lat. Cyclic)	2.00	1.41	1.28
θ_{1s} (Long. Cyclic)	-1.10	-1.34	-2.72

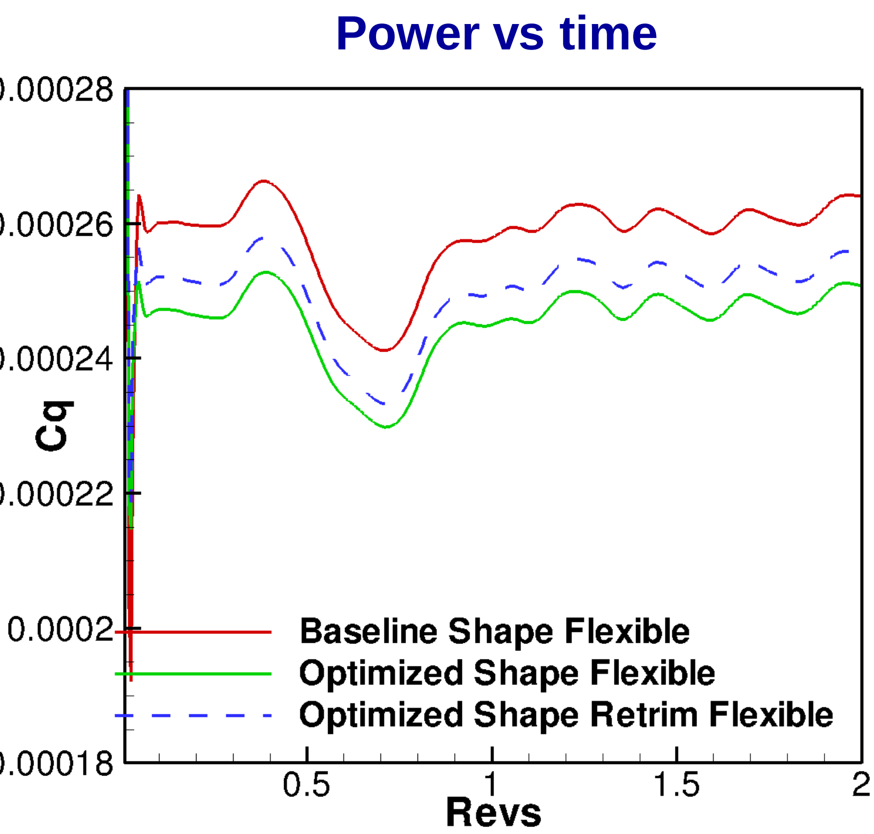
Flexible Rotor Trim Convergence

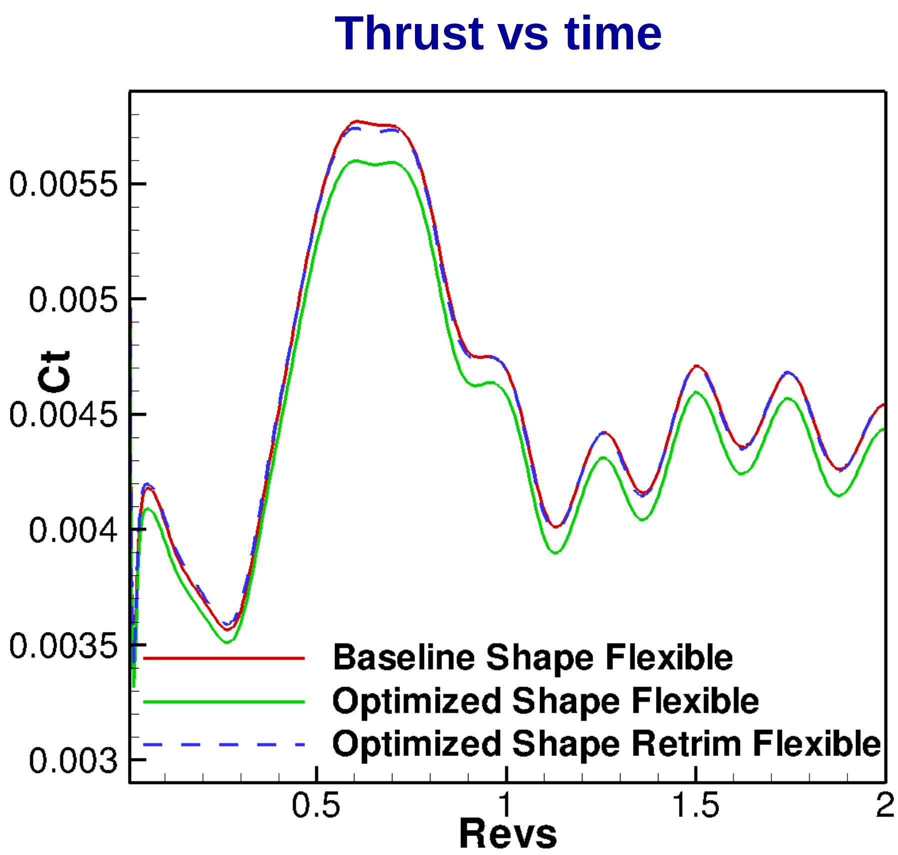




- Gradient drops more than 2 orders
- Objective converges by ~15 iterations
- Consistent pitch parameter convergence

Flexible Hart-II Shape Optimization





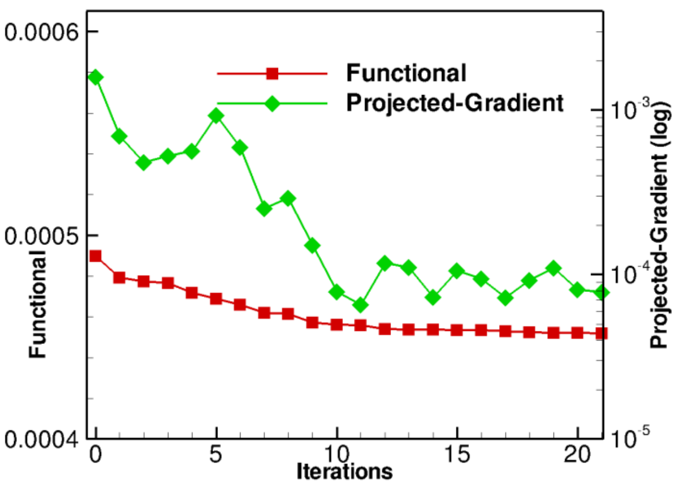
- Trimmed to target mean thrust (Ct=4.4e-3), zero moments (~1e-5)
- Overall ~**3.1**% power reduction w/ shape optimization after retrim

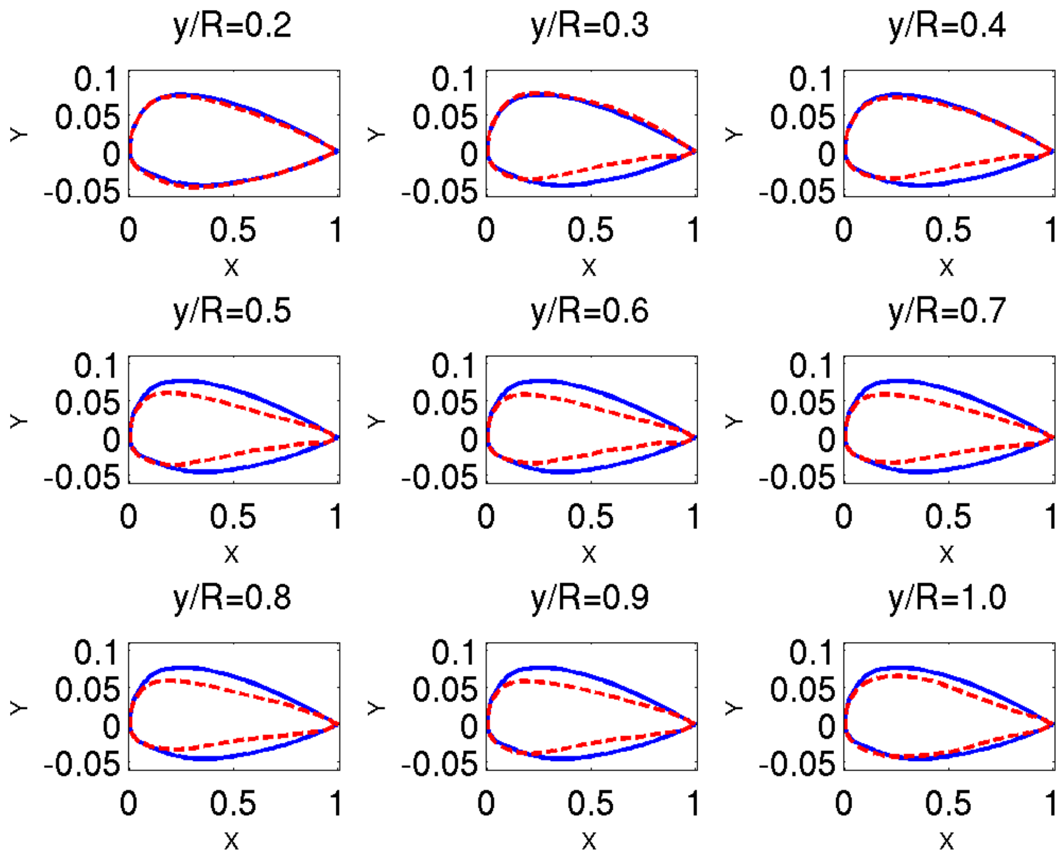
Flexible Optimized Blade Shape

Baseline

Optimized

Thicker inboard and thinner outboard sections





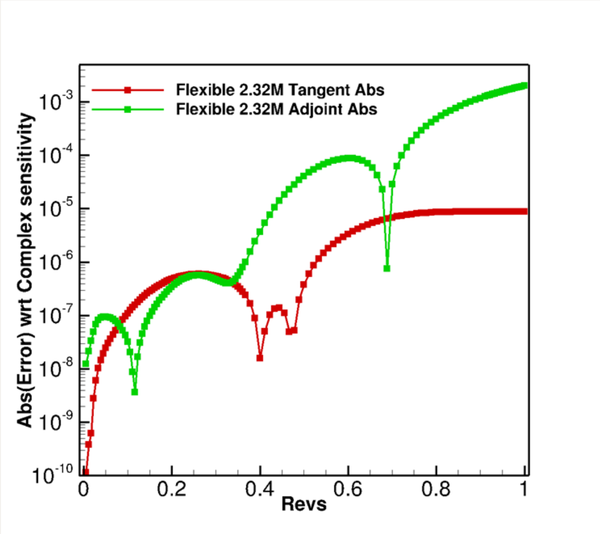
- 1 and ½ order gradient drop, objective plateaus

Conclusions

- Adjoint methods can be extended to problems of arbitrary complexity
  - Time-dependent
  - Multidisciplinary
- Requires methodical approach
  - Analogous to analysis formulation
  - Same data-structures
  - Reuse coupled disciplinary solvers
- Using discrete adjoint
  - Straight forward conceptually
  - Tedious

Conclusions

- Reverse time integration not large issue for URANS calculations with large implicit time steps
- Accuracy of gradients with partial convergence remains poorly understood



- Chaotic adjoint behavior possible with time-dependent problems
  - Not likely for URANS with large separation of scales
  - Occurs for turbulence resolving scales (LES)
  - Additional techniques required (Q. Wang 2013)