Part II: Plant-scale aeroelastically-coupled wind turbine response from geometrically exact beam theory

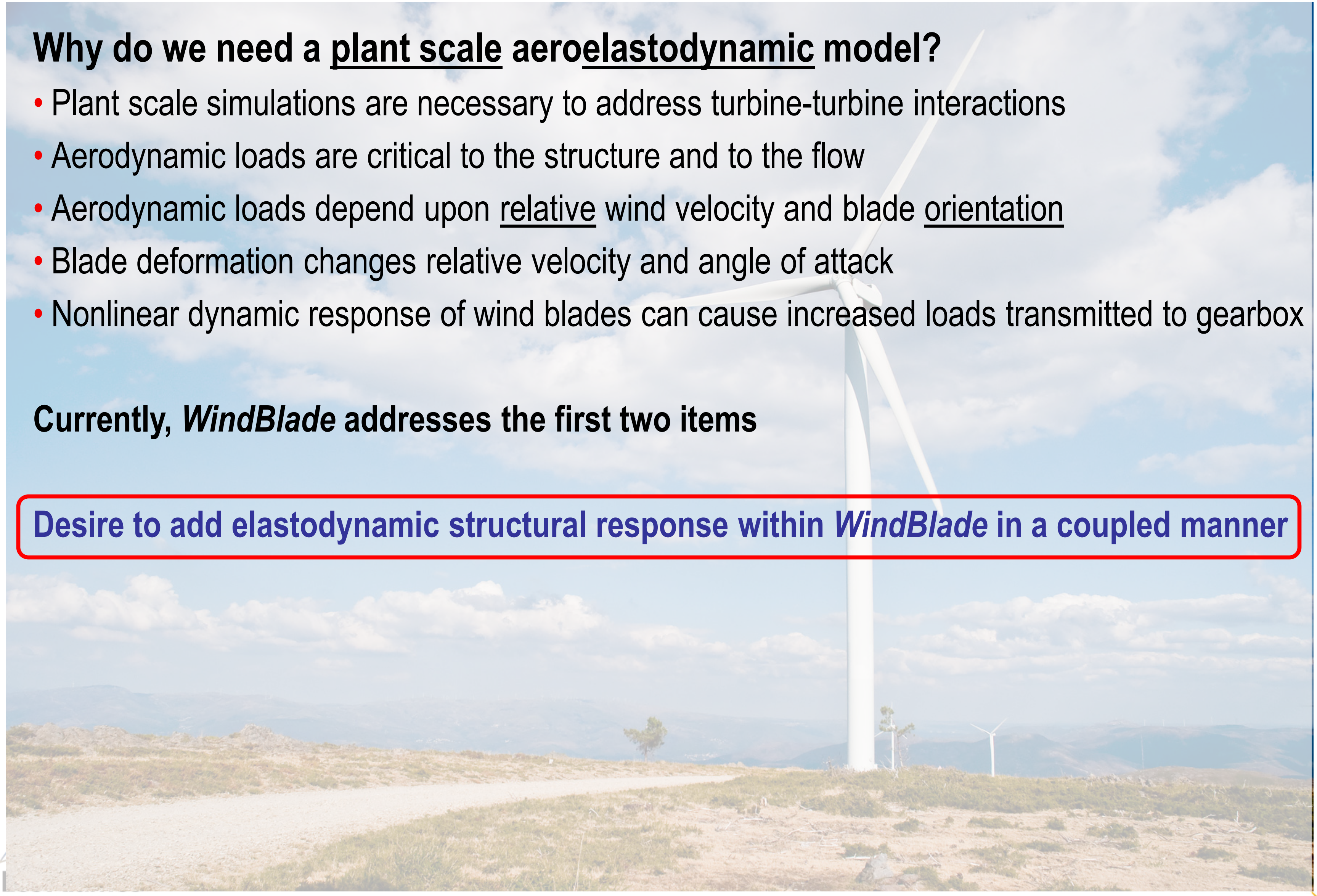
IWT LDRD Mid-Project Review
March 2, 2011

D.J. Luscher, T-3
Gretchen Ellis, AET-1

Los Alamos National Laboratory
Theoretical Division
Fluid Dynamics and Solid Mechanics (T-3)
MS B216, P.O. Box 1663, Los Alamos, NM 87545
505 665-2848, djl@lanl.gov

LA-UR-11-01406

Motivation for including elastic response within WindBlade



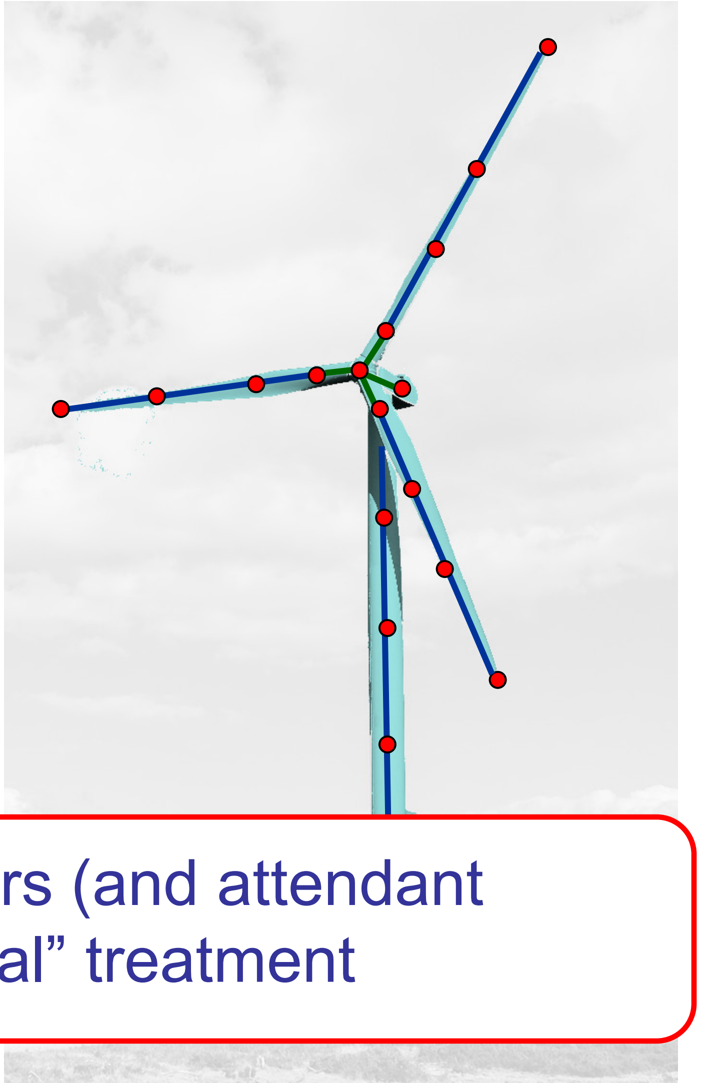
Why do we need a plant scale aeroelastodynamic model?

- Plant scale simulations are necessary to address turbine-turbine interactions
- Aerodynamic loads are critical to the structure and to the flow
- Aerodynamic loads depend upon relative wind velocity and blade orientation
- Blade deformation changes relative velocity and angle of attack
- Nonlinear dynamic response of wind blades can cause increased loads transmitted to gearbox

Currently, *WindBlade* addresses the first two items

Desire to add elastodynamic structural response within *WindBlade* in a coupled manner

Approach for including elastodynamics



Model each wind turbine as deformable body within WindBlade

- Tower and blades → Geometrically exact beam theory
- Gearbox and hub as nonlinear constraints
- Permits dynamic pitch and yaw control
- Readily extendable to offshore applications

Other approaches:

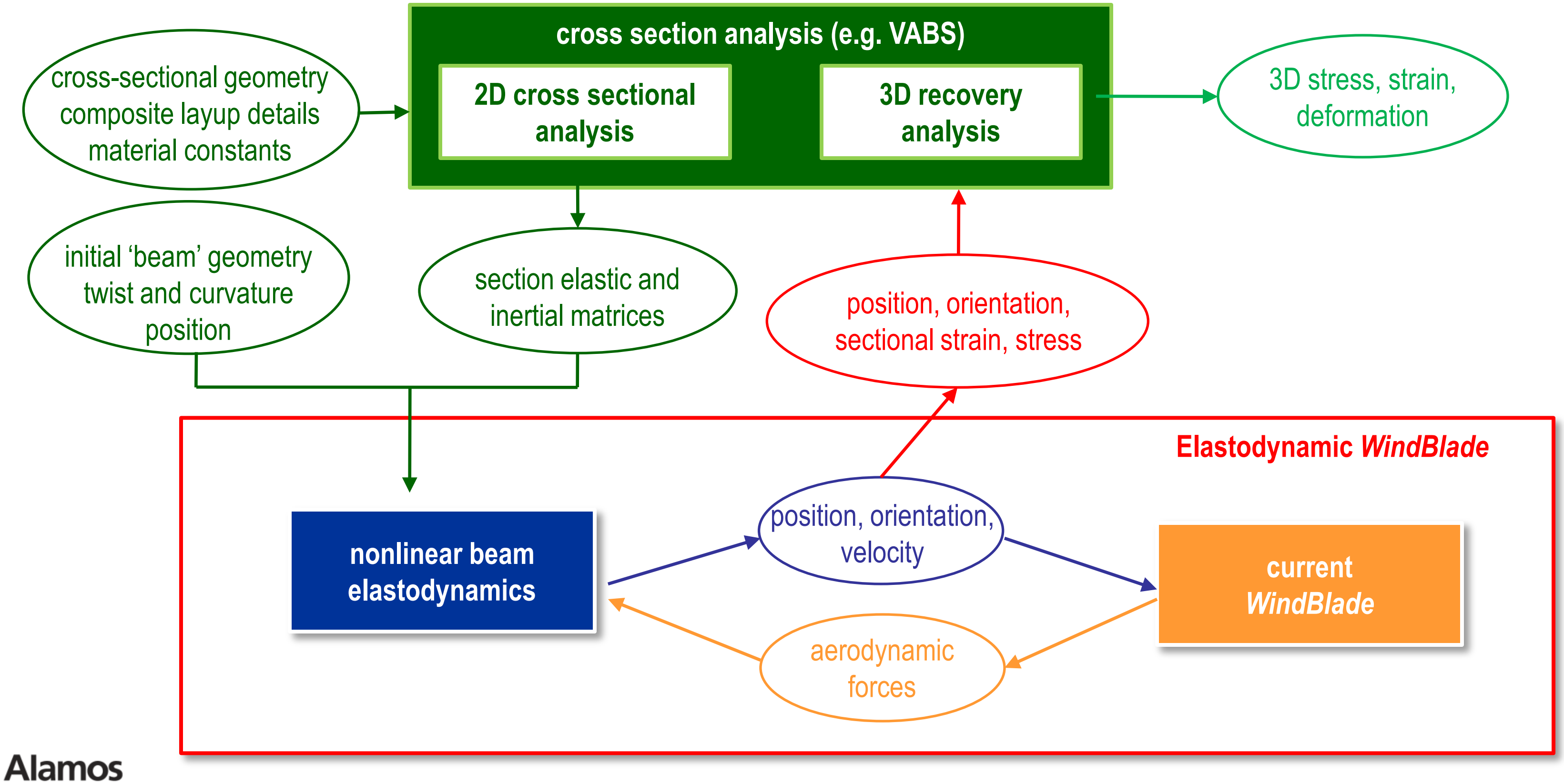
- Direct spatial coupling between deforming solid/shell Lagrange mesh and ALE fluid grid
- Modal dynamic (linear!!) FE methods

Anticipation of increasingly large rotors (and attendant deformation) motivates “beyond modal” treatment

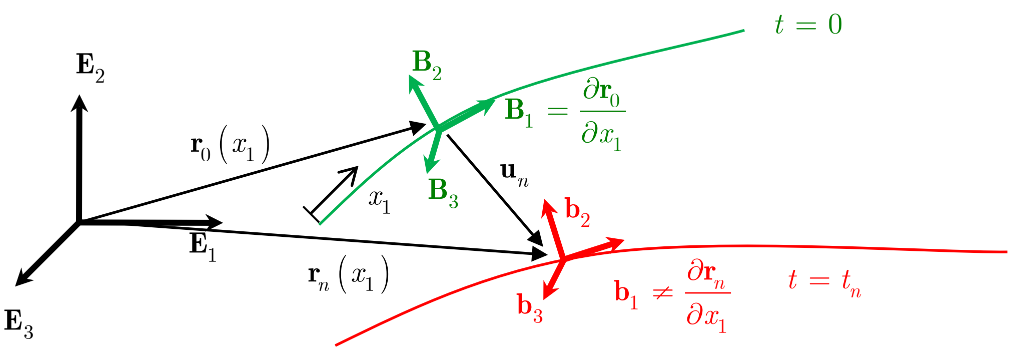
Conceptual overview of coupling strategy

Basic approach:

- Geometrically exact beam theory (fully nonlinear kinematics)
- Loose two-way coupling to aero within *WindBlade*
- Hodges et al. asymptotically correct analysis of anisotropic, composite cross-sections



Nonlinear (geometrically exact) beam theory



Beam Kinematics:

$$\mathbf{B}_i = \mathbf{\Lambda}_0(x_1)\cdot\mathbf{E}_i$$

$$\mathbf{b}_i = \mathbf{\Lambda}_n(x_1)\cdot\mathbf{B}_i = \mathbf{\Lambda}_n(x_1)\cdot\mathbf{\Lambda}_0(x_1)\cdot\mathbf{E}_i$$

$$\mathbf{\Lambda} = \mathbf{\Lambda}_n\cdot\mathbf{\Lambda}_0$$

$$\mathbf{R}_n(x_1,x_2,x_3) = \mathbf{r}_n(x_1) + \mathbf{\Lambda}(x_1)\cdot\begin{Bmatrix}0\\ x_2\\ x_3\end{Bmatrix}$$

Strain Measures:

$\boldsymbol{\gamma}_n = \mathbf{\Lambda}^T\mathbf{r}_n' - \mathbf{b}_1$ Sectional strains (axial, transverse shear)

$\boldsymbol{\kappa}_n = \mathbf{\Lambda}^T\mathbf{\Lambda}_n'$ Sectional curvature (torsional rate of twist, bending curvatures)

Sectional Forces (and Moments): from generally coupled strain-energy density (for anisotropic composite beams)

$U = \dfrac{1}{2}\begin{Bmatrix}\boldsymbol{\gamma}\\ \boldsymbol{\kappa}\end{Bmatrix}^T[C]\begin{Bmatrix}\boldsymbol{\gamma}\\ \boldsymbol{\kappa}\end{Bmatrix}$ $\mathbf{F}_N = \dfrac{\partial U}{\partial\boldsymbol{\gamma}}$ sectional forces $\mathbf{F}_M = \dfrac{\partial U}{\partial\boldsymbol{\kappa}}$ sectional moments

Weak Form of Momentum Conservation: $\mathbf{R}_I = \mathbf{R}_I^m + \mathbf{R}_I^d - \mathbf{R}_I^e = \mathbf{0}$ Total residual → 0

Inertial contribution to nodal forces: $\mathbf{R}_I^d = \displaystyle\int_0^L N_I(x)\begin{Bmatrix}\rho A\,\ddot{\mathbf{u}}\\ \dot{\boldsymbol{\pi}}\end{Bmatrix}dx$ $\dot{\boldsymbol{\pi}} = \widetilde{\mathbf{W}}\mathbf{J}_\rho\mathbf{W} + \mathbf{J}_\rho\mathbf{A}$

Stress contribution to nodal forces: $\mathbf{R}_I^m = \displaystyle\int_0^L \begin{bmatrix}N_I'\,\mathbf{I} & \mathbf{0}\\ -N_I\,\widetilde{\mathbf{r}}' & N_I'\,\mathbf{I}\end{bmatrix}\begin{Bmatrix}\mathbf{\Lambda}\cdot\mathbf{F}_N\\ \mathbf{\Lambda}\cdot\mathbf{F}_M\end{Bmatrix}dx$

Nodal force due to external loads: $\mathbf{R}_I^e = \displaystyle\int_0^L \begin{Bmatrix}N_I\,\rho A\,\mathbf{g}\\ \mathbf{0}\end{Bmatrix}dx + \sum N_I(x_1^*)\begin{Bmatrix}\mathbf{\Lambda}\cdot\mathbf{f}_{aero}\\ \mathbf{\Lambda}\cdot\mathbf{T}_{aero}\end{Bmatrix}$

Implementation

Two versions:

- Goal is to implement as Fortran subroutine library called from *WindBlade*
- In order to debug, evaluate, and test competing algorithms → python protype

Current Status (implemented):

- Fully nonlinear kinematics, conformal rotation vector, objective interpolation
- Petrov-Galerkin FE implementation (test function ≠ trial function)
- Newton-Raphson solver with Lagrange constraint enforcement
- Newmark two-parameter time integration
- Temporal and spatial interpolation of aerodynamic forces (permits dissimilar dt and dx)
- Gravity (body forces)

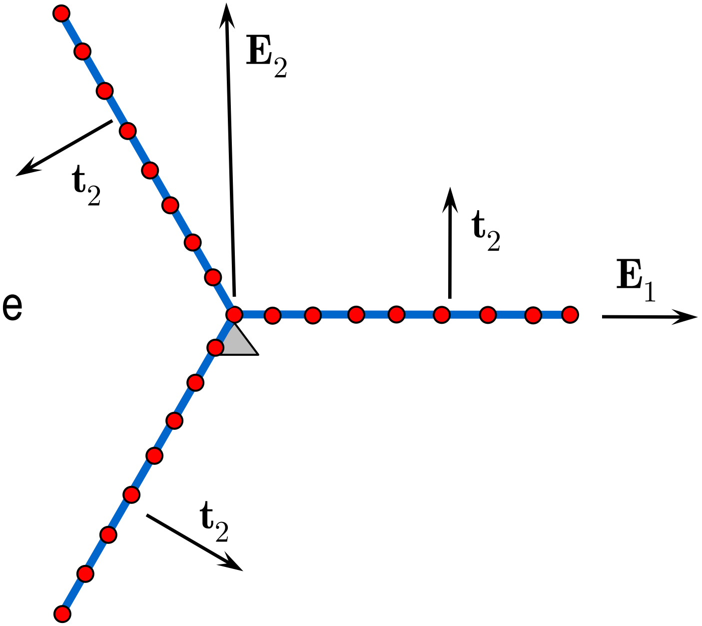
To do:

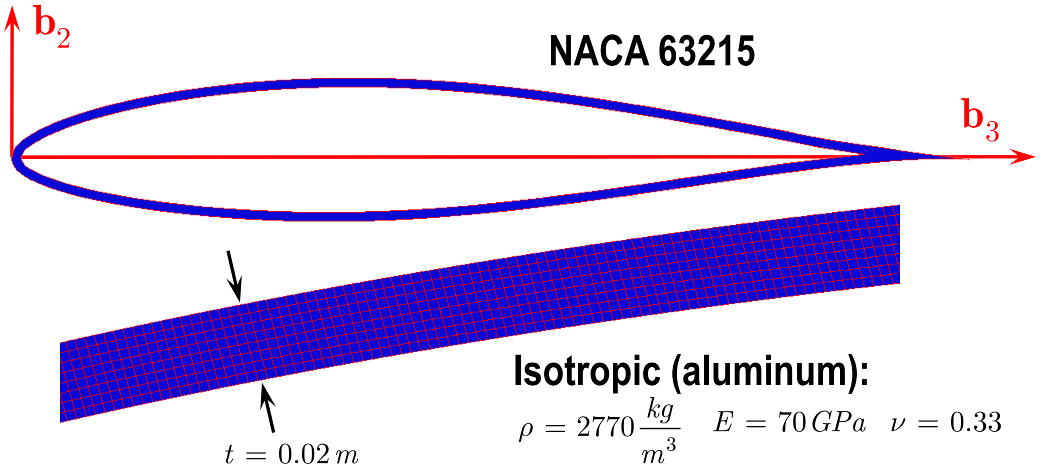
- Aerodynamic torsion from offset aero/shear centers
- Hub/gearbox/tower constraints
- Generalized alpha time integration
- Extension for large initial twist/curvature

Example: One way coupling for prismatic blade rotor

Simple example:

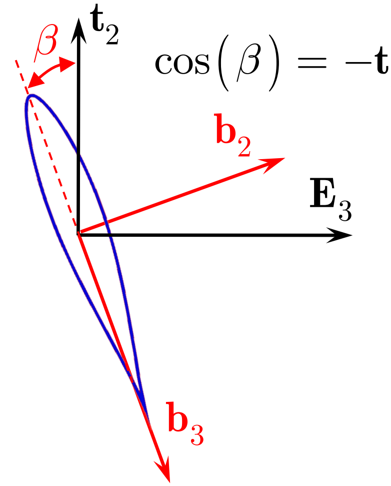
- Challenge theory and numerical implementation
- Aerodynamic forces from *WindBlade*
- Free-hub aerodynamic “spin-up” with gravity
- Highlight some important aspects of nonlinear dynamic response
    - large deformation → pitch angle





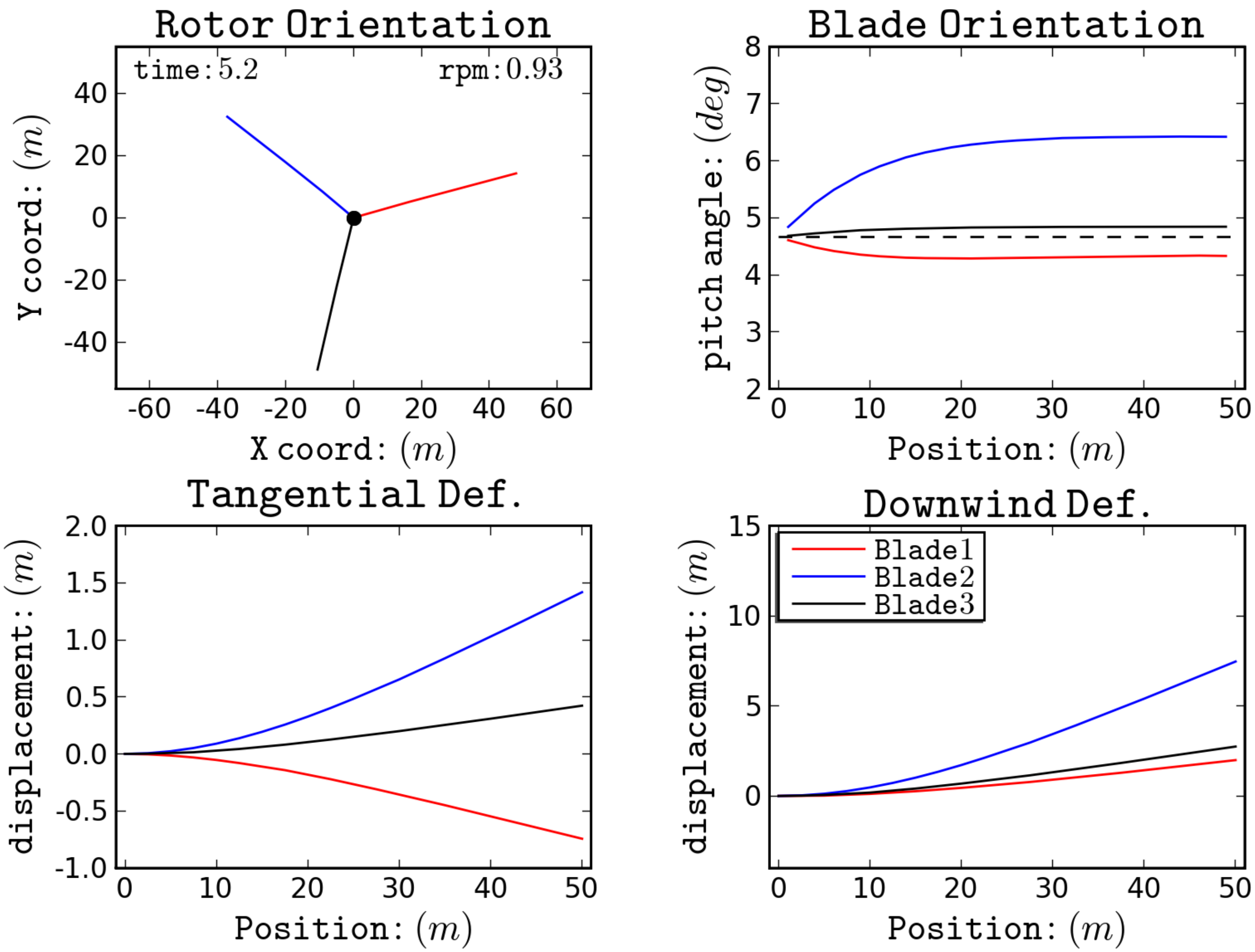
Isotropic (aluminum):

$\rho = 2770\dfrac{kg}{m^3} \quad E = 70\,GPa \quad \nu = 0.33$



$$\cos(\beta) = -\mathbf{t}_2 \cdot \mathbf{b}_3$$

Example: Results for aero induced spin up



Los Alamos NATIONAL LABORATORY EST. 1943

IWT LDRD DR

Summary

Geometrically exact beam theory for blades/towers:

- Current implementation (python) can handle:
  - Anisotropic materials
  - Material and geometric coupling: e.g. bend-twist, axial-twist
  - Offset aero, mass, and shear centers
  - Initial curvature and twist
- Extendable to include (significant future work):
  - Elastic cross-section warping effects
  - Damage

Gearbox and hub modeled by nonlinear constraint equations:

- Enables dynamic pitch and yaw control
- Can include transmission compliance, generator power, and mechanical loss

Turbine nonlinear elastodynamic response:

- Computational costs are larger than modal based approaches, but not excessive
- Easily amenable to offshore wind turbine modeling
- Approach could be used in conjunction with other aerodynamic loading models