DTU Library

Higher Fidelity Analysis in Wind Turbine Multi-disciplinary Design Optimization

McWilliam, Michael

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

Higher Fidelity Analysis in Wind Turbine Multi-disciplinary Design Optimization

Michael McWilliam

Danish Technical University

Outline

- Direct Optimization at Higher Fidelity
  - Medium Fidelity Analysis Tools
  - The FE Based Vortex Dynamics
  - Optimization Results
- Multi-fidelity Design Optimization
  - The AMMF Algorithm
  - Structural Design Case Study
- Closing Statements

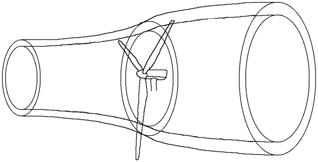
Direct Optimization with Higher Fidelity Analysis

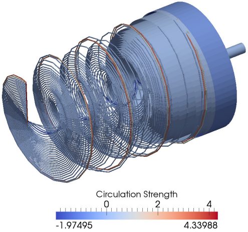
Multidisciplinary Design Optimization of Wind Turbines

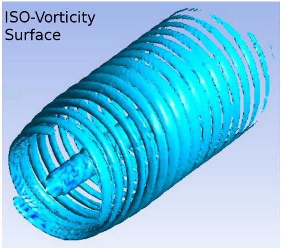
- Trends show wind turbines are getting larger
  - Higher turbines better winds
  - Improved economies of scale (*e.g.* offshore)
- Future growth will require advanced designs
  - Bend-twist coupling, curved blades, active load alleviation, winglets, coning, etc.
- Multidisciplinary Design Optimization (MDO)
  - Simultaneously optimize multiple disciplines (*e.g.* aero, structural, control, *etc.*)
  - Optimization based on holistic metrics (*e.g.* cost of electricity)
  - Wind turbine design constrained by unsteady loads (*i.e.* strong gusts and fatigue)

Medium Fidelity Analysis Tools

Analysis Tools







- Conventional preliminary design tools
  - Blade Element Momentum Theory and Linear beam theory
  - Fast and efficient, but lacks the fidelity required by advanced designs
- High fidelity analysis
  - Grid-based CFD and Shell and Brick based FEM
  - Excellent fidelity, very expensive for optimization
- Need medium fidelity analysis (improved fidelity, still efficient)
  - Vortex Dynamics (VD)
  - Nonlinear beam theory (GEBT)
  - Anisotropic Cross Section Analysis (VABS)

Aero-elastic Optimization with Conventional VD

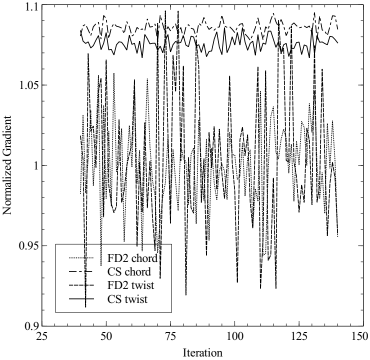


Figure from Lawton and Crawford 2015

- Aeroelastic model with Conventional VD, GEBT and VABS
- Obtained optimization results with
  - Pure aerodynamic
  - Aero-elastic with fixed wake
- Failed to obtain aeroelastic results with free wake simulations
  - Pure vortex methods are fundamentally chaotic
  - Numerical noise spoils the gradients and optimization
- **Conventional VD not suitable for aero-elastic optimization**

Michael K. McWilliam, Stephen Lawton, and Curran Crawford. “Towards a framework for aero-elastic multidisciplinary design optimization of horizontal axis wind turbines” In AIAA Annual Sciences Meeting, 2013

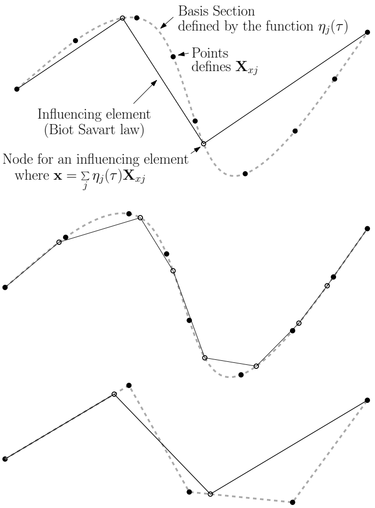
The Finite Element Based Vortex Dynamics

FEM Parameterization of the Wake

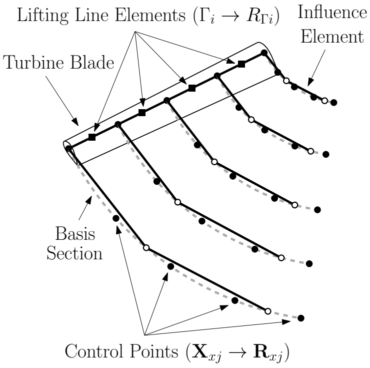
- Vortex position in the wake defined by interpolating splines:

$$\boldsymbol{x} = \sum_j \eta_j(\tau)\boldsymbol{X}_{xj} \quad \dot{\boldsymbol{x}} = \sum_j \dot{\eta}_j(\tau)\boldsymbol{X}_{xj}$$

- Can have an arbitrary number of influence elements and control points
    - Can add more influence elements to improve accuracy
    - Can remove control points to accelerate calculations



FEM Solution Algorithm



- Convergence defined by a residual:

$$\boldsymbol{r}_x \equiv \dot{\boldsymbol{x}} + \boldsymbol{\Omega} \times (\boldsymbol{x} - \boldsymbol{x}_0) - \boldsymbol{u}_\infty - \boldsymbol{u}_\gamma$$

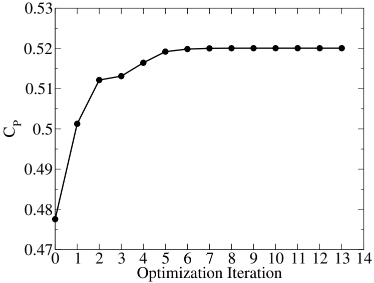
- Mapped to control points through Galerkin projection:

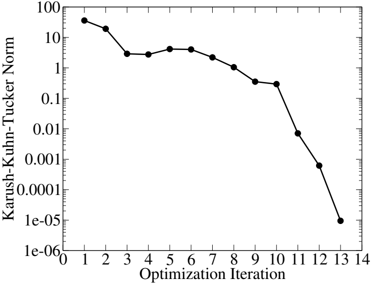
$$\boldsymbol{R}_{xj} = \int_{\tau_0}^{\tau_f} \zeta_j(\tau) \boldsymbol{r}_x(\tau) d\tau$$

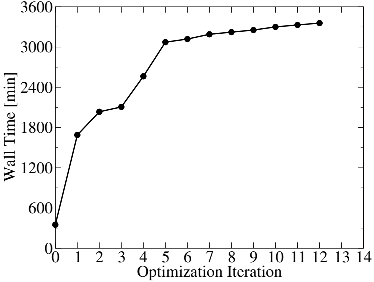
- Solved with a Newton iteration
  - Adaptive relaxation required to get reliable convergence
  - **See Video for example**
- Best results with a far-wake model
  - Avoids singularities
  - Eliminates wake-truncation errors

Optimization Results

Optimization Convergence with FEM-Based VD



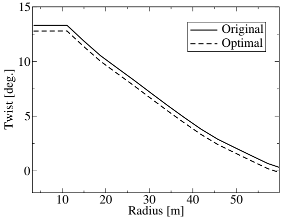


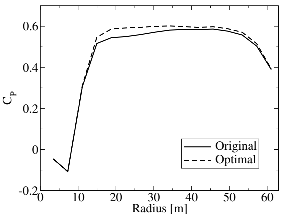


- Used analytic gradients
  - Explicit VD residual definition predicts changes in state
- Tight optimization tolerances
- Small changes avoid singularities

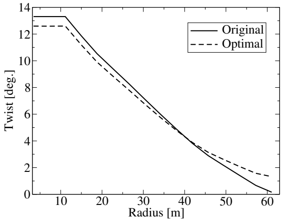
Optimization with FEM-Based VD

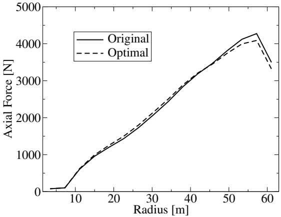
Aerodynamic Only Optimization:





Aero-elastic Optimization:





- Aeroelastic optimization created more efficient designs

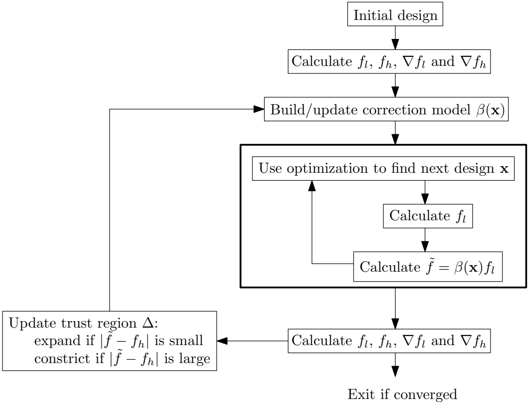
Multi-fidelity Design Optimization

The Multi-Fidelity concept

- Uses both a high fidelity and low fidelity model
  - Less expensive by using fewer high fidelity results
  - Reduces surrogate error with low-fidelity results
- Fidelity could be based on:
  - Formulation (*e.g.* RANS vs. BEM)
  - Grid resolution (*e.g.* fine vs. course)
  - Type of simulation (*e.g.* unsteady vs. steady)
  - *etc.*
- Low fidelity just needs to show similar trends

The AMMF Algorithm

The AMMF Algorithm



- High fidelity used for accuracy
- Low fidelity is used for speed
- Correction for first order consistency

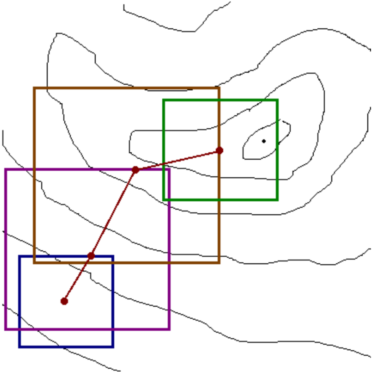
$$\tilde{f}(\boldsymbol{x}) = f_l(\boldsymbol{x}) + \beta(\boldsymbol{x})$$

$$\beta(\boldsymbol{x}) = f_{h0} - f_{l0} + (\nabla f_{h0} - \nabla f_{l0})\, \Delta \boldsymbol{x}$$

- Trust-region for robustness

The Trust Region Algorithm

- The trust-region defines the region where we can “trust” our approximation
- Constrained to stay within the trust-region
- Re-centered at every major iteration
  - Only when an improved is found
- Trust region is resized
  - If the approximation gives excellent agreement then it grows
  - If the trust region gives poor agreement then it shrinks
    - If the inner optimization fails to find an improvement, it will repeat within the smaller trust region
    - Similar to the line search algorithm
  - Otherwise maintain the trust region



Constraints in the AMMF Algorithm

- Constraints are corrected in the same way
- The constraints are present in the low fidelity optimization
- Constraints receive special treatment in Approximation and Model Management Framework (AMMF)
- First an estimated Lagrangian is calculated

$$\Phi = f + \tilde{\lambda}_e \cdot |\boldsymbol{c}| + \tilde{\lambda}_i \cdot \max(0, -\boldsymbol{c}_i)$$

  - $\tilde{\lambda}$ are the Lagrange multipliers estimated from previous iterates.
  - $\tilde{\lambda}$ is specified for the first iteration

- New iterate only accepted when $\Phi_i < \Phi_{i-1}$
- Trust region is expanded or contracted based on M:

$$M = \frac{\Phi_{i-1} - \Phi_i}{\Phi_{i-1} - \tilde{\Phi}_i}$$

  - Trust region expanded if M is close to 1
  - Trust region contracts if M is far from 1

Multi-fidelity Structural Design Optimization

Summary of Low Fidelity Tools

Position	EA	EIx	EIy	GJ
0.05	0.0	2.6	-4.9	-5.4
0.15	0.5	1.1	-3.0	-0.8
0.25	-0.4	-1.8	2.1	-1.4
0.35	-0.7	-2.6	1.7	-3.1
0.45	-0.7	-3.1	1.0	-5.5
0.55	-0.9	-3.1	-0.3	-7.7
0.65	-0.8	-2.9	-1.7	-9.3
0.75	-0.6	-2.2	-2.2	-9.2
0.85	-0.6	-1.7	-3.5	-5.9
0.95	-0.1	-1.2	-2.0	-2.0

Table: Percent Error with BECAS

- Low fidelity cross section tool
  - Thin-walled cross section assumption
  - Rigid cross section (Euler-Bernoulli)
  - Classic laminate theory
  - Written in C++
  - Python bindings with Swig
  - Will have analytic gradients
  - Within 10% compared to BECAS
- High fidelity cross section tool
  - Based on BECAS
  - BECAS uses an FE formulation
  - Solves the warping field
  - Gives fully populate matrix

Summary of Low Fidelity Tools

DTU

Operation	Calculation time [s]
Linear Beam Model	0.0035
LF cross section model	0.0074
BECAS	200.1866

Table: Speed Comparison of Low Fidelity Tools

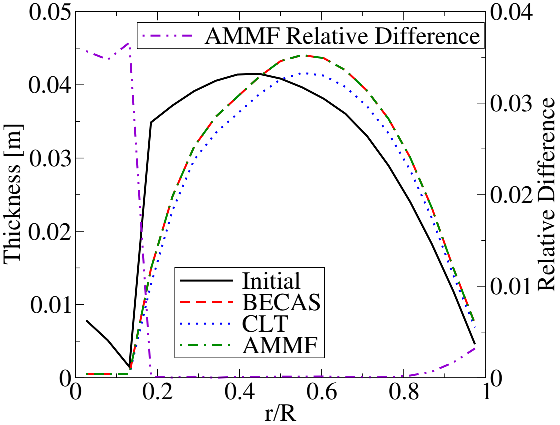
- Linear Beam Model
  - C++ code from my PhD
  - Analytic gradients wrt.
    - Positions
    - Orientation
    - Cross section properties
    - Applied forces
  - Solves equivalent forces for given deflection
- Speed comparison:
  - With python bindings
  - Calculation for whole blade
  - 19 elements
  - DTU $10MW$

Problem Description

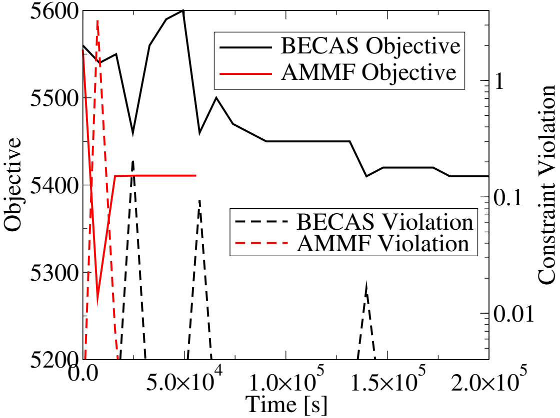
- Minimize DTU 10MW Blade Mass
- Varying spar cap thickness
- Subject to:
    - Tip deflection constraint
- Analysis based on the equivalent static problem (*i.e.* Frozen loads)
- Compared pure BECAS, pure CLT and AMMF
- Looked at various AMMF configurations:
    - Additive vs. Multiplicative corrections
    - Trust region size
    - Initial Lagrange multiplier (*i.e.* Penalty parameter)

Optimization Results

- Low fidelity model is not conservative
    - Will produce infeasible solutions
- AMMF reproduced the BECAS solution
    - AMMF had better constraint resolution
- AMMF gives accurate corrections
- Additive vs multiplicative corrections:
    - Gives similar solutions
    - Similar performance



Optimization Convergence

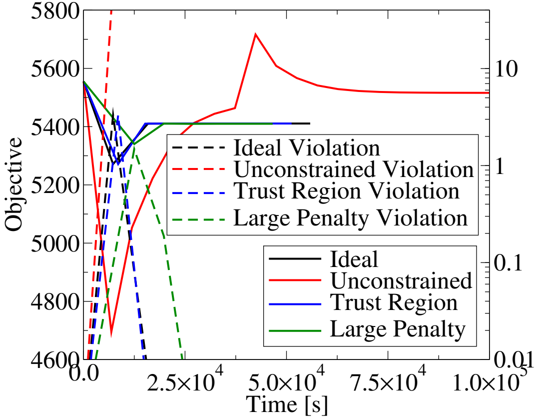


- AMMF converges 12 times faster
  - Just 2 major iterations
- AMMF had smoother convergence
  - Only 1 iteration with constraint violation
  - BECAS optimization ended due to maximum iterations
- Low fidelity models more suitable for optimization

AMMF Robustness

AMMF guards against poor approximations

- Unconstrained has all protections disabled
  - Large violations
  - Fails to converge
- Trust region is most robust
  - Same progress as ideal configuration
- Large penalties work without trust region
  - No large violations
  - More searching



Closing Statements

Conclusions

- Higher fidelity in direct optimization is challenging but possible
  - Underlying tools may be non-smooth
  - Tools may need to be re-written or re-formulated (optimization proof)
  - Developed a totally new formulation for vortex methods based on FEM
  - Successfully obtained aero-elastic optimization results with vortex methods
- Higher fidelity through multi-fidelity design optimization is promising
  - Effective when low fidelity gives similar trends much faster
  - Achieved a 12 times speed up using multi-fidelity techniques
  - The AMMF algorithm is robust in handling errors
  - Ongoing case studies focusing on difficult problems

Acknowledgments

This work was supported by

- The University of Victoria
- Natural Sciences and Engineering Research Council of Canada
- The Technical University of Denmark
- The Danish Energy Technology Development and Demonstration Program

Thank-you for your interest

Comments or Questions?