NREL Transforming ENERGY

WETO Software Stack User Workshops
Systems Engineering

June 10, 2024

Rafael Mudafort
Pietro Bortolotti
Garrett Barter
Dan Zalkind
Ernesto Camarena
Jeff Allen
Ethan Young

Agenda

Section	Duration	Time	Speaker
Intro	5’	0:00 - 0:05	Rafael Mudafort
WETO Stack Overview	10’	0:05 - 0:15	Rafael Mudafort
WETO Stack discussion	**15’**	**0:15 - 0:30**	**YOU**
WISDEM	10’	0:30 - 0:40	Pietro Bortolotti
WEIS	10’	0:40 - 0:50	Dan Zalkind
pyNuMAD	10’	0:50 - 1:00	Ernesto Camarena
WindSE	10’	1:00 - 1:10	Jeff Allen and Ethan Young
Polls / open-ended questions	**2’**	**1:10**	**YOU**
Community discussion	**30’ - 40’**	**1:15 - 1:50**	**YOU**
Wrap up	5’	1:50 - end	Garrett Barter

Holistic Modeling Project

WETO Software Portfolio Coordination

US DOE & Lab-based Wind Research Projects

WETO invests in wind energy **software** that enables and accelerates the innovations needed to advance wind energy.

NREL's active WETO projects

Floating Downwind Turbines: A Conceptual System-Level Design and Feasibility Study for U.S. Waters
Wind Grid Integration Stakeholder Engagement
North American Energy Resiliency Model (NAERM)
Atmosphere to Electrons (A2e) Performance Risk, Uncertainty and Finance (PRUF) Analysis Support
Automating In-Situ Grinding and Repair for Thermoplastic Blades
Codesign and Intelligent Approaches for Cost-Effective Operation and Maintenance of Generators and Power Converters
Evaluating Deterrent Stimuli for Increasing Species-Specific Effectiveness of an Advanced Ultrasonic Acoustic Deterrent
High-Fidelity Modeling Toolkit for Wind Farm Development
Eagle Topic Area 3 Funding Opportunity Announcement (FOA) Support
Wind Turbine Drivetrain Reliability Assessment and Remaining Useful Life Prediction (TCF)
Enabling Autonomous Wind Plants through Consensus Control (TCF)
Study on the Potential Application of Additive Manufacturing in Wind Turbine Components and Tooling
Fusion Joining of Thermoplastic Composites Using Energy Efficient Processes (TCF)
Atmosphere to Electrons to Grid (A2e2g)
Wind Standards Development
North American Renewable Integration Study
High-Fidelity Modeling
Co-Simulation Study and Control of a Wind Farm for Conversion Services
Working Together to Resolve Environmental Effects of Wind Energy (WREN)
Wind Power as Virtual Synchronous Generation (WindVSG)
Big Adaptive Rotor
Continental-Scale Transmission Modeling Methods for Grid Integration Analysis
Multiscale Integration of Control Systems (EMS/DMS/BMS)
Modeling and Validation for Offshore Wind
Energy Sector Modeling and Impacts Analysis
Advanced Modeling, Dynamic Stability Analysis, and Mitigation of Control Interactions in Wind Power Plants
Technology Development and Innovation to Address Operational Challenges
Enabling Larger Rotors Through Modular, Customizable, Inflatable Blades

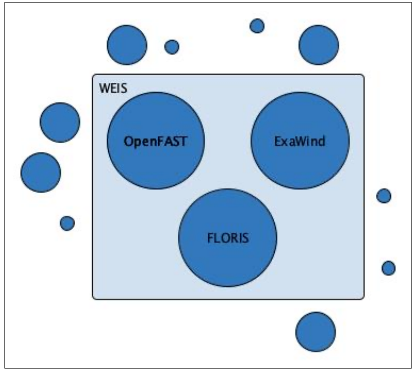
Argonne National Laboratory | Berkeley Lab Lawrence Berkeley National Laboratory | Lawrence Livermore National Laboratory | NREL National Renewable Energy Laboratory | Pacific Northwest National Laboratory | Sandia National Laboratories

https://www.energy.gov/eere/wind/wind-energy-technologies-office-projects-map

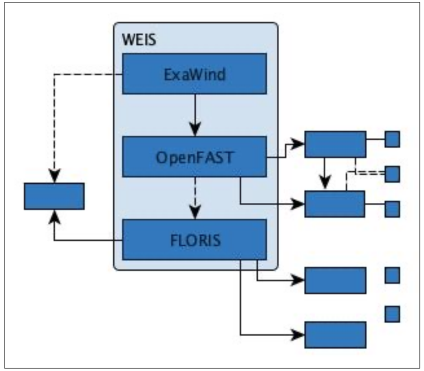
Holistic Modeling Project

Objective

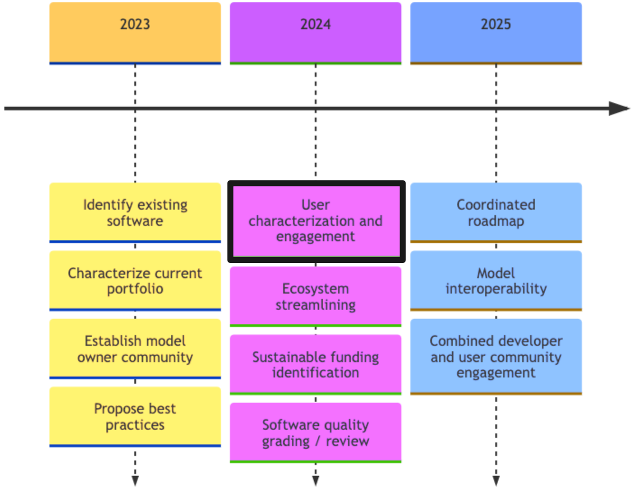
Past: Loose collection of software



Future: Cohesive software stack



Project Timeline

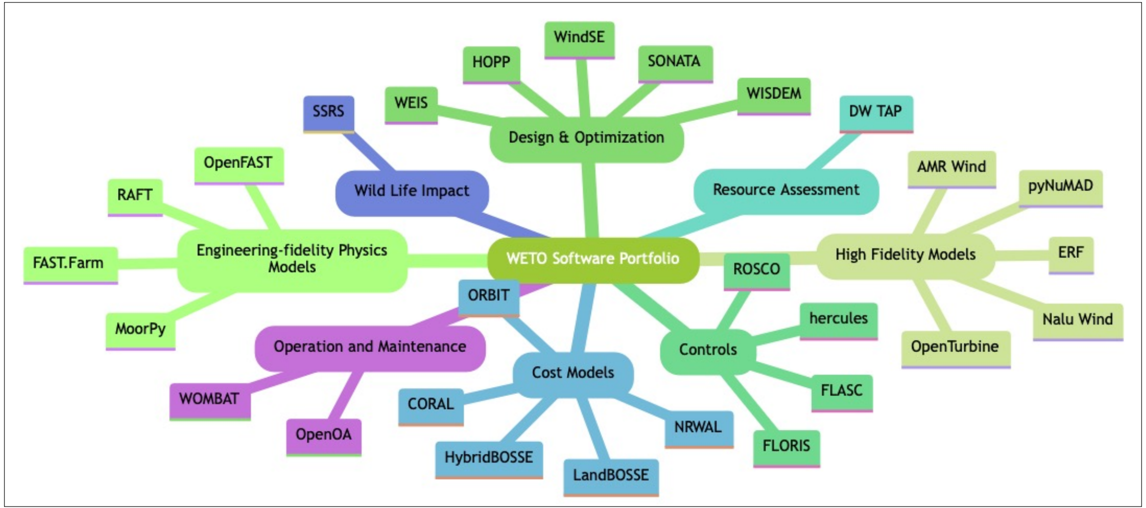


WETO Software Stack

Overview

WETO Software Stack

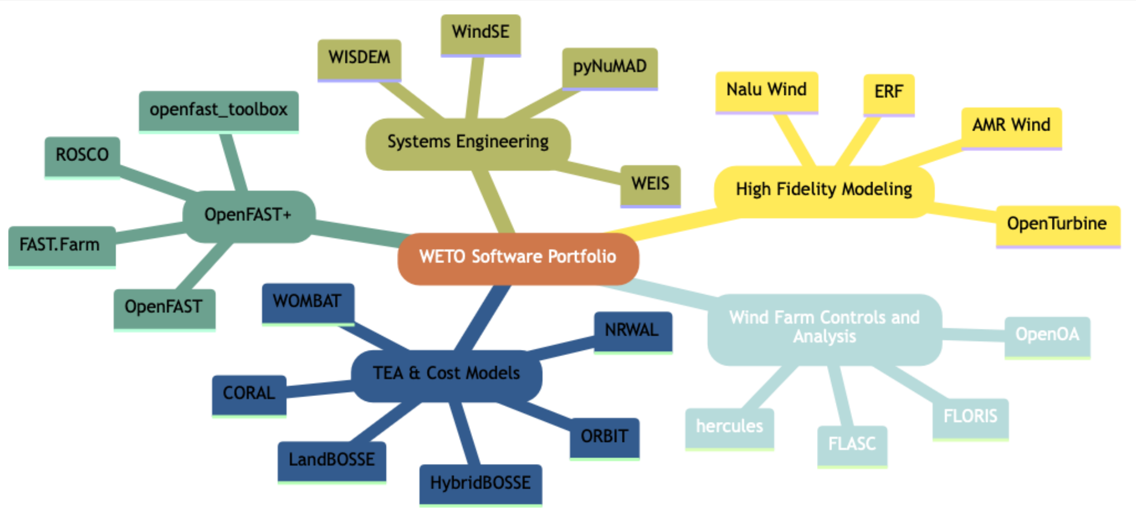
Grouped by what it does



https://nrel.github.io/WETOStack/portfolio_analysis/software_list.html

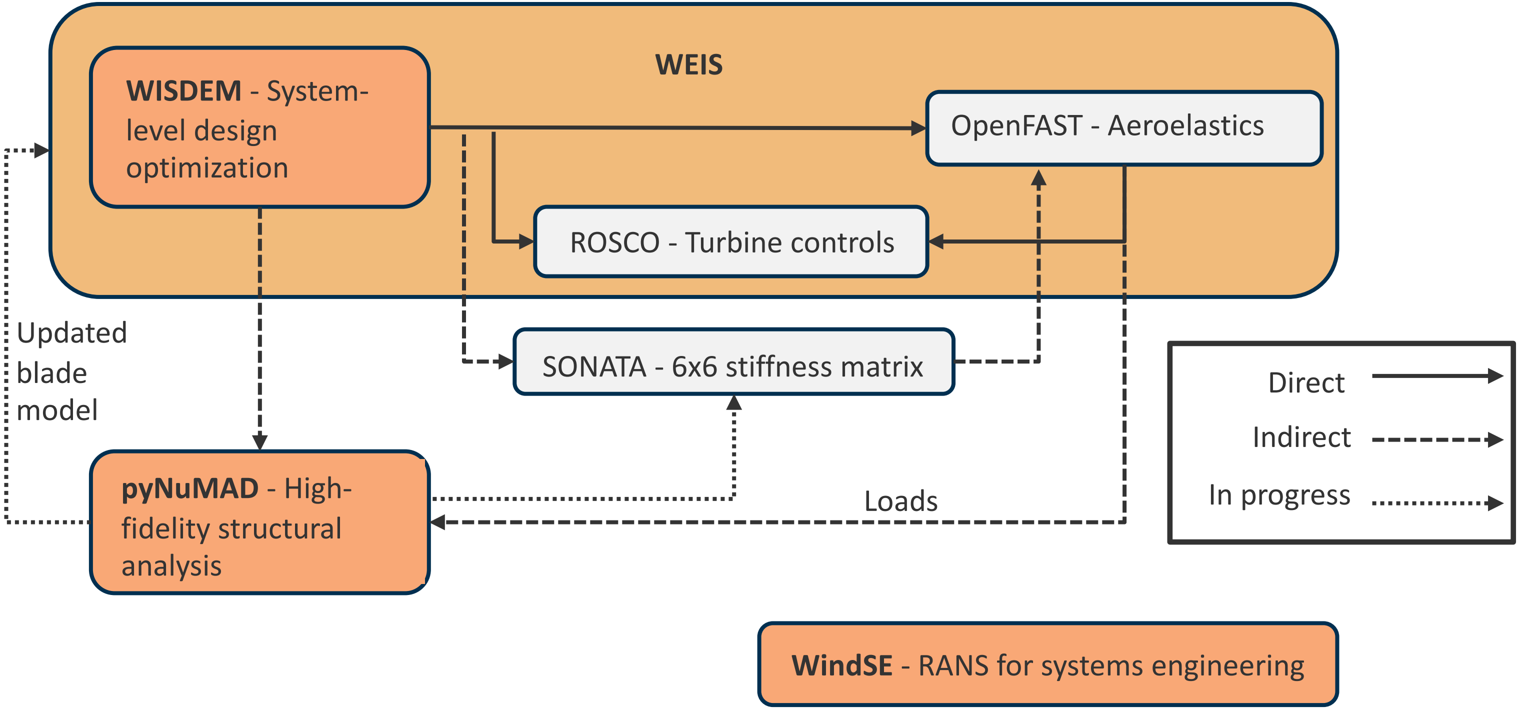
WETO Software Stack

Grouped by how its used



Systems Engineering

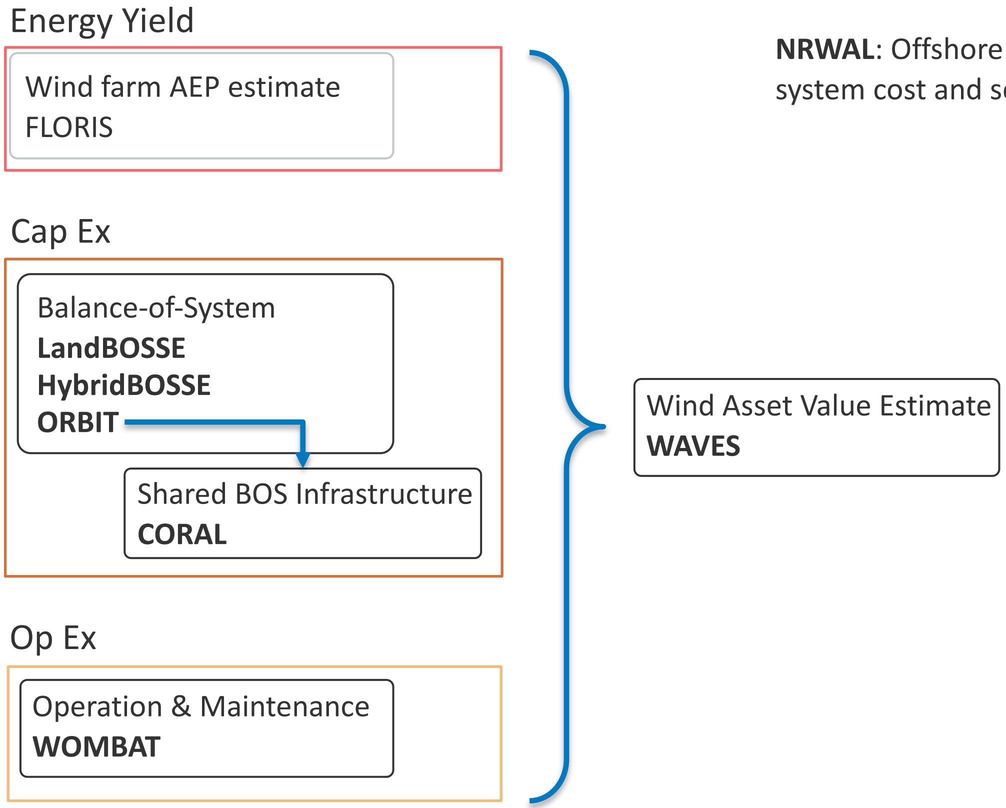
Workshop: June 10



Adapted from Big Adaptive Rotor (BAR) project

Technoeconomic Analysis / Cost Modeling

Workshop: June 12

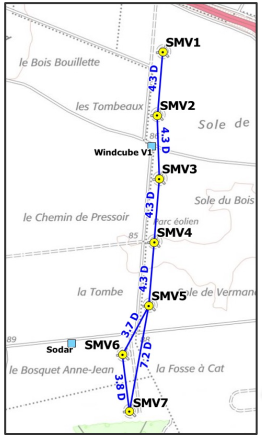


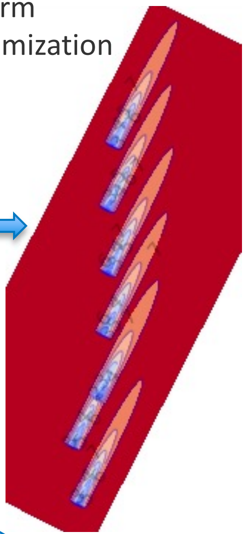
NRWAL: Offshore wind system cost and scaling model

Wind Farm Controls and Analysis

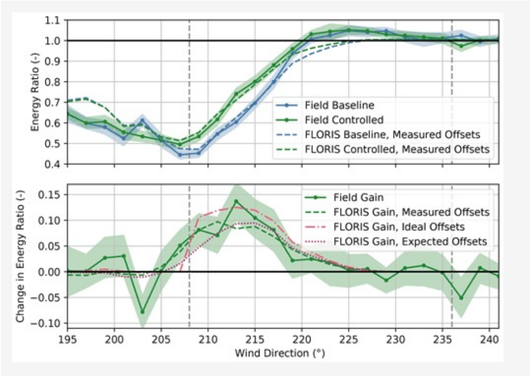
Workshop: June 18

FLORIS: Steady-state modeling, farm controls optimization

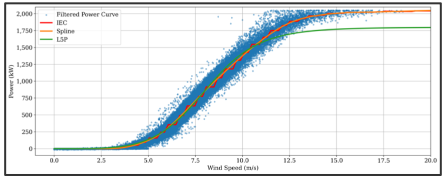




FLASC: Validate FLORIS model with SCADA, compare control methods



Hercules: Realtime high-fidelity simulator for hybrid power plants with a specific focus on wind farm controls.

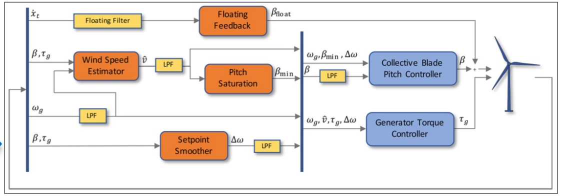


OpenOA: Characterize plant performance and quantify sources of operational loss

OpenFAST+

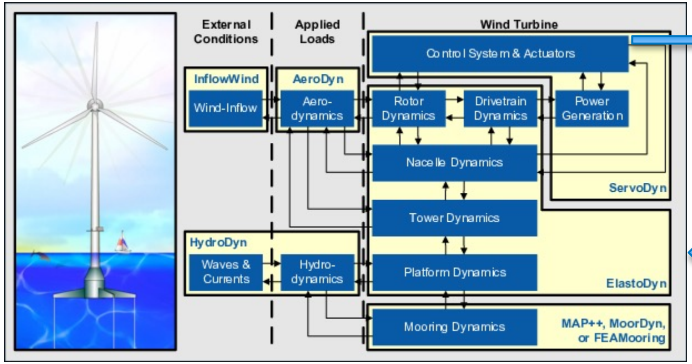
Workshop: June 20

ROSCO



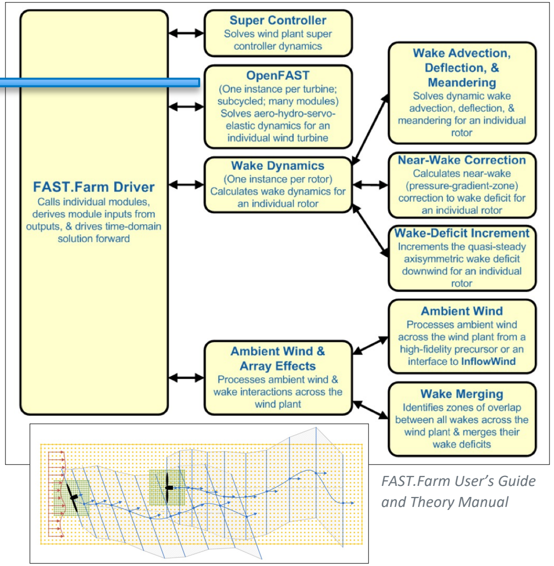
N. J. Abbas et al.: A reference controller for wind turbines

OpenFAST



OpenFAST v3.5.3 documentation

FAST.Farm

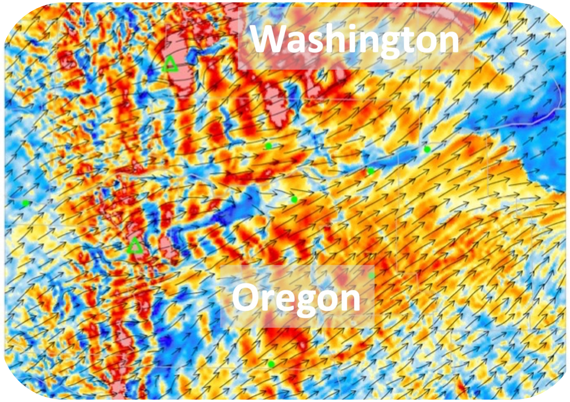


FAST.Farm User's Guide and Theory Manual

`openfast_toolbox`

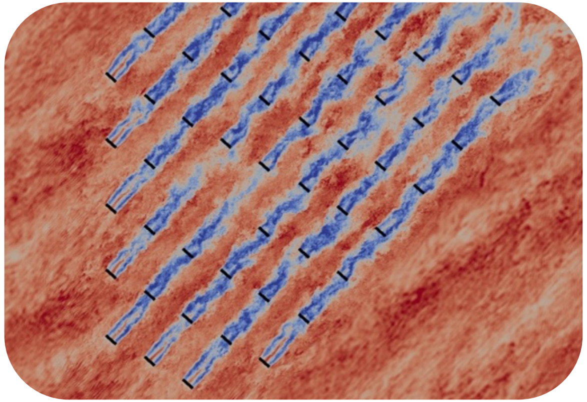
High Fidelity Models

Workshop: TBD



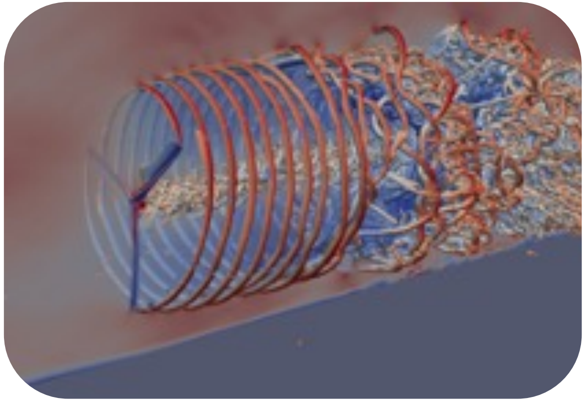
Mesoscale: ERF

- Regional scale weather
- **Scales 10 km to 1000 km**
- WRF numerics & models, built on AMReX
- GPU compatible
- Compressible



Microscale: AMR-Wind

- Atmospheric boundary layer
- **Scales less than 10 km**
- Large Eddy Simulation built on AMReX
- GPU compatible
- Structured grid with refinement zones
- Incompressible



Turbine scale: NALU-Wind

- Turbine, rotor, tower, nacelle
- **Scales less than 1 km**
- Unsteady Reynolds Averaged Navier Stokes
- GPU compatible
- Unstructured grid, geometry resolving
- Incompressible

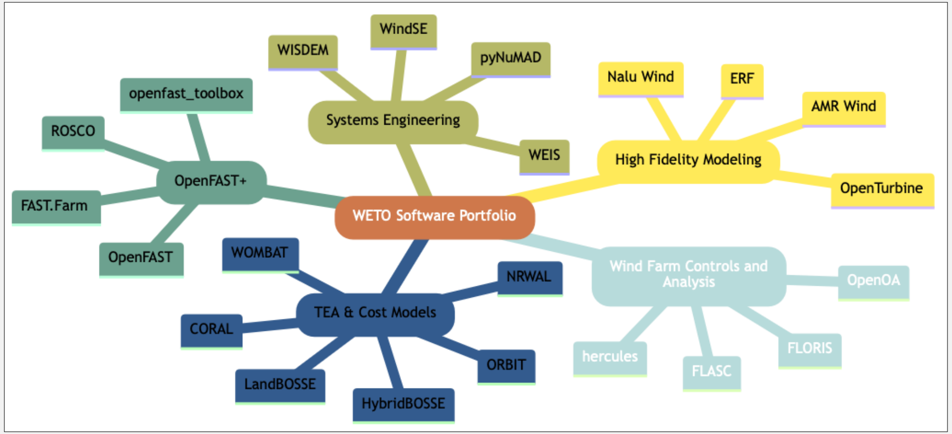
WETO Software Stack

Open Discussion

WETO Software Stack

Raise your “hand” and we’ll call your name to ask your question.

- **Discussion topics**
  - Prospective / new users:
    - What is your experience in the learning or onboarding process?
  - Experienced users:
    - What have been your primary pain points or bottlenecks?
    - What has or has not worked in integrating WETO software into your workflows?
    - How thoroughly do you feel you understand the capability of the tools available in the WETO Software Stack?
    - What has helped or hindered your open-source contribution to the WETO Software Stack?

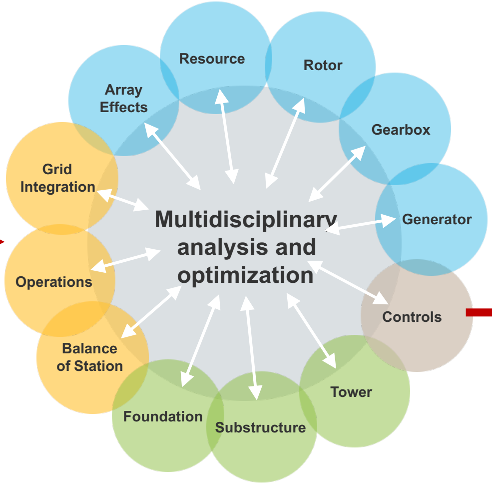


WISDEM

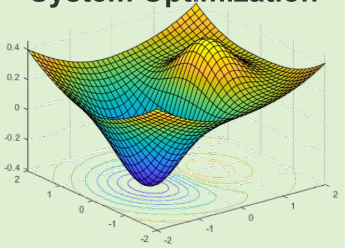
Pietro Bortolotti

WISDEM: Bringing innovations into the wind turbine-farm system

Innovations



System Optimization



Sensitivity Analysis

- **Conceptual design**
- **LCOE rollup**
- **Cost-benefit tradeoffs**

Questions

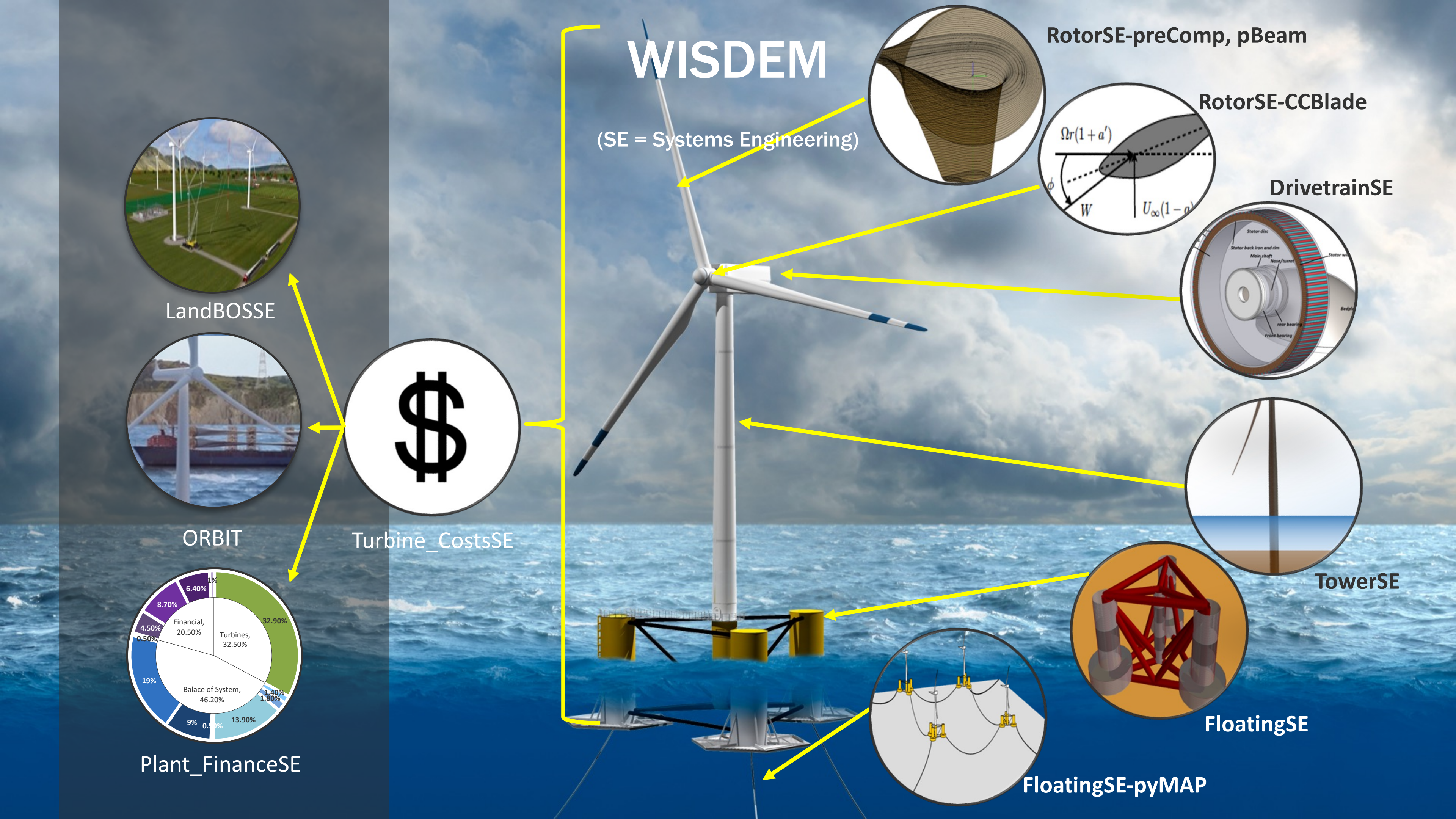
- What are the system-level cost-benefit tradeoffs of an innovation?
- Design a turbine or plant with specific properties
- Develop new design methodologies for wind turbines & plants?

Approach

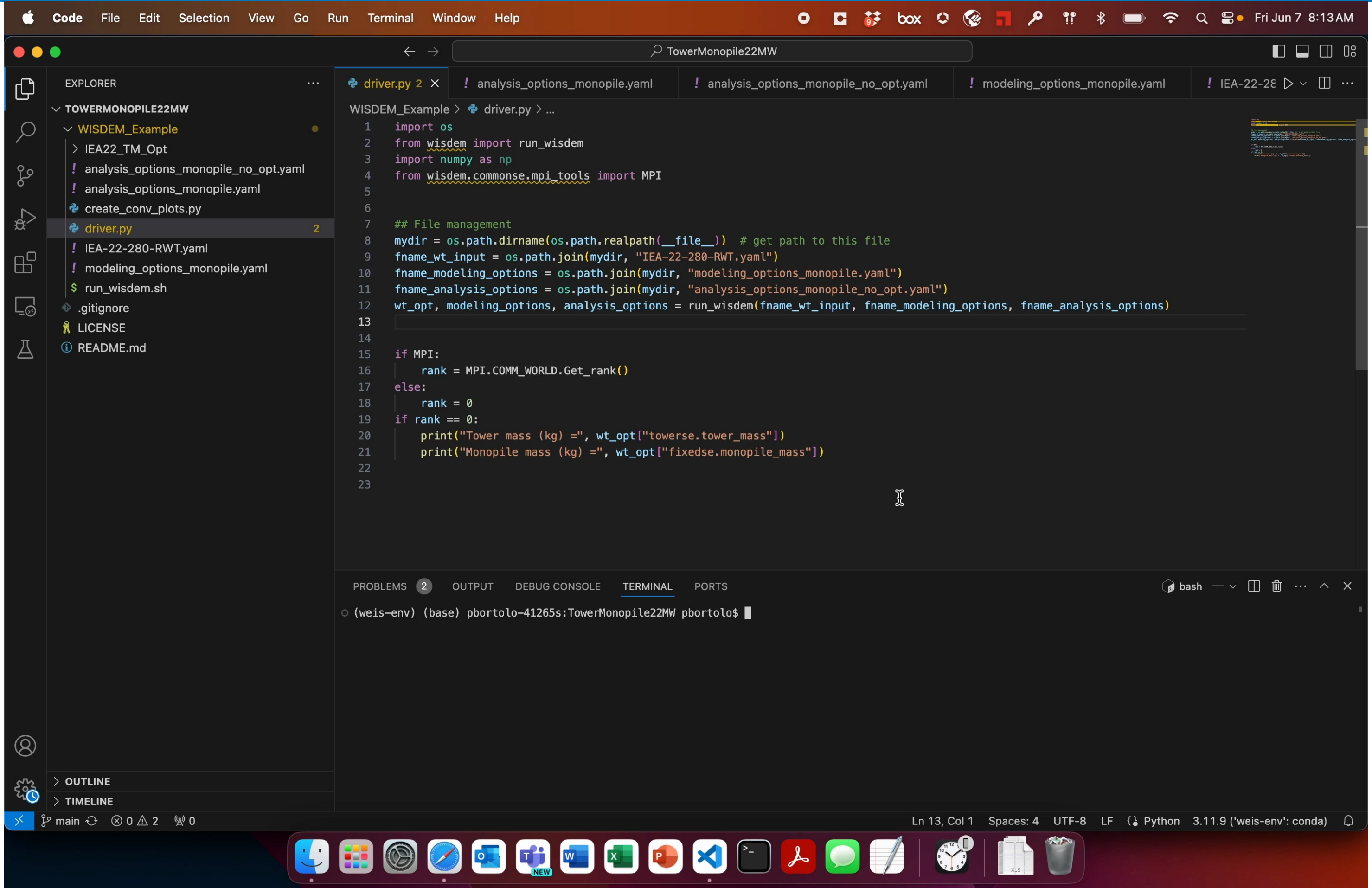
- Integrate multidisciplinary engineering and cost models in open frameworks
- Apply and develop advanced design and optimization methods
- Promote collaborative research among laboratories, industry, and academia
- Identify pathways to performance improvement and cost reductions by capturing system interactions

Impact

- Designed IEA Wind reference turbines
- Evaluate LCOE cost-benefit tradeoffs and sensitivities of new technology and/or manufacturing or logistic innovations


WISDEM
(SE = Systems Engineering)
RotorSE-preComp, pBeam
RotorSE-CCBlade
DrivetrainSE
TowerSE
FloatingSE
FloatingSE-pyMAP
LandBOSSE
ORBIT
Turbine_CostsSE
Plant_FinanceSE
Financial, 20.50%
Turbines, 32.50%
Balace of System, 46.20%
32.90%
6.40%
8.70%
4.50%
0.50%
19%
9%
13.90%
1.40%
1.80%
1%

WISDEM Demo


TowerMonopile22MW
EXPLORER
TOWERMONOPILE22MW
WISDEM_Example
IEA22_TM_Opt
analysis_options_monopile_no_opt.yaml
analysis_options_monopile.yaml
create_conv_plots.py
driver.py
IEA-22-280-RWT.yaml
modeling_options_monopile.yaml
run_wisdem.sh
.gitignore
LICENSE
README.md
OUTLINE
TIMELINE
WISDEM_Example > driver.py > ...
import os
from wisdem import run_wisdem
import numpy as np
from wisdem.commonse.mpi_tools import MPI
File management
mydir = os.path.dirname(os.path.realpath(__file__)) # get path to this file
fname_wt_input = os.path.join(mydir, "IEA-22-280-RWT.yaml")
fname_modeling_options = os.path.join(mydir, "modeling_options_monopile.yaml")
fname_analysis_options = os.path.join(mydir, "analysis_options_monopile_no_opt.yaml")
wt_opt, modeling_options, analysis_options = run_wisdem(fname_wt_input, fname_modeling_options, fname_analysis_options)
if MPI:
rank = MPI.COMM_WORLD.Get_rank()
else:
rank = 0
if rank == 0:
print("Tower mass (kg) =", wt_opt["towerse.tower_mass"])
print("Monopile mass (kg) =", wt_opt["fixedse.monopile_mass"])
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
(weis-env) (base) pbortolo-41265s:TowerMonopile22MW pbortolo$
Ln 13, Col 1
Spaces: 4
UTF-8
LF
Python
3.11.9 ('weis-env': conda)

WISDEM Roadmap Across Five Areas

USABILITY: Goal is for WISDEM to be a professionally written, documented, tested, and maintained software that is accessible to wind industry and research engineers

- Improvements made in documentation and installation, but documentation could be better and GUI capability enhanced
- More plug-and-play modularity for user-based calculations of mass, stiffness, and cost parameters

FIDELITY: Goal is for WISDEM to be a low-fidelity conceptual design tool for land-based and offshore wind turbines/plants

- Empirical extrapolation to unsteady & fatigue loads (with ML or other means) as otherwise WISDEM is steady-state only
- 6x6 cross-sectional tool (ANBA4 or similar) to replace precomp (mostly in support of OpenFAST-WEIS)
- Iterative solve for rotor aeroelasticity
- Drivetrain still has a lot of simple sizing instead of optimization-driven design

COST MODELS: Goal is to be able to represent cost-benefit tradeoff studies of new technologies and logistical processes

- Blades, towers, monopiles use “bottom-up” cost models, but other components instead use cost regressions that do not adapt to new technologies, new scales, or new manufacturing
- This has been done for the balance of station models, but not the turbine components themselves or O&M costs

COMPONENTS: Goal is to be able to represent the performance and system impact of new technologies

- Add an electrolyzer module for wind-hydrogen hybrid systems

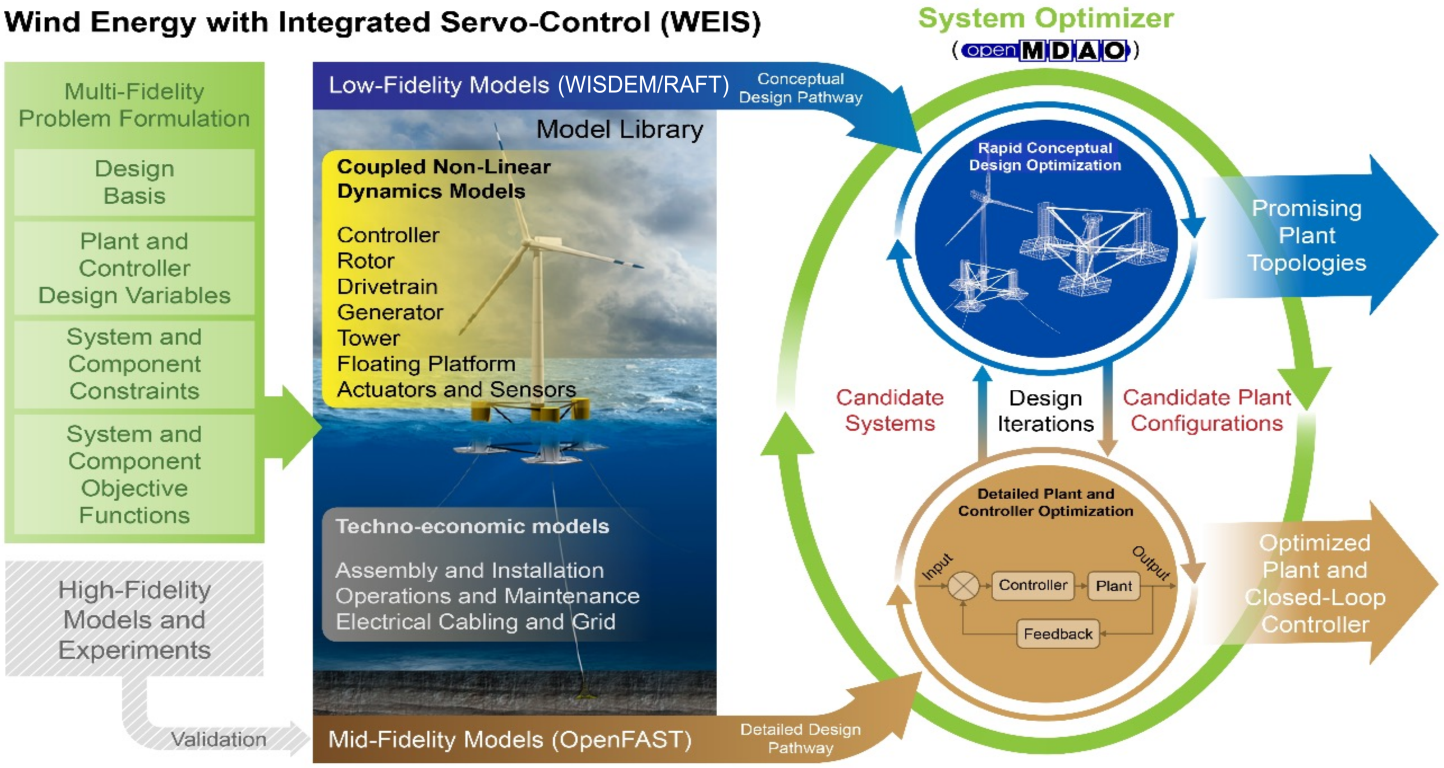
NUMERICS: Goal is to be able to apply provide broad design optimization tools and advanced optimization and uncertainty quantification techniques

- Better optimization, gradient based support, UQ techniques, etc

WEIS

Dan Zalkind

Control Co-Design using WEIS



https://github.com/WISDEM/WEIS

WEIS Stack

Structural postprocessing	**WISDEM***
Loads postprocessing	**pCrunch**
Aero-servo-hydro-elastic modeling	**OpenFAST*** / **RAFT***
Hydrodynamic preprocessing	**pyHAMS***
Controller tuning	**ROSCO* + ROSCO_Toolbox***
Other preprocessing	(many NREL utilities)
Geometry, mass, and cost	**WISDEM***
Parameterized turbine + floater	**WindIO***

Level 3 | Level 2 | Level 1

WEIS (glue code)*

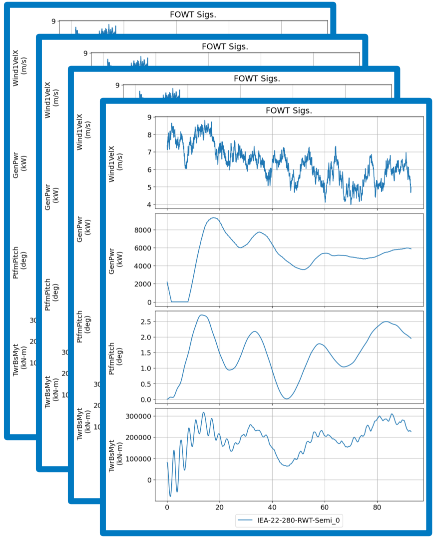
Co-Design Optimization*

WEIS Capabilities

- Each module of WEIS can be used individually for analysis and model generation
  - WISDEM
  - RAFT
  - OpenFAST
  - ROSCO
  - pCrunch
- Generate OpenFAST models programmatically from WISDEM geometry
  - Tune and optimize a ROSCO controller automatically
- Optimization or parameter study of any available design variable
  - IEA-22MW RWT floating system and controller
- Automated running of OpenFAST cases and some IEC DLCs
- Create open aeroelastic models of platforms and turbines given limited input information

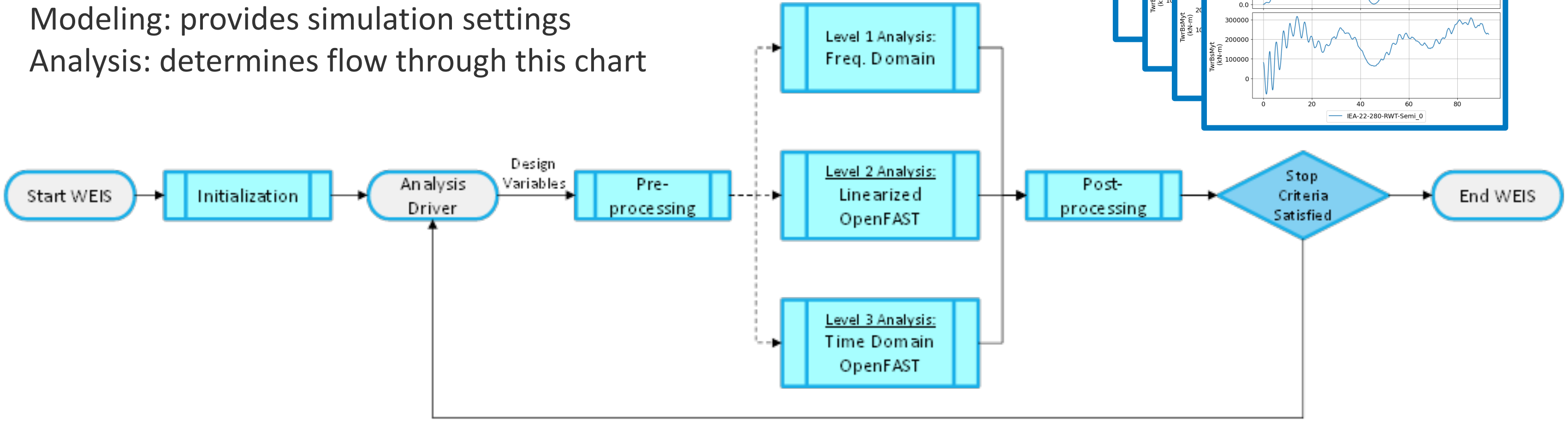
- Tool can be adapted to a wide variety of solutions with a reasonable effort

WEIS Data Flow



WEIS Inputs:

- Geometry: specifies FOWT model
- Modeling: provides simulation settings
- Analysis: determines flow through this chart

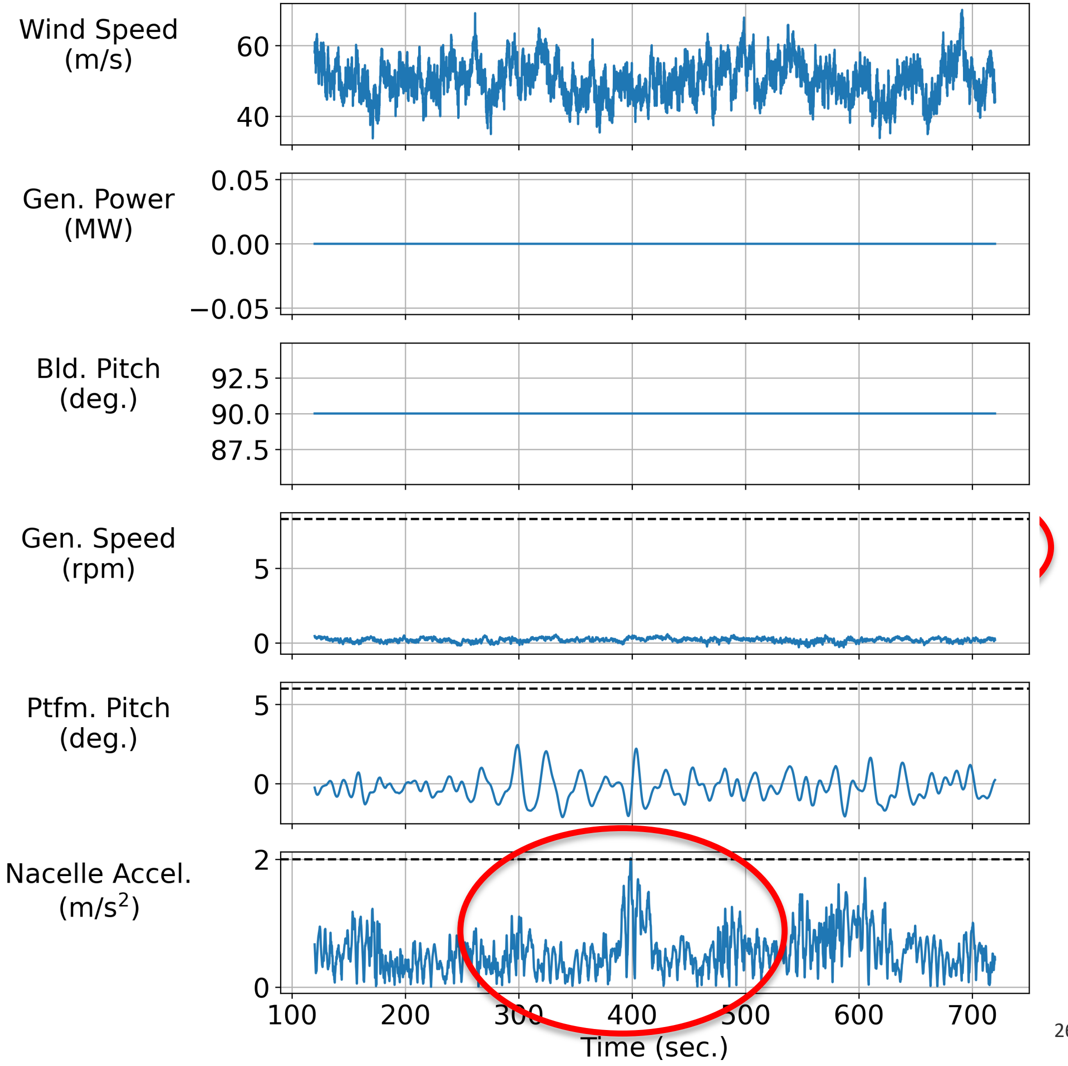


Aggregate & System Engineering Outputs

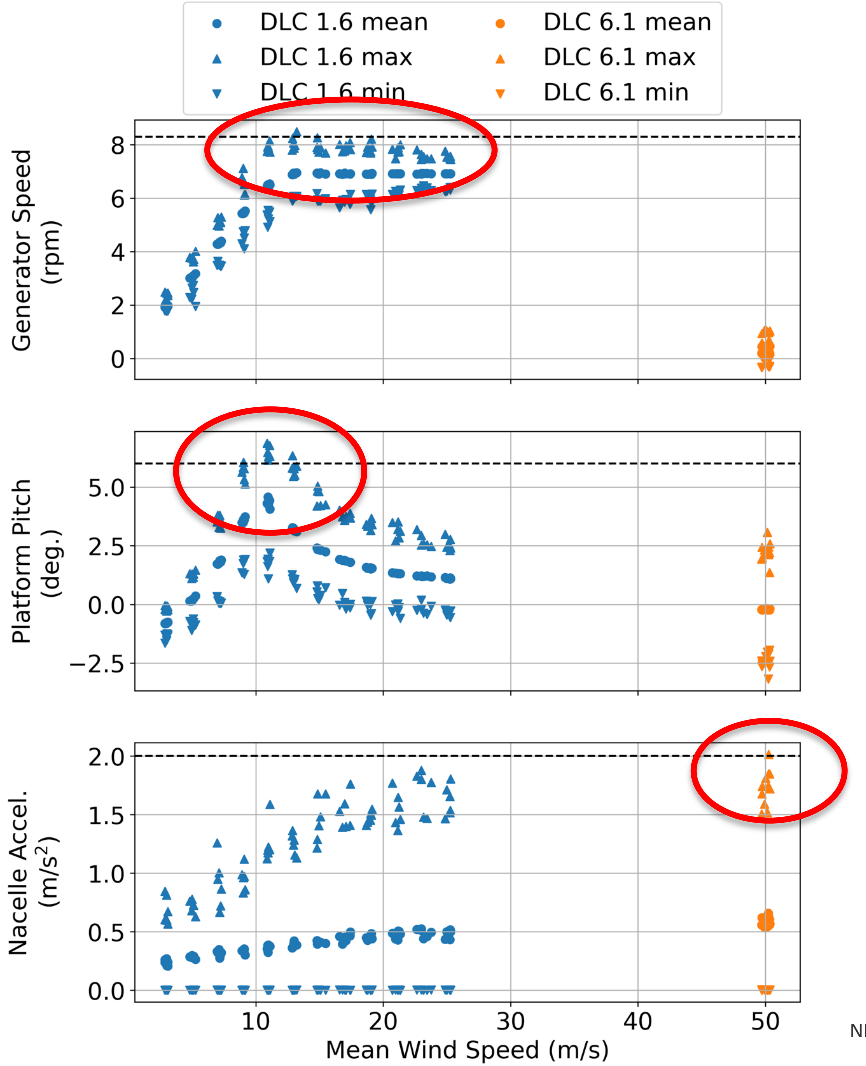
- Platform/hull mass and cost
- Natural periods
- Power and AEP
- Loads and stresses

Demo: Individual simulations

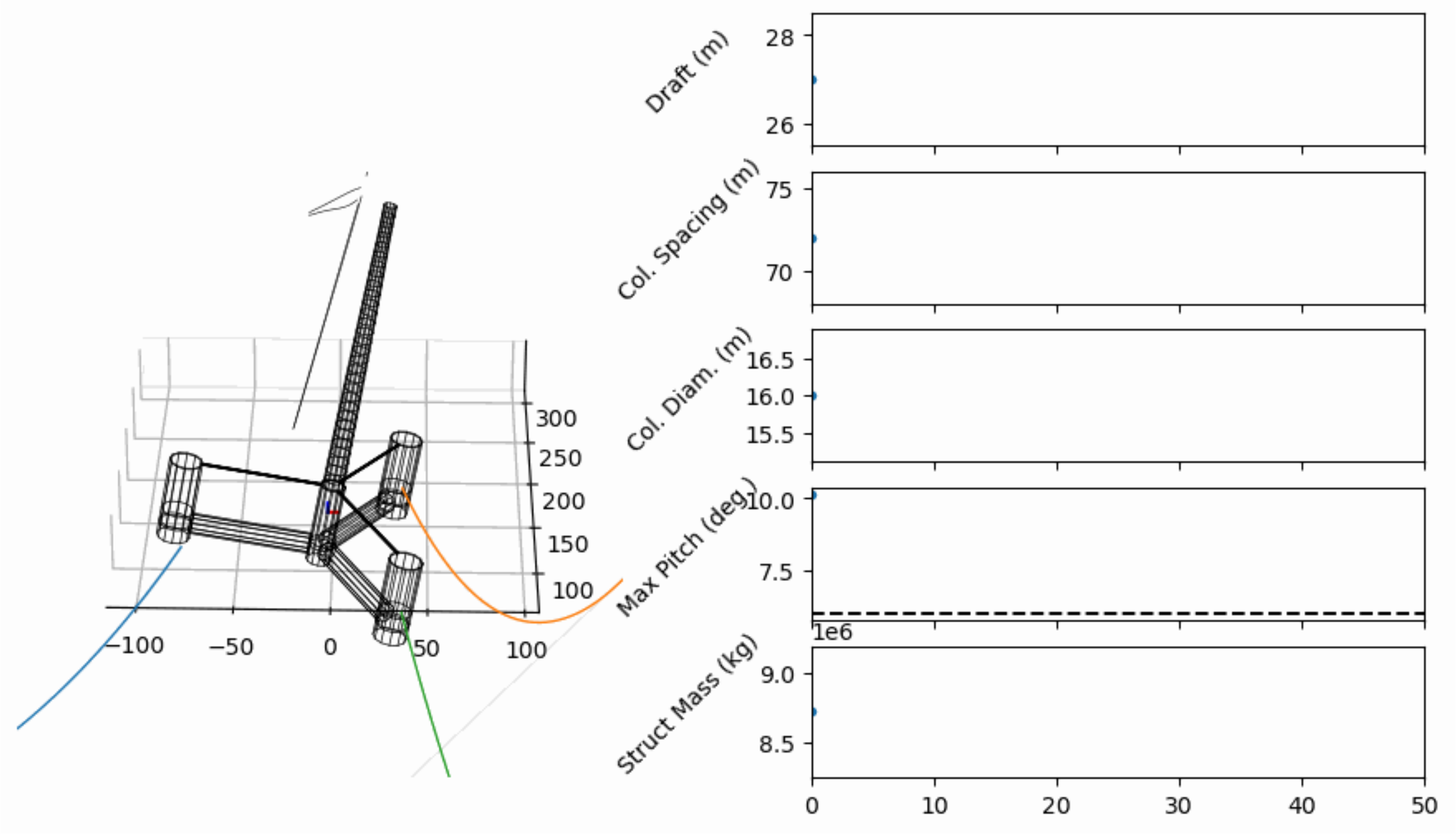
- DLC 1.6
- DLC 6.1



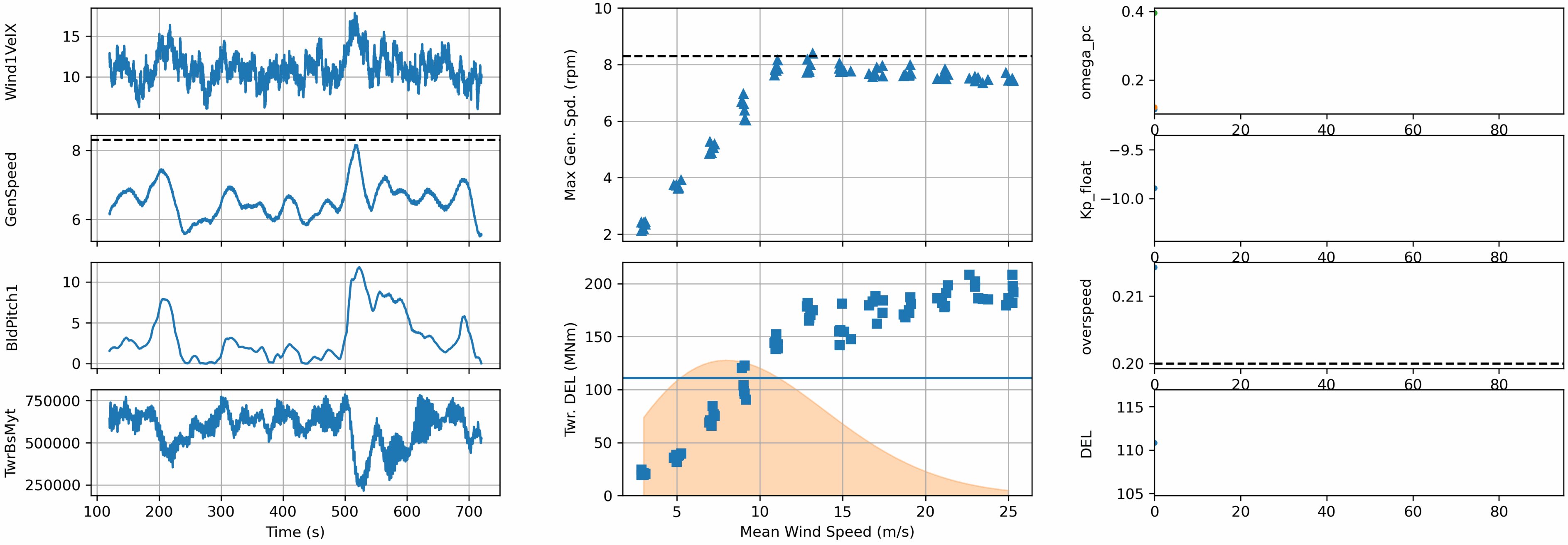
Demo: DLC-level Constraints



Platform Design Variables and Optimization



Controller Optimization



WEIS Roadmap

Entry point to WETO software tools

- Focus has been on usability:
- How would you have liked to learn to use WEIS?
- How can using this tool be made easier?
- How can we better explain the breadth and depth of all parts of this tool?

Workflows to solve new research problems

- WEIS is set up to be flexible for solving new problems and responding to research calls and requests
- Is there a new tool/technology that you would like to integrate into WEIS?
- What problem can a tool like WEIS help you solve?

pyNuMAD

Ernesto Camarena

Sandia National Laboratories

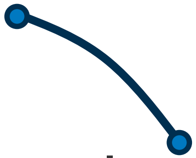
Exceptional service in the national interest

INTRODUCTION TO PYNUMAD

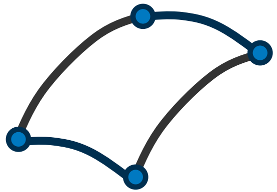
A python-based version of NuMAD for accessible and versatile wind blade design

pyNuMAD

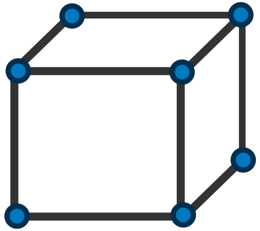
- pyNuMAD is the python version of NuMAD (Numerical Manufacturing And Design)
  - Simplifies the process of creating structural models of wind turbine blades
  - MATLAB-license independent
  - GUI-free to promote automated blade design optimization
  - Sandia exclusively developing pyNuMAD (NuMAD on static repo)
- Extended NuMAD beyond pure shell. Now includes:
  - Beam model with adhesives
  - Shell model with or w/o adhesives
  - Solid model with adhesives
- Beam and shell model workflows embeddable in system design optimization (*eg* WEIS)
- Solid models suitable for high-fidelity work (comparable to Large Eddy Simulations)



Beam element

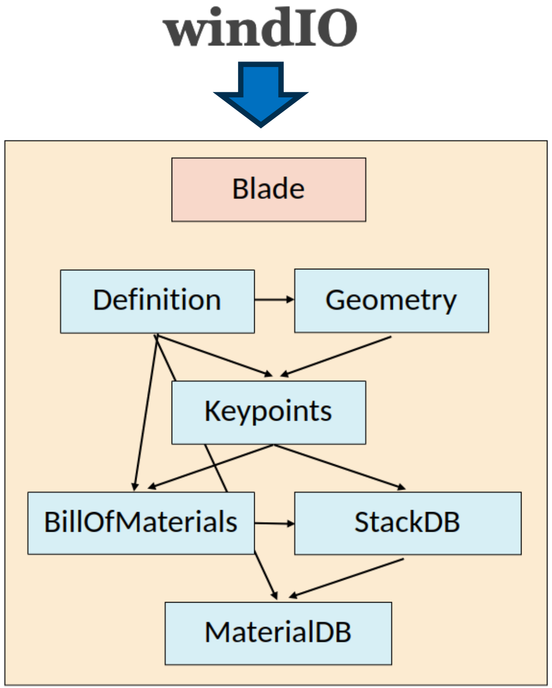


Shell element

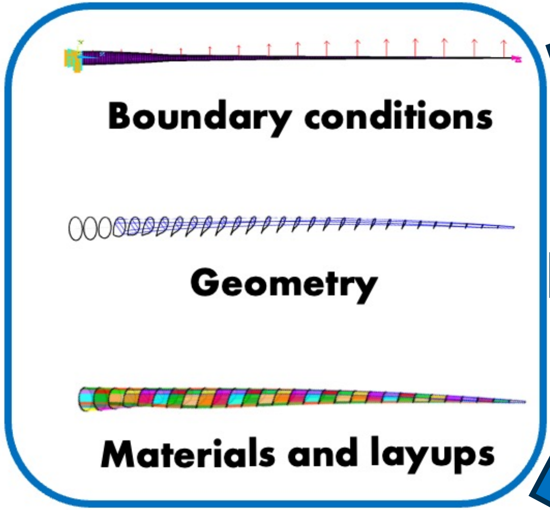


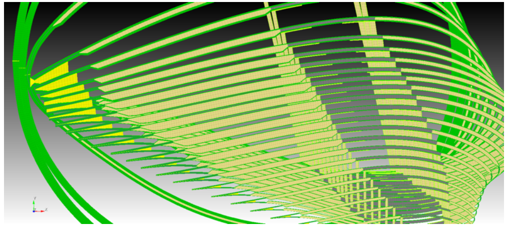
Solid element

pyNuMAD workflow

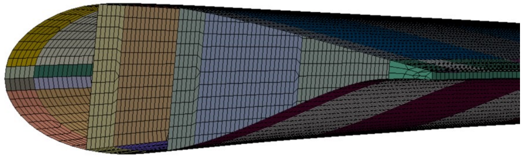


Blade object

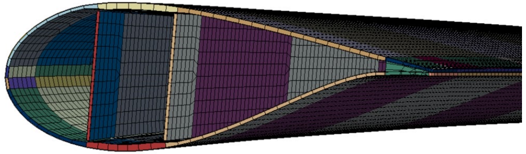




Beam model



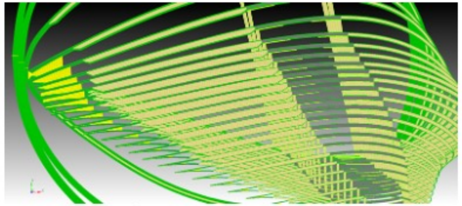
Shell model



Solid model

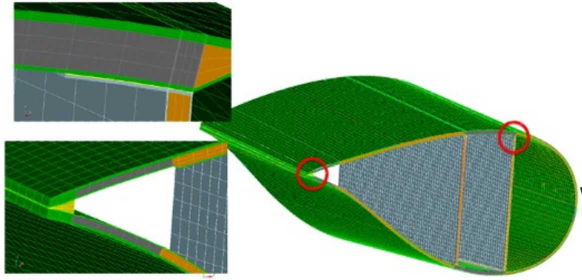
pyNuMAD Meshing

Cubit based mesher



2D Cross sections

BeamDyn

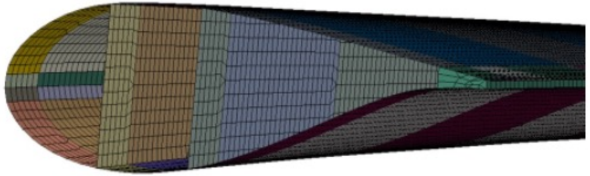


Solid element mesh

SIERRA MECHANICS

SIMULIA ABAQUS

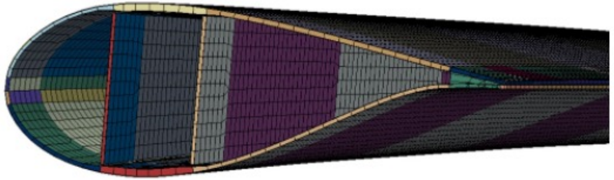
In-house mesher



Shell element mesh

Ansys

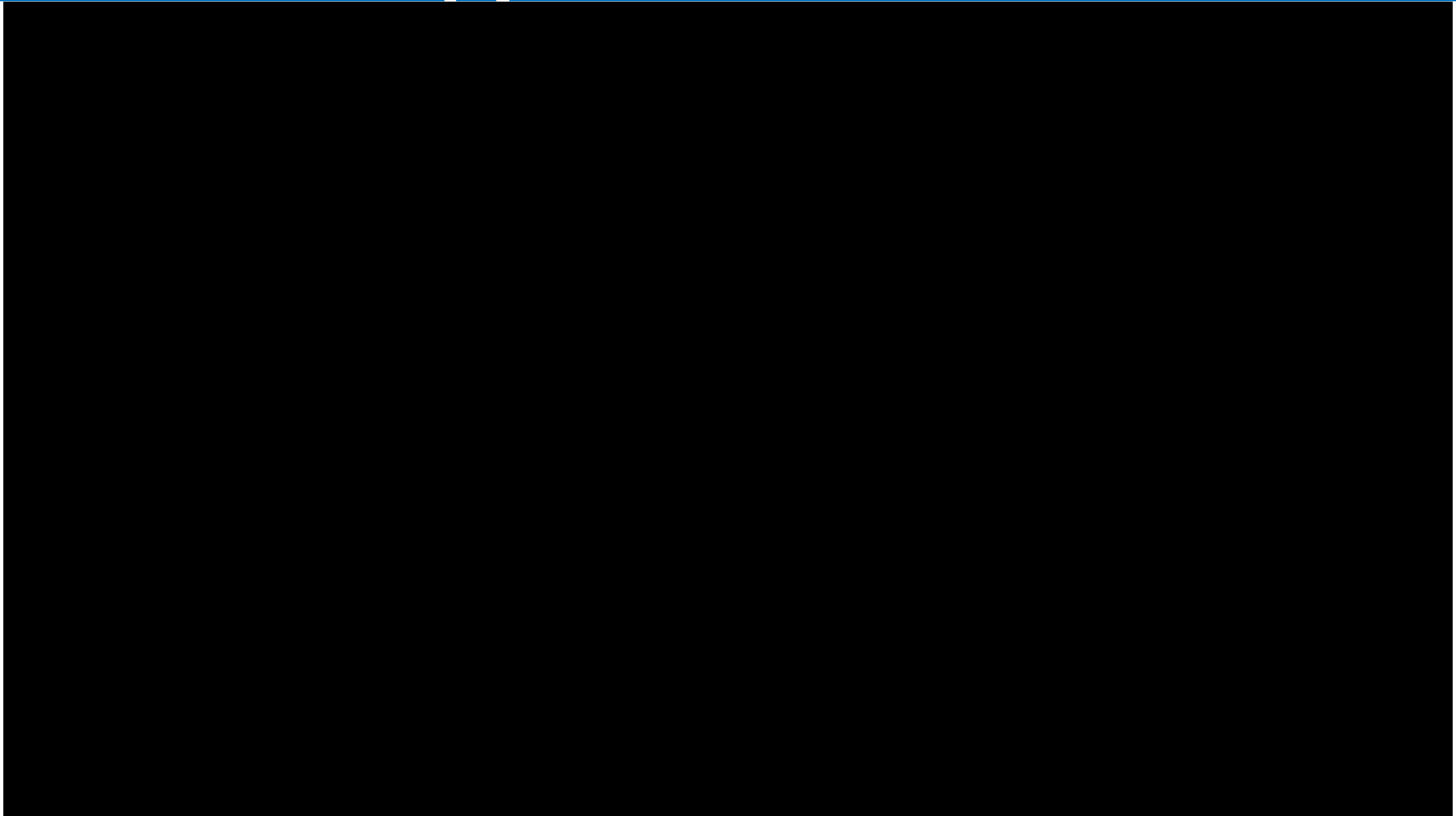
SIMULIA ABAQUS



Solid element mesh

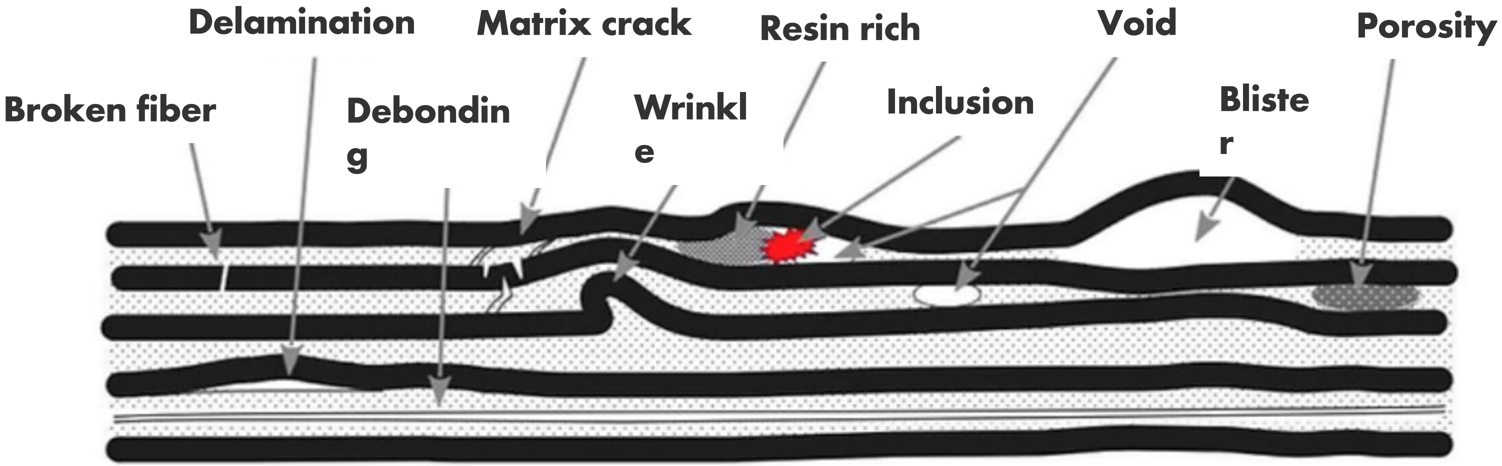
SIMULIA ABAQUS

pyNuMAD demo



FUTURE WORK

- Near-term
  - – Improve yaml reader to keep up with latest blade ontology changes
  - – Remove the need to have only two shear webs
- Long-term
  - Flaw and damage modeling
  - Probabilistic modeling
  - Assess new blade manufacturing methods



addcomposites.com

WindSE

Jeff Allen and Ethan Young

WindSE Mission Statement

WindSE is an open-source, Python package designed for the simulation and optimization of turbines and wind farms

Key Design Philosophies:

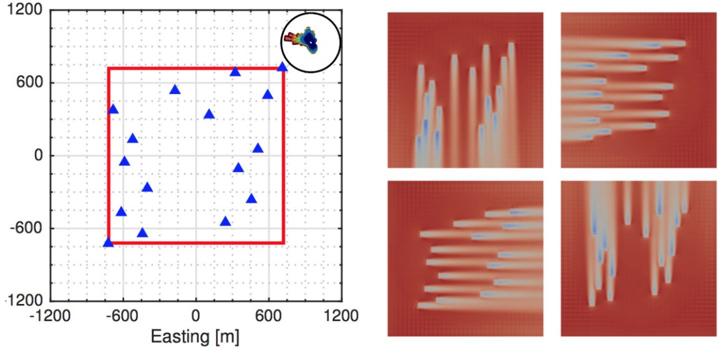
Strike a balance between engineering scale models and flow-resolving CFD that is ideal for optimization problems requiring simulated flow physics

We care more about capturing trends rather than chasing the highest possible fidelity

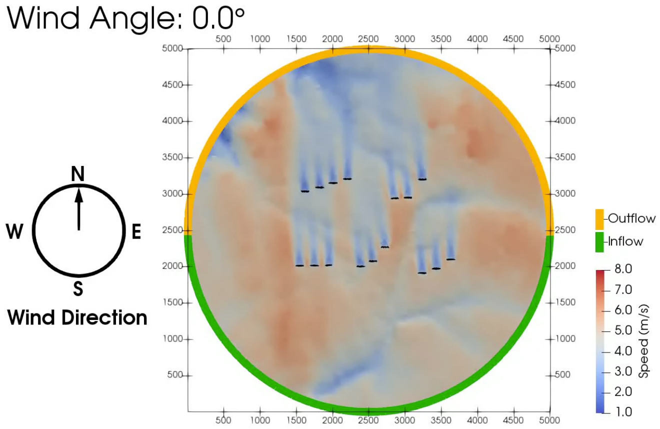
WindSE Features

https://fenicsproject.org/

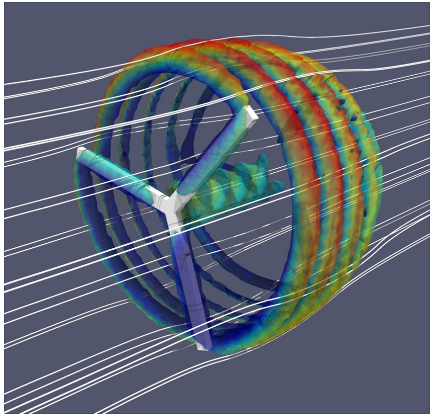
- Open source, python-based, **mid-fidelity** CFD model for UQ & optimization
- RANS/LES with 2D and 3D, steady and unsteady, flat and complex terrain capabilities
- **Automatic differentiation tools for gradient based optimization using adjoints**
- https://github.com/NREL/WindSE (Apache 2.0 license)



Layout AEP optimization



Complex terrain simulations


Actuator line simulations

Quick Demo

se_workshop_2024 — -bash — 127×63

```
Last login: Thu Jun  6 14:55:37 on ttys004
NOTICE TO USERS:
This is a Federal computer system and is the property of the United States Government. It is for authorized use only.  Users (a
uthorized or unauthorized) have no explicit or implicit expectation of privacy.

Any or all uses of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected
, and disclosed to authorized site, Department of Energy, and law enforcement personnel, as well as authorized officials of oth
er agencies, both domestic and foreign. By using this system, the user consents to such interception, monitoring, recording, co
pying, auditing, inspection, and disclosure at the discretion of authorized site or Department of Energy personnel.

Unauthorized or improper use of this system may result in administrative disciplinary action and civil and criminal penalties.
By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use.  LOG OFF IMME
DIATELY if you do not agree to the conditions stated in this warning.

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
eyoung$ initconda-intel
[conda-intel] eyoung$ conda activate windse
(windse) [conda-intel] eyoung$ cd Documents/repos/WindSE/se_workshop_2024/
(windse) [conda-intel] eyoung$ ls
3D_Skew_Steady.yaml output          skew_terrain.txt    workshop_sim.yaml
(windse) [conda-intel] eyoung$
```

workshop_sim.yaml

```yaml
general:
    name: workshop
    output: ["mesh","initial_guess","height","turbine_force","solution"]
    dolfin_adjoint: True
    debug_mode: True

wind_farm:
    type: grid
    ex_x: [-650, 650]
    ex_y: [-650, 650]
    grid_rows: 3
    grid_cols: 3

turbines:
    type: disk
    HH: 90
    RD: 130.0
    thickness: 13.0
    yaw: 0.0
    axial: 0.33

domain:
    type: box
    x_range: [-1300, 1300]
    y_range: [-1300, 1300]
    z_range: [0.04, 1300]
    nx: 32
    ny: 32
    nz: 16
    interpolated: true
    terrain_path: skew_terrain.txt

refine:
    warp_type: split
    warp_percent: 0.7
    warp_height: 260
    refine_custom:
        1:
            type: box
            x_range: [-780, 780]
            y_range: [-780, 780]
            z_range: [0, 780]

function_space:
    type: linear

boundary_conditions:
    vel_profile: log
    HH_vel: 8.0
    k: 0.4
    inflow_angle: 270
    # inflow_angle: [0, 360.0, 4]
    boundary_types:
        inflow: ["west","north","south"]
        no_stress: ["east"]
        free_slip: ["top"]
        no_slip: ["bottom"]

problem:
    type: stabilized
    viscosity: 5
    lmax: 50

solver:
    type: steady
    # type: multiangle
    save_power: true
    nonlinear_solver: newton

# optimization:
#     opt_type: maximize
```

Line 38, Column 11

WindSE Development Roadmap

- Advanced Automatic Meshing Tool – Automate mesh sizing and refinement to capture necessary physics with minimal problem size, adapt mesh around complex terrain and in inter-turbine regions, minimize burden on user to generate high-quality meshes
- AEP Estimation – Automate the calculation of AEP to take advantage of parallel computational resources. Use Gaussian integration to intelligently sample wind rose parameter space to minimize uncertainty with each new simulation
- Complex Constraints – Add ability to compute extra constraints such as Loads, Cabling, Costs, environmental impact, for more complete optimization. Perform LCOE optimization.
- Co-design Optimization – Simultaneous optimization of wind plant design and controls operation: layout, turbine design, sensor placement, controller (turn on all the controls)
- Support Developer and User Communities – Grow maturity of codebase and user community through workshops, modern software development practices, continued V&V efforts

Software Context

Software	Fidelity	Physics	Turbine	Wakes	Time	Loads
FLORIS	Low	Analytical	CCBlade Lookup Table	Superposition	Steady	Not Yet
FAST.Farm	Mid	Dynamic Wake Meandering	FAST	Superposition	Unsteady	Aeroelastic (IEC DLC capable)
WindSE	Mid	RANS/LES with adjoints	Actuator Disks Actuator Lines	PDE	Both	Flapwise Bending DEL’s
SOWFA	High	LES	Actuator Disks Actuator Lines	PDE	Unsteady	Aeroelastic
Nalu-Wind	High	LES	Actuator Lines Blade Resolved	PDE	Unsteady	Aeroelastic

Systems Engineering

Polls

Open Discussion

Systems Engineering

Raise your hand and we'll call your name to ask a question.

- **Discussion topics**
  - Prospective / new users:
    - What are your thoughts on the learning or onboarding process?
  - Experienced users:
    - What have been your primary pain points or bottlenecks?
    - What has worked or not worked in helping to integrate WETO software into your workflows?
    - How thoroughly do you understand the capability of the tools available in the WETO Software Stack?
    - What has helped or hindered your open-source contribution to the WETO Software Stack?

Thank you for your time today!

- Need help with a particular problem?
  - GitHub Issues or Discussions pages for any of the models
  - NREL User Forum (for NREL models): forums.nrel.gov

- Have further thoughts that you want to share? Send feedback to Rafael.Mudafort@nrel.gov
- How could we have done better? Send feedback to Rafael.Mudafort@nrel.gov

- Software repositories:
  - WISDEM: https://github.com/wisdem/wisdem
  - WEIS: https://github.com/wisdem/weis
  - pyNuMAD: https://github.com/sandialabs/pyNuMAD
  - WindSE: https://github.com/NREL/WindSE