WEC Development Project

Winter Semester 2025/2026

Study Manual

Project Name: **Optimus Syria**

Sub Project: **Load & Dynamics**

Authors:	**Matriculation number:**
Paresh Dulabhai Nakum	770552
Md Aman	770822
Arham Memom	770880

Supervisor:

Mr. Andreas Manjock

Date: 27.01.2026

Table of Contents

List of Abbreviations

BldPitch1	Blade Pitch angle of the Blade 1 [deg]
DLC	Design Load Case
DNV	Det Norske Veritas
FA	Fore-after direction
GenPwr	Generator Power [kW]
GenTq	Generator Torque [kN]
GL	Germanischer Lloyd
IEC	International Electrotechnical Commission
IEA	International Energy Agency
NREL	National Renewable Energy Laboratory
MExtremes	MATLAB-based generator for extreme-event tables
LSS	Low-Speed Shaft
OpenFAST	Opensource program for Fatigue-Aerodynamics-Structures-Turbulence
PSF	Partial Safety Factor
RootFxc1	Out-of-plane shear, in-plane shear, and axial forces at the root of blade[kN]
RotSpeed	Rotational Speed of the low- speed shaft
TipClrnc1	Blade 1 tip-to-tower clearance estimate [m]
TipClrnc2	Blade 2 tip-to-tower clearance estimate [m]
TipClrnc3	Blade 3 tip-to-tower clearance estimate [m]
TwrBsFxt	Fore-aft & side-to-side shear, and vertical forces at base of the tower [m]

List of Symbols

- **ρ** (rho): density [4] (Units: kg/m^3)
- ω(omega): angular velocity (Units: rad/s)
- C_p: Power Coefficient (Units: dimensionless)
- P: Power (Units: W or kW)
- T: Torque (Units: N . m)

List of Figures

1. Introduction

In the third semester of master's program in wind Energy Engineering at Hochschule Flensburg, we students engage in a collaborative project titled "development of a wind turbine". For this year we must develop a wind turbine for Syrian environmental condition. Turbines need to be reliable for that environmental condition which need to be reliable in setting up in Syrian region. The name of the project is Optimus Syria, and the rated power of the wind turbine is 5MW, with a rotor diameter of 160m and hub height of 100m. The specialty of this was the introduction of LiDAR as a feed forward controller. The initiative divides participants into a specialized team Each team focuses on distinct aspects of wind turbine design and provides a contribution in the development of wind turbines. Our team loads and dynamics, we were response for the simulation of the whole turbine to calculate the forces and moments on different components and sections of the wind turbine. So, with the help of our calculated values other teams check their design for safety and can easily optimize cost in terms of material and dimension. This manual is mainly for the work done by load and dynamic team what type of simulation we have done. What parameters do we have required for the calculation of load.

It's not about the physical behavior of a wind turbine but in a few sections, we really dive into some detail to Give you an idea of how complex Simulations is going to and how we can get a valid result with those simulations.

2. Objective

Our team's primary task is to provide reliable load data to other teams to achieve a project certification this process is divided into estimated load and load from simulations. The estimated loads are calculated by interpolating databases from the previous Optimus project and DNV.

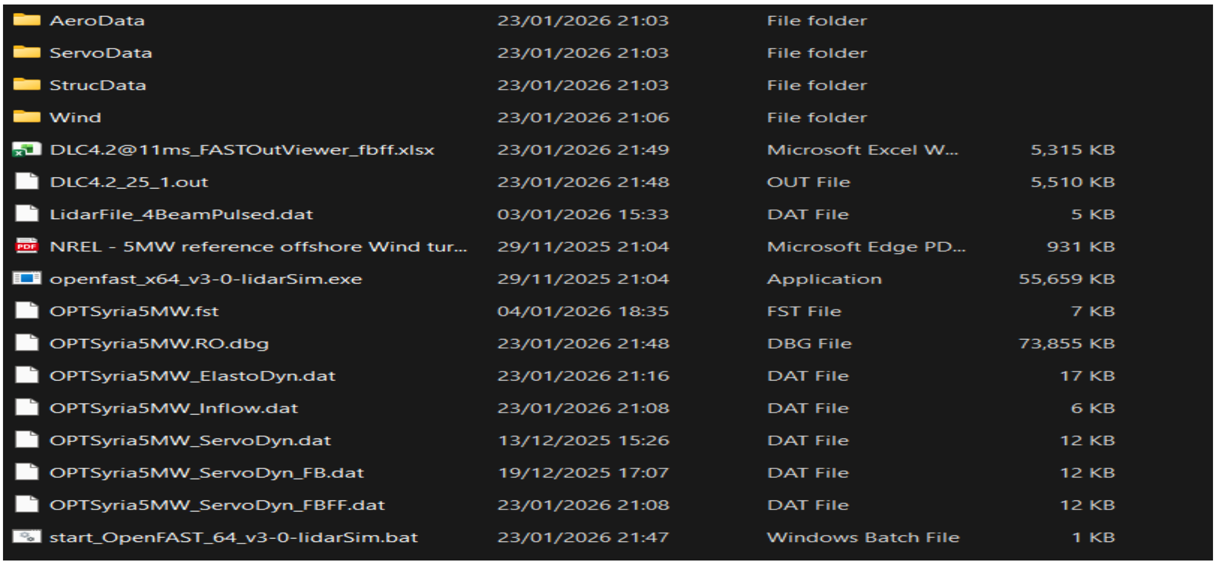
For this project needs to be recalculated several times because of the change in turbine Configuration, Controller configuration. The simulation will include several extreme events that will be experienced during its lifetime to prepare our turbine design for project certification we will implement these extreme events according to DNVGL [1].

In this project we will create various tools to calculate processes and present our findings the findings are important for design and evolution for the other teams in this project so it's essential to understand and use these tools effectively.

This study manual doesn't cover detailed information about channels, design load cases definition or abbreviation because it all things are already explained in loads and dynamic team report, So we are focusing on how we run simulation during our Project, how we do post processing of our result during the project, what different tools we have used in our project, how we have analyzed and studied our result.

This handbook is designed to guide you step by step to help you with setting up a model and how to proceed with corresponding to the guidelines and adjustment and requirement. This manual will consist of design load cases which are from DNV guidelines for Turbine standards of 2016.

3. TurbSim Simulation



AeroData	23/01/2026 21:03	File folder	
ServoData	23/01/2026 21:03	File folder	
StrucData	23/01/2026 21:03	File folder	
Wind	23/01/2026 21:06	File folder	
DLC4.2@11ms_FASTOutViewer_fbff.xlsx	23/01/2026 21:49	Microsoft Excel W...	5,315 KB
DLC4.2_25_1.out	23/01/2026 21:48	OUT File	5,510 KB
LidarFile_4BeamPulsed.dat	03/01/2026 15:33	DAT File	5 KB
NREL - 5MW reference offshore Wind tur...	29/11/2025 21:04	Microsoft Edge PD...	931 KB
openfast_x64_v3-0-lidarSim.exe	29/11/2025 21:04	Application	55,659 KB
OPTSyria5MW.fst	04/01/2026 18:35	FST File	7 KB
OPTSyria5MW.RO.dbg	23/01/2026 21:48	DBG File	73,855 KB
OPTSyria5MW_ElastoDyn.dat	23/01/2026 21:16	DAT File	17 KB
OPTSyria5MW_Inflow.dat	23/01/2026 21:08	DAT File	6 KB
OPTSyria5MW_ServoDyn.dat	13/12/2025 15:26	DAT File	12 KB
OPTSyria5MW_ServoDyn_FB.dat	19/12/2025 17:07	DAT File	12 KB
OPTSyria5MW_ServoDyn_FBFF.dat	23/01/2026 21:08	DAT File	12 KB
start_OpenFAST_64_v3-0-lidarSim.bat	23/01/2026 21:47	Windows Batch File	1 KB

Figure 1 Working Folder

TurbSim which means turbulent simulations, in the software we generate turbulent wind field for our turbines. To run this software, we need to create an input file. So, with the help of that input file, we can generate a wind field for our turbine, in the input file we set grids side, we set wind models is our wind model is ntm i.e. normal turbulent model our ewm i.e. extreme wind model or eog i.e. extreme operation of gust. We also set wind speed in this input file. After adjusting in our input file, we saved that file and start with working procedure of turbsim.

With the help of command prompt in our computer run turbsim and able to generate wind field simulation for our turbine. In the above Image we can see there is a folder named wind inside that wind folder we have our input file, and we have also our wind field generated file which we generated with the help of command prompt in our computer. And we must open our inflow file which is Shown in above image and set our conditions according to bts i.e binary turbulent file or .wnd file. And save the file by following this step we have successfully generated turbulent wind field for our turbine.

4. OpenFAST Simulation

OpenFAST is a powerful and flexible simulation tool used to analyze the complex dynamic interactions within wind turbines. It accounts for a wide range of physical processes and modeling fidelities, making it a crucial framework for wind energy research and development. At its core, OpenFAST acts as an integrative "glue code" that combines various computational modules to model aerodynamics, control and electrical system dynamics and structural dynamics.

This integration enables detailed simulations of coupled nonlinear aero-hydro-servo-elastic behavior in the time domain, offering insights into the performance and structural integrity of wind turbines under diverse operational conditions. OpenFAST supports the analysis of multiple wind turbine configurations, including two- or three-bladed horizontal-axis rotors, pitch- or stall-regulated systems, rigid or teetering hubs, and upwind or downwind rotors and it can simulate turbines with either lattice or tubular towers, catering to scenarios onshore or offshore, with fixed-bottom or floating substructures.

For this Optimus project we are working with OpenFAST v3.0. But previously in other optimus project they have used Openfast version 3.5 3. We are using OpenFast version 3.0 because this version is LiDAR simulation enabled version so for the Optimus Syria turbine we are developing turbine with a LiDAR Feedforward controller.

For running a complete Openfast Simulation we required a lot of different which we can see in the image. The image is the working directory for our simulation the

1. first folder that is Aerodata which consist of aerodyne and blade airfoils file which we got from Blade Aerodynamics team.
2. Second folder that is servo data which of controller and which is linked with servodyn.dat file, we those filed from feedback and lighter controller.
3. In the third folder that is our struck data this folder consists of structural file for our and for our tower.
4. Theelastodyne.dat file consist of turbine properties specification which we have to adjust by ourself with the help of system integrator.
5. The inflow.dat file and .Fst file are being adjusted by according to the conditions.
6. And servodyn.dat file which consist of time period of working of controller needs to be adjusted according to the design load cases of DNV.
7. In the last .bat executable batch file, so after configuring our all set we have to run this batch file to make a complete simulation and we'll get our.out result. Before running this batch file we have to make sure that the directory is correct and we can check the directory by opening this batch file in a notepad.
8. So in last we'll our out file and we will open dat out file in notepad and copy all the data and paste it in fastout viewer excel file and analyze our result.

So, in this we will have our result for our Openfast simulation in many we face aborting situation during running our simulation and that is due to some error from our side in the configuring of above points file. We must make sure that everything needs to be correct, but if we get any error or our simulation got, we will look for the solution which itself written Above the aborting notification in open fast.

5. Structure

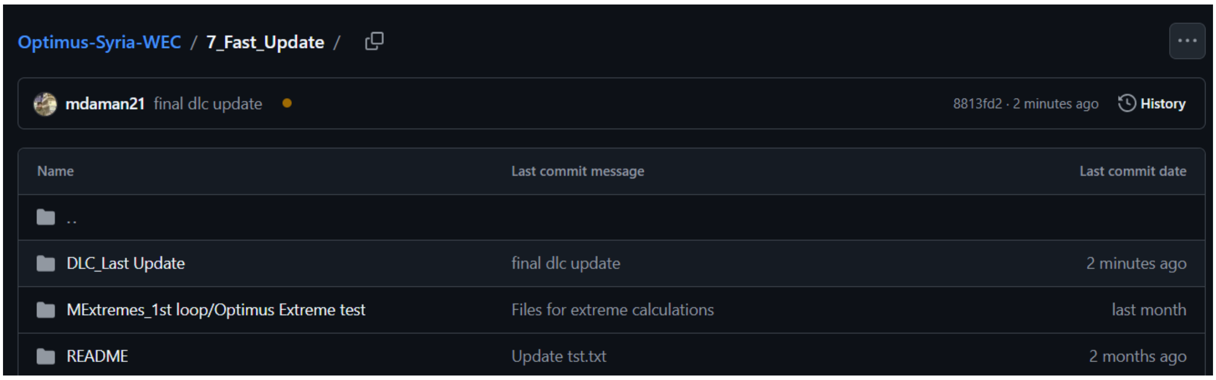


Figure 2 Working Folder on Github

For this project a structured directory system established to organize files in their respective categories and those files with their designated folders are available on GitHub to optimize workflow in our team and interconnection with other team specially with Lidar Feedback Controller team, so on GitHub there is a separate folder with name fast update and inside that folder there is a structured directory the design load cases which we have for this project. Now coming on the requirement our team was dependent on many other teams for collecting data like the first team on which we were dependent was blade aerodynamics Team 4 Airfoils Aerodyne blade structure files. The second theme on which we were dependent was Tower team for tower structure file and the 3rd team on which we were dependent were feedback plus feed forward lidar controller team for getting controller file. And last as per the information of system integrator and project management team we update Blade Elastodyn file that is the turbine properties file. Below there is a cyclic loop of our work demonstrated how we must do work and what steps we must take to run a complete simulation.

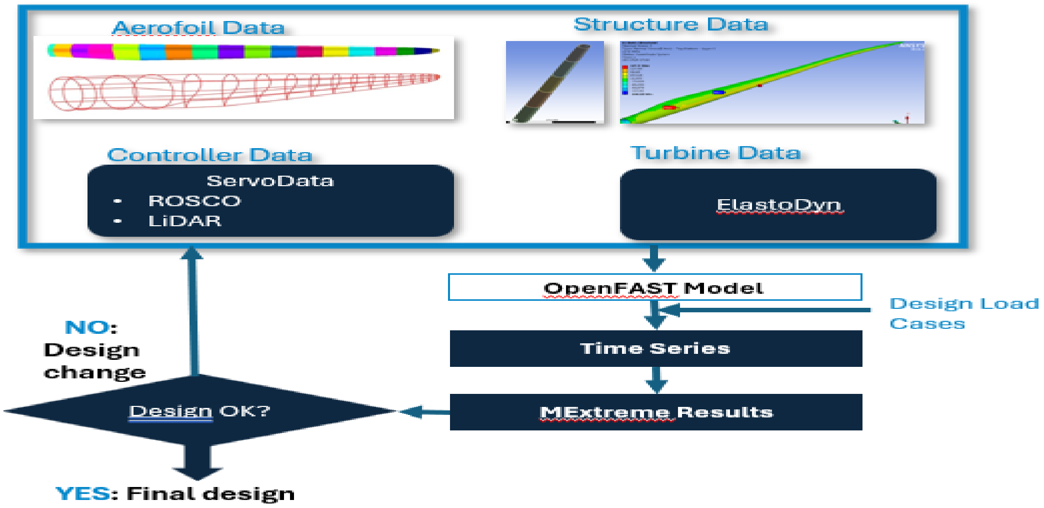


Figure 3 Working Loop for Simulation

6. DLC Data

Design Situation	DLC	Wind Condition	Marine Condition					Type of Analysis		Partial safety factor
			Waves	Wind and wave directionality	Sea currents	Water Level	Other Conditions:	Offshore	Onshore	
1) Power Production:	1.2	NTM Vin < Vhub < Vout	NSS Joint prob. distribution of Hs, Tp, Vhub	MIS, MUL	No Currents	NWLR or ≥ MSL		F/U	F/U	F/N
4) Normal shutdown	4.2	EOG Vhub = Vr ± 2 m/s and Vout or ETM Vin < Vhub < Vout	NSS Hs = E[Hs|Vhub]	COD, UNI NCM MSL	COD, UNI NCM MSL	COD, UNI NCM MSL	The timing of the gust and the shutdown event chosen for minimum 6 distinct points.	U	U	N
5) Emergency Stop	5.1	NTM Vhub = Vr ± 2 m/s and Vout	NSS Hs = E[Hs|Vhub]	COD, UNI NCM MSL	COD, UNI NCM MSL	COD, UNI NCM MSL	Azimuth position at the time of the emergency stop shall be randomly selected.	U	U	N
6) Parked (standing still or idling)	6.1	EWM Vhub = Vref	ESS Hs = Hs,50	MIS, MUL	ECM U = U50	EWLR	Yaw misalignment of ±8 deg Possible yaw slippage.	U	U	N
7) Parked and fault conditions:	7.1	EWM Vhub = V1	ESS Hs = Hs,1	MIS, MUL	ECM U = U1	NWLR	Fault that produces deviations from the normal turbine behavior while parked, including loss of electrical network.	U	U	A

Standard, DNVGL-ST-0437 – Edition November 2016

For this Optimus Syria project we have analyzed a total of five design load cases. Our design load cases cover the normal operation of a turbine, the normal shutdown of a turbine, emergency stop within a turbine, parked and idling condition of a turbine, and parked and fault in a turbine. Since it is not possible to generate an optimized turbine, so we are working with interpolated values for our calculation Which help us to generate some reasonable solutions for the turbine. We have analyzed all five DLC mentioned above and every case inside that DLC, that means like for DLC 1.2, it's a normal turbulent model and for that the variation in wind speed is from cut in velocity to cut out velocity with a separation of 2m per second and for each wind speed we have generated a total six out file and at different Randseed.

6.1 Fst file Updation

In this paper, you set up numerous basic configurations, which the simulation utilizes. It makes connections amongst the distinctive modules that Open FAST can utilize.

In this paper, you set up a numerous basic configuration, Which the simulation utilize It makes connection amongst the distinctive modules That open fast can utilize

Simulation Control

The most important aspect in the section is the need to define the length of the simulation and the time increment. In each of these simulation the time step was fixed to 0.1 seconds And the length of the simulation varies depending on the load cases as the recommendation it is always wise to extend the length of the simulation to at least 10 seconds after the simulation start in the consideration of the time open first Text to stabilise tour numerical simulation there is no physical consideration in the context

Feature switches and Flags

Open Fast is the modular simulation tool, so you may add different modules based on your turbine's location (onshore or offshore) or the information provided about the turbine (example BeamDyn). In this section, you specify which modules are required for the simulation and which variant should be utilised.

Environmental Conditions

To create a realistic simulation, this section allows you to choose environmental factors such as add density and viscosity.

Input Files

For the previously described the modules to function correctly, multiple input files are required Here, their paths are outlined.

6.2 Structural-Properties: ElastoDyn.dat

This file specifies numerous turbine properties. The initial conditions settings are determined by the design load case or an explicit turbine behaviour, whilst the remaining setting, such as turbine configuration (Figure 4), describe the turbine's actual components. the settings are the same for all load cases; thus, we created a text file form the start to document and gather every value update to optimise our workflow in terms of the time-use and Confusion about which values are up to date, and which are not.

Degrees of Freedom

Degrees of Freedom refer to a component's ability to move or deform. Only if all of them are set to true can you simulate a real body. It is advisable to only enable degrees of freedom if suitable parameters such as stiffness and the mass have already been defined. Otherwise, you will get deceptive results.

Initial Conditions

There characteristic indicate the turbine's condition and simulation time 0, and there are extremely dependent on the situation your turbine should encounter in simulation.

The blade pitch of DLCs 6.1 and 7.1, which are used in the park and idling mode, should be set to 90 degrees. For a power production load scenario, it is beneficial to begin with the rotational speed to establish a realistic point in the simulation.

Turbine Configuration

In this section of the file, you define numerous turbine properties related to geography, such as the number of blades and the cone angle.

Mass and Inertia

As the weight and geometry of the various components Influence the simulation of physical bodies, several masses and inertias must be included in the section.

Blade, Rotor Teeter, Drivetrain, Furling, Tower

Each of these pieces describes a single component. For the blade and tower, you describe the number of the nodes (Simulation points on the component) and the link to more detailed data in the StrucData. But for the drivetrain, just four parameters (gearbox ratio and efficiency, torsional spring and damper) are required.

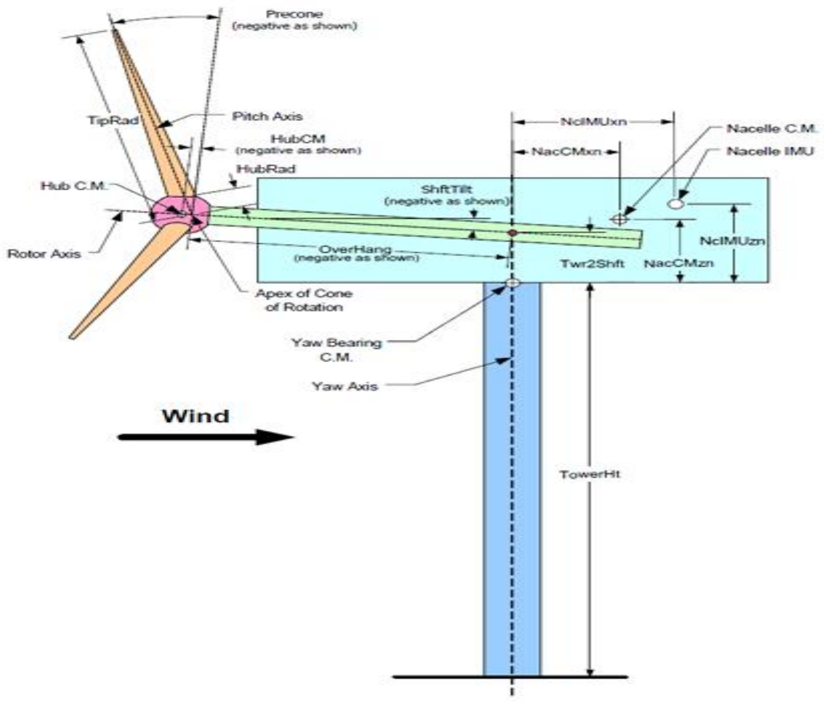


Figure 4 Layout of a Convenient, Upwind, Three-Bladed Turbine

6.3 Controller-Setting: ServoDyn.dat

This file contains only information on control parameter. It is possible to configure several modes for each controller, most of which allow you to choose between no controller and a pre-designed controller This file also has three generator model options, from which we choose 'simple variable speed control'.

Some DLCs need manual manoeuvres that overrule the controller.

Pre-designed Controller

The pre-designed controller can be configured in various ways. In our scenario, the Control Team provided an appropriate controller. you must specify the location to the controller file in the 'Bladed Interface' section of this file. The separate controller must be activated to integrate the pre-designed control; otherwise, the initial state from the 'ElastoDyn' will remain constant during the simulation. A link to 'ServoData' is mandatory.

Manual Maneuvers

The best example of a manual manoeuvre is DLC 5.1 -Emergency Stop. This DLC simulates a manual shutdown during electricity production. To shut down, turn off the generator, pitch the blade out of the wind to a pitch angle of 90 degree and activate the high-speed shaft brake.

The manual manoeuvre requires a starting point from which to begin the manoeuvre as well as an end pitch position. The time required relies on the pitch rate, which is defined by the pitch drivers and should not be very high. If the blade pitches too fast, the turbine generates negative thrust, resulting in excessive bending moment to the front for the tower top and bottom.

The generator control covers general attributes such as generator efficiency and manoeuvre parameters such as the generator's on/off timing.

The high-speed shaft brake only requires three settings. The deployment time, maximum braking torque, and time required to deploy that torque. The maximum braking torque should be comparable to the maximum torque of the rotor. The figure shows that the generator is turned off after 110 seconds, resulting in an increase in rotational speed. As a result, the aerodynamic break with a manual pitch control manoeuvre is deployed, followed by the deployment of the high-speed shaft brake at 118 seconds, at which point the rotational speed has already been reduced by half of the rated rotational speed.

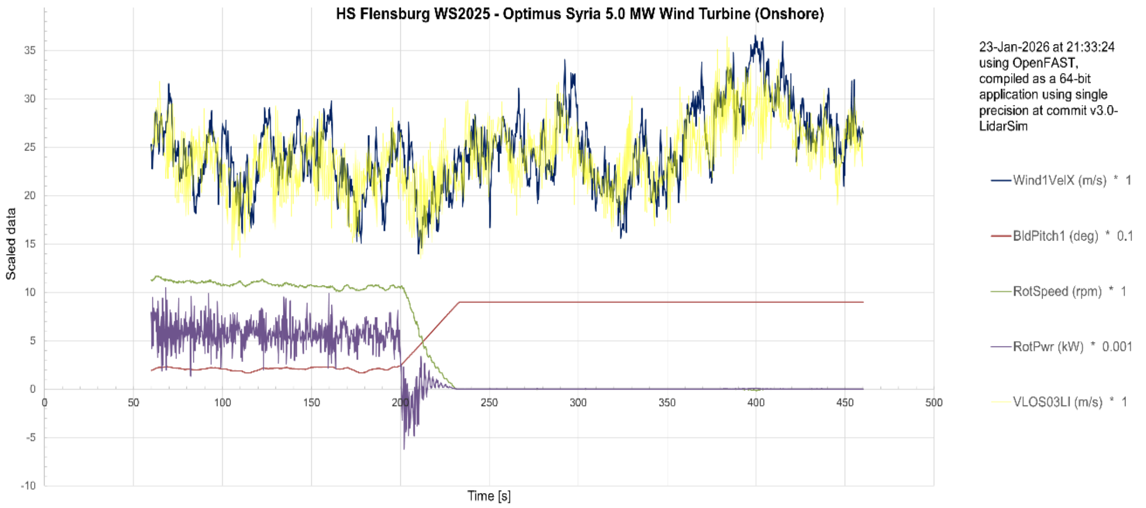


Figure 5 Manual Maneuver in the DLC 5.1 - Emergency Stop

6.4 Wind: Inflow.dat

This file defines the wind properties for the simulation. You can activate or deactivate various sections of the file by setting different modes for the wind kinds, as shown below:

: Wind Type = 1: For a consistent wind speed

Used for test runs.

: Wind Type = 2: Specifies wind behaviour without turbulence

Calculates guts or natural frequency

Can be set up using excel

: Wind Type = 4: Because we are working with Openfast v3.0 so we need to generate .wnd file.

Use turbSim to set up realistic wind with turbulence and constant mean wind speed.

Suitable for power production load scenarios.

This file must include a path to the corresponding wind file. Because wind files are huge and utilised in multiple load cases and turbine models, they are typically stored outside of the model folder in their own directory.

6.5 Turbine Data

The various turbine data folders will be detailed in subsequent chapters. They are the same across all DLCs and represent specific components.

6.6 Structured Data

This directory may contain files about the structure of the Blade, Tower, Soil, Earthquake Forces and Structural Control. Our simulation only includes the 'ElastoDyn' module, thus information about that module is saved there.

The 'ElastoDyn.dat' file in the DLC-Data already represent several turbine settings. Blade and tower are the most significant structural component. To accurately replicate their structural behaviour, more information is required and kept in distinct files for the blade and tower.

6.7 OptSyria_blade.dat

The blade is a lengthy component whose geometry changes regularly from root to tip. That is why the structural qualities must be distributed along the blade.

The distribution of mass and stiffness has the greatest influence on load and deflection. Because the geometry and spar caps are not round, the stiffness distribution is separated into flap and edge stiffness. To simulate a cutting-edge blade, the blade's instructional twist can be applied.

To accurately reflect the blade's movement, many natural shapes can be stabilised. Two natural shapes are required in the flap wise direction, whereas one is sufficient in the edge wise direction. Due to the high stiffness caused by geometry, it is believed that the second natural frequency will not be attained. It is also feasible to employ various damping and tuning parameters, but we choose to go with the properties provided by the rotor blade team.

Team involved: Blade Aerodynamic and structure Team.

6.8 OptSyria_tower.dat

The tower has a simpler geometry than the blade, but we still need to consider the distribution of the mass density and stiffness in the fore after (FA) And side to side directions due to the changing diameter.

Because of the circular shape of the tower, both stiffness is usually equal. Because of this, two natural shapes are required for both bending directions. Similar damping and turning factor are applicable here as well.

Team involved: Tower Team.

6.8.1 ServoData

This folder contains the controller files, which are provided by the Controller Team. However, it is critical to specify the right path for these files because the simulation would not function without them

Teams involve: Control.

6.8.2 AeroData

The AeroData contains the fundamentals for simulating aerodynamic. We started with AeroDyn Version 15 and converted to version 14 later owing to certain difficulties with the blade parameter. Working with version 15 is highly recommended because it takes a more comprehensive approach.

Polar files describe the blade's shape, Whereas the AeroDyn15_blade.dat file contains blade parameters such as sectioning, chord, and twist. The AeroDyn15.dat file refers to the polar files and has a section on tower division. These contributions are often provided by the Blade Team.

Teams involved: Rotor Blade and Tower.

6.9 Wind folder

The wind folder contains several wind files. There are two types of wind files that are used to simulate design load situation.

6.10 Uniform wind – *.wnd-file

This wind file is particularly useful for wind scenarios in which the wind changes at a certain moment and behaves in a specific way. A significant increase or reduction in wind speed, as well as a shift in wind direction, are got instances of such behaviour. In these instances, the wind change dominates the loads, therefore the turbulence present in genuine wind events can be ignored. As a result, the *.wnd files are reduced to a simple text file with the table type shown below:

6.11 Outputs

OpenFAST produces two types of outputs. The most interesting post-processing outputs are the loads and control parameters generated throughout the simulation, while summary files can be used to ensure your preprocessing inputs.

6.11.1: Summary files – Pre-Processing - *.sum

The summary files provide previously provided information as a summary. It is useful for verifying your input and gaining an understanding of the physical properties.

In our situation, it assisted us in determining mass dispersion and whether we had correctly implemented them. The ElastoDyn *.sum file contains the rotor blade mass; however your input is merely a mass distribution, which can be difficult to comprehend.

6.11.2. Output-files – Processing - *.out

The information obtained during the simulation is stored in the output files. The time is documented in the *.fst file, whereas the information is documented in the corresponding file. It is recommended to establish the desired output from the start to avoid changes in post-processing, which save time. The OpenFast documentation has a list of all accessible module output sensors.

Wind – inflow.dat

This file can only define wind-related outputs, such as the wind vector.

Structural – ElastoDyn.dat

This file allows you to define structural-related outputs that refer to certain components and coordinate systems. Load and motion parameters can be specified here.

Several areas must be considered when doing design checks on the blade and tower. These sections can be defined within this file. While the blade parts correspond to the sections specified in the ElastDyn_blade.dat, the tower sections are determined by the number of nodes defined in the ElastoDyn rather than the ElastDyn_tower.

Control – ServoDyn

In this file, you define the control-related output. It is possible to obtain information on a variety of control orders, but in our instance, we simply documented generator torque and power.

Team involved: All teams

6.12 Plausibility Checks

Plausibility checks are required to validate turbine behaviour. Several parameters can be evaluated.

Numerical Instability

OpenFAST simulates using a variety of numerical methods. It makes sense to determine whether the extreme value occur at a reasonable time and with a reasonable peak. It is advised to start the output after at least 10 seconds. For more complex applications or models, openFAST may require longer time to achieve steady behaviour.

Validation of the input parameters

This mostly refers to the control parameters for manual movements but also includes the initial conditions and wind files.

Tower strikes

The AeroDyn15 includes tower geometry to calculate the associated aerodynamic load, therefore the simulation can simulate tower impact directly. This is not the case with AeroDyn14; However, you must manually determine whether the blade tip tower clearance is less than the tower radius at the appropriate height.

7. Post-Processing with MExtremes

MExtremes acts as a tool for the analysis and the processing of the output data generated from the Open FAST simulation. MExtremes is used for the calculation of extreme value for example maximum and the minimum load values, for the various output channel values used during the simulation.

MEXtreme is a tool used for the analyse of output data which is helpful in identifying the important load value and enabling the assessment of the wind turbine performance and safety for various condition MExtreme stream Extensive tables and visual charts for the extreme value thus providing important information for assessment of the wind turbine limitation.

The utilisation of MExtreme ensures the evolution of the all-load conditions which is important for the designer and meeting the industry safety standards am act as an important tool for the validation of the output data generated from the simulation.

7.1 The Installation

The installation process for MExtreme team was difficult due to scarcity of clear and thorough documentation. However, with our professor's instruction and the assistance of the past member of the load and dynamic team the manual book was useful. As a result, below are full walkthrough steps that I am pleased to share with incoming student. You can also ask for mwloads software for making these calculations. For that you need to write Ms Carlotta.

- After numerous trial and errors, it is usually suggested to instal MATLAB version R2011b.
- After installation in Windows, navigate to "C:\Program Files\MATLAB\R2011b" and create a new folder called "MExtremes". Inside this folder, place "MEtremes_win64.exe".
- To set the path to "C:\Program Files\MATLAB\R2011b" in MATLAB, open the file and click set path. Then, in the new box, click 'Add Folder' and save the changes.
- The file "source" can be found in the extracted file "MExtremes_v1.01.00a.exe" from the MExtremes | Window Research | NREL folder. from the NREL website, will explain with a photo later.
- Copy the" Source" folder from the MExtremes folder and place it in "C:\Program Files\MATLAB\R2011b" as seen in the figure below.
- Reopen MATLAB and select the "Set Path…" from the file menu. Choose "And with Subfolders…" Go to "C:\Program Files\MATLAB\R2011b" and choose "Source" folder that you added in the previous step. Save and close. You must do this every time, and you launch Matlab.exe
- Copy and paste the "CertTest" folder to "C:\Program Files\MATLAB\R2011b".
- You must specify the working directory where you desired MExtremes post processing calculation is in MATLAB using the Current folder drop down menu.
- To run the MExtremes, place your output file in the data folder and specify them in the ". mext" input file.
- Program runs by this command:
  settingsFile = 'Put_the_file_name_here.mext'
  MExtremes (settingsFile);

- **In the command window of MATLAB.**
  You must ensure that you follow the path depicted dead in the graphic below.
  Make sure the ". mext" file is in the current folder.

7.2 Input-File for running MExtremes

The Major purpose of using MExtremes for post processing is to find the most extreme values, such as maximum and minimum loads, for each load situation in all output files. To accomplish this, users must create a file with the extension ". mext" to act as an MExtremes input, in addition to the simulation output files.

This ". mext" file serves as a configuration file, defining the channels and parameters to be evaluated. Proper file preparation is vital to ensuring accurate and efficient simulation data processing, which allow for precise insights into the wind turbine's critical load scenarios.

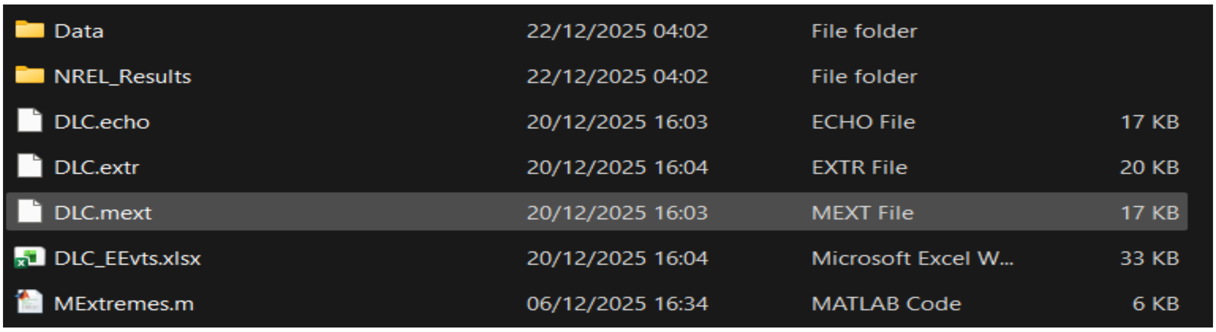


Figure 6 Working Directory MExtreme

7.2.1 Input-Data Layout

The input data layout an important portion of the ". mext" files since it dictates how the input data is formatted and interpreted by the MExtremes program. This section guarantees that wind turbine simulation data is post processed accurately and meaningfully by providing the placement of the title, Channel name, units and data, as well as the scale, offset and processing type.

The input data layout section defines the structure of the input data file which includes:

- Locate the title, channel names and unit in the files.
- Where the data starts.
- Specify scaling upset and processing type for each channel.
- This session guarantees that the software understands the how to read and interpret the output data from the Open FAST, which is required for accurate post processing.

Types of Channels titles:

Here are some popular types of channels. Titles that depended on the modules used in the simulation in OpenFast such as, AeroDyn, ServoDyn, ElastroDyn.

Structural Response:

For example, Blade Tip displacements in the X, Y, Z direction which symbolizes in **"TipClrnc1"** and Tower base moments which symbolize in **"TwrBsFxt"** in the X-direction.

Aerodynamic Loads:

For example, Blade root forces in the X-direction which symbolizes in **"RootFxc1"**.

Control System Response:

For example, Generator torque which symbolizes in **"GenTq"** and Blade pitch angles which symbolizes in **"BldPitch1"**.

Rotor and Generator Dynamics:

For example, Rotor speed which symbolizes in **"RotSpeed"** and Generator power which symbolizes in **"GenPwr"**.

7.2.2 Calculated Channels

The calculator channel section is used to define additional channels that are calculated from the raw data in the input files. These channels are not physically measured but rather derived from the mathematical calculation. This is especially useful for studying coupled effects like resultant forces or moments which are frequently used in wind turbine design and analyse. The rotation of the component and corresponding bearing frequently distinguishes between radical and axial loads. An additional calculator channel might be a realistic tower tip clearance as the Open FAST sensor describes the distance between the blade tip and tower axis, ignoring the tower diameter. A computed channel could follow the tower geometry.

The Key Components of the Calculated Channels Section:

● **NumChan**:

▪This specifies the number of calculated channels.

▪Each calculated channel is defined by a title, units, and a mathematical equation.

● **Seed**:

▪This is an integer seed used in the random number's generator.

▪Used for the stochastic calculation to ensure the reproducibility in simulation with randomness.

● **Col_Title, Units, Equation**:

▪Each calculated channel is defined by:

▪Col_Title: The name of the calculated channel (e.g., RootFxyb1).

▪Units: The units of the calculated channel (e.g., (kN) for kilonewtons).

▪Equation: The mathematical formula used to compute the channel. The formula typically references other channels from the input data.

Examples of Calculated Channels Resultant

```
-----  Calculated Channels  ----------------------------------------------------
8                 NumCChan          The number calculated channels to generate.
1234567890        Seed              The integer seed for the random number generat
Col_Title   Units    Equation       Put each field in quotes.  Titles and units ar
"RootFxyb1" "(kN)"   "sqrt( FileInfo.Time(:,18).^2 + FileInfo.Time(:,19).^2 )"
"RootMxyb1" "(kN-m)" "sqrt( FileInfo.Time(:,21).^2 + FileInfo.Time(:,22).^2 )"
"LSShftFxy1" "(kN)" "sqrt( FileInfo.Time(:,28).^2 + FileInfo.Time(:,29).^2 )"
"LSShftMxy1" "(kN-m)" "sqrt( FileInfo.Time(:,31).^2 + FileInfo.Time(:,32).^2 )"
"RootFxyc1" "(kN)"   "sqrt( FileInfo.Time(:,24).^2 + FileInfo.Time(:,25).^2 )"
"RootMxyc1" "(kN-m)" "sqrt( FileInfo.Time(:,26).^2 + FileInfo.Time(:,27).^2 )"
"YawBrFxy1" "(kN)" "sqrt( FileInfo.Time(:,38).^2 + FileInfo.Time(:,39).^2 )"
"YawBrMxy1" "(kN-m)" "sqrt( FileInfo.Time(:,41).^2 + FileInfo.Time(:,42).^2 )"
```

Figure 7 Resultant Calculation

Here are some samples from the file and the relevance.

1.RootFxyb1 (kN):

Equation: “sqrt (FileInfo.Time(:,18).^2 + FileInfo.Time(:,19). ^2)”

Explanation: This determines the resultant force in the XY plane for blade 1. It applies the Pythagorean theorem to combine the root forces X and Y components (RootFxb1 and RootFyb1).

Purpose: This is useful for determining the total lateral force acting on the blade root, which is critical to structural stability.

2.RootMxyb1 (kN-m):

Equation: “sqrt (FileInfo.Time (:, 21). ^2 + FileInfo.Time(:, 22).^2)”

Explanation: This calculates the resultant moment in the XY plane for blade 1. It combines the X and Y components of the root moment (RootMxb1 and RootMyb1).

Purpose: This aids in determining the bending moments at the blade root, which are the critical for fatigue and strength analysis.

3.LSSFyza1 (kN):

Equation: “sqrt (FileInfo.Time(:,29).^2 + FileInfo.Time(:,30).^2)”

Explanation: This calculates the resultant force in the YZ plane for the low-speed shaft (LSS). It combines the Y and Z components of the LSS force.

Purpose: This is used to determine the radial forces operating on the low-speed shaft which is important for the drivetrain design.

4.TwrBsMxy1 (kN):

Equation: “sqrt (FileInfo.Time(:,52).^2 + FileInfo.Time(:,53).^2)”

Explanation: This calculates the resultant moment in the XY plane at the tower base. It combines the X and Y components of the tower base moment.

Purpose: This is important for assessing the total lateral moment acting on the tower base, which is crucial for foundation design.

5.YawBrMxy1 (kN):

Equation: “sqrt (FileInfo.Time(:,46).^2 + FileInfo.Time(:,47).^2)”

Explanation: This calculates the resultant moment in the XY plane at the yaw bearing. It combines the X and Y components of the yaw bearing moment.

Purpose: This aids in understanding the moments acting on the yaw system which is required for yaw mechanism design.

7.2.3. Extreme Events

The '. mext' file's Extreme Event section is vital for analysing peak loads and extreme situation in wind turbine simulation. This section specifies the software's channel, wind speed ranges, and output format to reliably recognise, analyse and report extreme event

-Ensure all 'ChanTitle' entries match OpenFAST outputs precisely. Also, make sure to set the proper amount of channel. Define the 'ChanList' entries in the 'TableName' section using their associated parameters.

The '. mext' file's Extreme Event section has options like TableName, Chans, and ChanList. They identify how extreme event tables works and what options are added in each table. Let's go to each of these factors in detail, explaining their purpose and significance.

1:TableName

◦ Purpose: Identify the name of the extreme event table

◦ For example, bld_root,Hub_rotor, and Tower_Top.

◦ Significance: the table name works as the descriptive tag for the analysed channel. It identifies the wind turbine relevance (e.g., blade root, hub, tower top)

◦ Examples include bld_root, which focuses on extreme events at the blade root.

2: Chans

◦ Purpose: This parameter identifies the number of channels to be added in the extreme event table.

◦ for example, 12 for bld_ root, 6 for tower_top.

◦ Significance: This parameter is used to identify the number of channels which are included in the table. Every channel shows a type of measurement, such as forces, movement and displacement

: for the case of bld_root, there are 12 channels which shows the forces and moments in different routes at the root of the blade.

: for the case of tower_top, there are 6 channels, which represent the forces and moments at the top of the tower.

3: ChanList

◦ Purpose: It specifies the list of channels that will be included in the extreme event table.

◦ Example: for bld_root includes 8 19 20 21 22 23 24 25 26 27 158 and 159.

◦ Significance: The ChanList identify the channels to be analysed in the table. The channels correspond to measurement (e.g., forces, moments) related to the component or aspect under study.

◦ Example

- bld_root, the ChanList includes channels like:
- RootFxb1: Force in the X-direction at the blade root.
- RootFyb1: Force in the Y-direction at the blade root.
- RootMxb1: Moment in the X-direction at the blade root.
- RootMyb1: Moment in the Y-direction at the blade root.

4: InfoChans and InfoChanList

The information channel is useful for validating the DLC. Sensors can be defined here and used for each severe event. We choose to record the time, wind speed, and blade pitch.

7.2.4. Input files

Make sure your files are always stored in the Data folder by specifying the path where you put them. To efficiently arrange them, make subfolders inside that folder according to your simulations date or loop number.

It's more manageable, in my opinion to separate the file into their respective DLC directories rather than labelling them according to their DLC, but you have the option.

Finally, to reduce the amount of space the output files take up in the final table, keep their name as brief as you can.

Important Notes:

- the channel name includes capitalisation and formatting, must precisely match those in the "out" file.
- Secondly, the unit always match exactly; even a small change such capital or lower-case letters are prohibited.
- The safety factor is shown by the "PSFtype" parameters, which are particularly based on the design load scenario.
- As seen in the illustration, you always ensured that there is no space at all data while combining DLC.

7.3. Validation of the Extreme Load Table

A post processing tool which creates extreme load table is called MExtremes. To make sure that the MExtreme input file is error free and that the DLC and turbine data have been implemented properly two types of check must be performed after MExtreme has been run.

7.3.1. MExtremes Input File Validation

By selecting a single extreme load and comparing the MExtremes result with the corresponding output file the input file can be verified. We had some trouble specifying the input data layout which resulted in a very heavy and meaningless table. After altering the output channels, it is critical to modify the input data.

7.3.2. DLC Validation

A plausibility check for the DLC implementation is the DLC validation. You can verify a legitimate extreme load table in several ways with the use of the extreme load table. An example validation procedure is display in the figure. Several blocks will be described in the section that follow.

Diversity Check

To make sure that no output file or load case dominates the extreme load table this cheque looks at the variety of severe event. An OpenFAST numerical problem or incorrect DLC setting may be the cause of a dominant file or case.

Examine the duration of the excessive load can reveal numerical problem for instant in a shutdown load scenario the excessive load should happen close to the shutdown. Extreme load at the start of the simulation is also highly suspicious and may result from an extended initialization. At that point the DLC recorded time should be modified.

The pitch rate for manual manoeuvres and incorrect wind behaviour is two examples of incorrect DLC setting. Negative thrust caused by an excessively high pitch rate result in extremely high bending moment on the top and bottom of the tower. The rule outline wind behaviour however we did not just depend on these criteria for the 4.2 gust. As a result, we had an extremely high load due to an overconservative wind.

Comparison of different DLCs

We perform MExtremes for each DLC independently to compare them. Next to determine which load case is critical for which component we compare the severe load of a few sensors. When a load in a DLC does not meet expectations, this comparison is illogical.

8. OpenFast Error Handling

We noticed various Issues while using the OpenFast program. This session documents our effort to remedy these concerns, including the step we took. Effort to address them, as well as our recommendation for others who face similar issue.

8.1. False inputs

OpenFAST is vulnerable to misleading input. We encountered this issue frequently with several input. Investigating the shell code can reveal the root cause of the error. The application reads file sequentially and stops if an error occurs. This behaviour allows you to identify which file the false input was defined in. The error code indicates where and what to search for.

8.2. Tower Strike

The error “tower strike” occurs when using the AeroDynv15. This happen when the blade’s tip deflects significantly and collide with the tower. Deflecting the blade tip in ElastoDyn requires real degrees of freedom for the blade. When you see AeroDynv14, you must perform manual tests because the AeroDynv14 does not take tower geometry into consideration.

The reason for this issue is that the blade stiffness is too low. We tested many stiffness distributions and value to achieve realistic behaviour. Typically, the rotor blade team is responsible for implementing spar caps and shear. The next phase in designing a rotor blade is to consider webs, which are crucial for structural qualities. Typically, this requires multiple design loops and a significant amount of effort.

When considering aerodynamic, it’s important to include the ‘bow-wave’ or ‘tower-dam’ effect, as seen in the equation. Figure shows the ‘tower-dam’ effect. Therefore, the minimum tip-tower clearance should be larger or at least one tower diameter.

8.3. Working in the Post-Processing Phase

The post processing phase is of particular interest as it describes the result. Initially, challenges were experienced during the whole post processing phase due to finite resources that made a cycle of failure. Though, in a bit to address the challenge, several recommendations were made, and a transition from MATLAB version 2024 to MATLAB 2011b was made, and the results were as expected. It is therefore recommended that the MATLAB version 2011b be used. To improve efficiency in the next revision, it is recommended that the output channel order be consistent for all load instances and as many sensors as possible be defined during the initial stage.

9. Automation

```
@ECHO OFF
REM - Add FAST.exe directory to Windows search path ---------------------------------
ECHO * Actual Windows search path is:
ECHO * ------------------------------
ECHO * %PATH%
ECHO *
        set PATH=%PATH%D:\FST_V3.0_ON\DLC_7.1\RANDSEED1;
ECHO * New Windows search path including FAST directory is:
ECHO * ----------------------------------------------------
ECHO * %PATH%
ECHO *
REM - Call FAST -------------------------------------------------------------------
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_015.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_030.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_045.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_060.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_075.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_090.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_105.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_120.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_135.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_150.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_165.fst
        OpenFAST_x64_v3-0-lidarSim OPTSyria5MW_180.fst
PAUSE
EXIT /B
```

Figure 8 Automation in fst File

9.1. TurbSim

We created an automation for running TurBsim to reduce the time spent preparing wind files. For example, DLC 1.2 required 72 wind files. For this automation, we fixed the settings, such as the turbulence model according to the DLC, and define arrays with the required wind speed and random seed count. Appendix 9.1 contains a detailed code.

9.2. Single DLC

According to the instructions, we tested each load case with a specified number of wind files. Depends on the load condition, we had to change the properties. To execute all the possible instances for one DLC, we used the following automation.

9.2.1. Different wind files

This automation was important for each DLC. We followed a script from the last year's group for load and dynamic. This script executes the case, save the output file in a specific folder, improve the location of the wind file in inflow.dat, and restarts OpenFAST.

9.2.2. Yaw misalignment

DLCs with varied yaw misalignment need automation to assemble azimuth angle. To setup this function, we put a loop to the script from chapter 9.2.1 that changes the basic condition in ElastoDyn.dot. The script reads the wind files in the specified folder and manage the yaw position depend on a provided array, storing any misalignment differences.

9.2.3. Azimuth angle

The azimuth angle influences the load situation that depict shutdown scenario. Because of the load's influence on blade position, different azimuth angle must be simulated in these circumstances. To avoid turbine shutdowns at optimal times, use four alternative beginning blade location in 30-degree section due to the symmetry of our 3 bladed turbine. The script is like chapter 9.2.2, with the addition of a loop for the yay position, which refers to the azimuth angle and its associated array.

9.3. All DLCs

This script performs automation scripts for individual DLCs based on an array of routes.

9.3.1 Reading the *.out-file names

This file contains only DLC out files which need to be analyzed for mextreme calculations so we need to create a separate folder in the data file of extreme and inside that we use same nomenclature for writing on the name of out files.

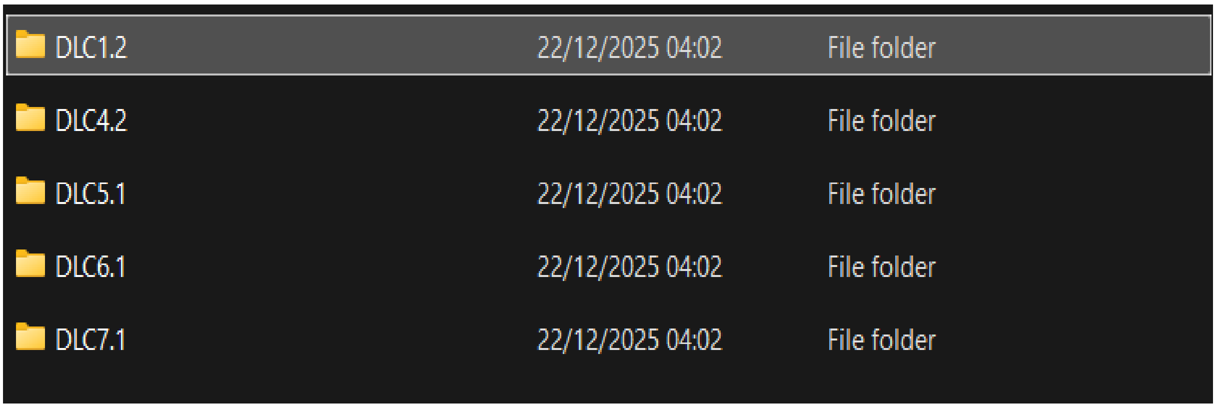


Figure 9 DLC Out File Names

9.3.2 Running all *. mext-Files

To validate and analyse the outputs, we utilised a MATLAB 2011b script to run all *. mext-file.

DLCs = {'All_DLC.mext', 'DLC12.mext', 'DLC13.mext', 'DLC14.mext', 'DLC42.mext', 'DLC51.mext', 'DLC61.mext', 'DLC71.mext'}.

For i = 1: length (DLCs)

settingsFile = DLCs{i}.

MExtremes (settingsFile).

End

10. Conclusion

We had various challenges along the procedure. We communicated clearly and effectively to resolve technical and practical issue. So for the first in Optimus series we are using Lider Roscoe as a controller which gives us an optimized result in comparison with the previous controller so from this project we a lot of things and we learn how to manage Openfast version 3.0.0.

During the phase of the project, we face a lot of technical challenges and error but apart from all we got succeed in managing all those and able to generate a complete set of simulations for the analysis and for that we would like to our professor Mr. Manjock. The technical challenges were solved using three pillars: Try an error, a website for managing OpenFast, and of course, Mr Manjock's assistance. To obtain expertise and understanding about OpenFast, try and error is the most successful method. However, there are various helpful resources for fundamental difficulties and general information. Mr Manjock's expertise with OpenFast and the Optimus Project provide valuable insight throughout the project.

References

[1] STANDARD DNV GL AS Loads and site conditions for wind turbines, 2016.

[2] M. L. B. J. J. M. Jonkman, FAST User's Guide, National Renewable Energy Laboratory (NREL), 2015.

[3] "Google Maps," [Online]. Available: https://www.google.com/maps.

[4] "The MathWorks MATLAB version: 7.13.0.564 (R2011b)," 2011. [Online]. Available: https://www.mathworks.com.

[5] http://github.com/OpenFAST/openfast/releases/tag/v3.0.0