

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx
Revision	B
Author	Dean Steinke
Co-authors	DSA: Ryan Nicoll, André Roy, AJ Baron UPEI: Matthew Hall
Status	Draft
Prepared for	OERA
Client reference / project	n/a
DSA project	OERA-FAST
Last revised	2018-07-10
Comments / Notes	Project Title: Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS Project Number: Project Start Date: May 1, 2018 Project End Date: Oct 30, 2018 Recipient Name: Dynamic Systems Analysis Ltd. Project Lead: Dean Steinke Report submission date: Oct 1, 2018
Pages (incl. cover)	16



DSA Pacific Office
101 – 19 Dallas Road
Victoria, BC V8V 5A6
+1.250.483.7207

DSA Atlantic office
210 – 3600 Kempt Road
Halifax, NS B3K 4X8
+1.902.407.3722

info@dsa-ltd.ca
www.dsa-ltd.ca

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

Revision history

Revision	Revision description	Date	Summary of changes / additions	Revisions by	Checked by	Issued to
A	Issue for review; external	2018-10-05	Report release to OERA	DS	n/a	External, Rodrigo Menafra, OERA
B	Issue for review; External	2018-12-10	Report release with incorporation of reviewer's comments	DS	n/a	Gregory Decker

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

Executive Summary

Research has been undertaken to couple ProteusDS with the National Renewable Energy Lab's (NREL) wind turbine code FAST. FAST is an open-source software package developed by the NREL. Currently, Dr. Robynne Murray of NREL (formerly Dalhousie) is modifying FAST to provide the capability to model tidal turbines through inclusion of buoyancy and added mass effects. DSA's ProteusDS software excels at numerical modelling of tidal passages, floating tidal platforms, cables, and mooring systems. DSA worked with Dr. Matt Hall from the University of PEI who has experience connecting mooring models with the FAST software to developing a coupling methodology and to begin the software code development that will enable DSA to leverage the advanced turbine modeling capabilities of FAST.

The coupling approach that was determined in the project is to embed AeroDyn as a library (DLL) in ProteusDS. ProteusDS will call the AeroDyn DLL to provide turbine loads to ProteusDS, based on fluid domain data provided by ProteusDS.

To achieve coupling the AeroDyn driver example in OpenFAST was used. The example isolates the AeroDyn model in a form that can be run independently. A number of changes to OpenFAST had to be made to make this stand-alone code suitable for coupled use. These include:

- Setting of turbine rotor kinematics from an external source rather than pre-set values,
- Setting of stochastic water flow kinematics from an external source rather than uniform pre-set values,
- Inclusion of added mass forces and summation of added mass/inertia over the rotor,
- Output of hydrodynamic forces and added mass matrix.

After making these additions to OpenFAST, UPEI developed a wrapper library that DSA can call from ProteusDS. However, due to Cygwin compilation issues, DSA has been unable to yet fully couple the codes. However, the framework in ProteusDS was completed that allows for adding a FAST turbine to ProteusDS. A rough tutorial on the usage of the new turbine feature functionality in ProteusDS was created.

Next steps for the research will be to develop a sample case which can be downloaded from DSA's website demonstrating the new functionality. Tests to ensure that coupling is accurate will be completed. Additional work will be completed to enable ServoDyn and BeamDyn to be used in addition to AeroDyn.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

Contents

Revision history	2
Executive Summary	3
Contents	4
1 Introduction	5
1.1 Overview	5
1.2 Objectives	5
2 Discussion	6
2.1 Determination of coupling method	6
2.2 Description of coupling plan	8
2.2.1 Develop AeroDyn DLL	8
2.2.2 Develop ServoDyn coupling	8
2.2.3 Develop BeamDyn coupling	8
2.3 Code development	8
2.3.1 Solution process	10
2.4 Description of AeroDyn tests and sample simulation	11
2.5 Tutorial on new ProteusDS FAST-enabled turbine model	11
3 Deliverables	14
4 Conclusions	14
5 Recommendations & Next Steps	15
6 Project Budget	15
7 Employment summary	15

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

1 Introduction

1.1 Overview

The report reviews the work completed by DSA and UPEI on its OERA Open Call R&D project entitled *Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS*.

The primary objective of the project was to couple ProteusDS with the National Renewable Energy Lab's (NREL) wind turbine code FAST. FAST is an open-source software package originally developed to model wind turbines. Currently, Dr. Robynne Murray of NREL (formerly Dalhousie) is modifying FAST to provide the capability to model tidal turbines through inclusion of buoyancy and added mass effects. This development is in response by the needs of tidal community to model tidal turbines where these effects are important. Once you have developed the turbine model for tidal turbines, you must also model floating platforms and their moorings. DSA's ProteusDS software excels at numerical modelling of tidal passages, floating tidal platforms, cables, and mooring systems. The Blade Element Momentum Theory (BEMT) turbine model provided by FAST is required to predict the turbine loading on the floating platform given spatial variations in flow speeds and turbulent effects. These turbine loads are passed to mooring systems.

Large turbines (e.g. 5m in diameter) result in large variation in predicted thrust based on the location of a blade in the water. For example, there is a vertical shear in velocity (high current at surface, and low current at seabed), and there can be large variation in flow horizontally as well. Because of this, there is a clear requirement for BEMT based model for tidal energy turbines, as the BEMT model will capture the variation in thrust and torque load due to local water velocity differences at different span locations along a blade.

To couple a turbine model with a marine dynamic analysis package like DSA's ProteusDS requires ensuring that loads, moments, inertial loads, and gyroscopic effects are correctly accounted for. It also requires detailed knowledge of the marine dynamic analysis code, and, of the turbine model. Each of these codes will have unique assumptions, coordinate conventions, and formulations for their dynamics. It is not often trivial to couple dynamics software codes as a result.

DSA worked with Dr. Matt Hall from the University of PEI who has experience connecting mooring models with the FAST software. Their main gap in knowledge at the beginning of the project surrounded the best way to couple the software. FAST is written in FORTRAN, and contains many modules, such as the AeroDyn module which models the turbine blade loading. As such, much of the project work focused around evaluating the coupling strategy.

1.2 Objectives

The project objectives were:

1. Develop a coupling method plan between ProteusDS and FAST
2. Implement the coupling plan
3. Develop a sample simulation in ProteusDS that includes the FAST-enabled load effects

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

During the project, the coupling method and strategy was developed by UPEI and DSA. UPEI developed coupling code that enabled ProteusDS to interact with FAST through a wrapper to the AeroDyn module in FAST.

2 Discussion

2.1 Determination of coupling method

For reference, the tasks in each of the project activities are provided below as they were submitted are presented in Table 1.

To start the work, DSA surveyed the marine renewables industry that makes use of FAST to assist in modeling their turbines, including Principle Power and Scotrenewables. This information, in combination with the background provided by NREL, provided valuable feedback on the coupling methodology that DSA should proceed with.

Two options for coupling were identified:

1. **OrcaFAST approach:** Couple ProteusDS with FAST following the OrcaFlex approach, where FAST calls OrcaFlex to return mooring loads. In general, FAST handles all motions estimation, and OrcaFlex is only used for mooring loads.
2. **Embedded AeroDyn approach:** Have ProteusDS call the AeroDyn turbine module to extract turbine loads. ProteusDS will pass boundary condition information (like water velocity at node positions at points along the blade) to AeroDyn, and AeroDyn will return loads to ProteusDS due to the turbine.

Through discussions with UPEI, DSA selected the “Embedded AeroDyn approach”. This decision was made due to the following reasons:

- A mooring design process requires evaluation of the mooring strength under a range of load cases. These load cases check the mooring loads and floating platform motion in varied wind, wave, and current conditions, as well as turbine operational states. Batch processing of these load cases is crucial for industrial application of the technology. This requirement was emphasized in discussion with industrial contacts as well as DSA’s own internal consulting experience. ProteusDS has batch processing tools which make it easy to process load cases required by standards such as IEC
- ProteusDS contains many pre and postprocessing tools (like results visualization) which would not be utilized if one were to rely on FAST to model the 6 degree of freedom rigid body motions.
- Solutions using the OrcaFAST approach already exist, and based on feedback from industry, were challenging to work with¹. A more streamlined approach is needed to effectively model floating turbine platforms, their moorings and their turbines.

¹ A primary challenge is that you must run all of your analysis through FAST, and that it is cumbersome to run multiple sea-state realizations using FAST. Second, FAST relies on input files, and you must interact with OrcaFlex as a DLL – rather than use a full User-Interface.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

Table 1 Initial plan for R&D project

Activity #	Description	Who?	Methodology
Activity 1	Development of coupling method plan	UPEI, DSA	A series of online meetings will be held between Matt Hall (UPEI), Robynne Murray (NREL) and DSA to discuss the proposed coupling approaches. The coupling strategies possible will be evaluated, and a decision will be determined based on technical feasibility and the sector's requirements (e.g. ease of use, maintenance, batch processing requirements, etc.). Robynne Murray will present the advancements being made in FAST. Potential solutions to be investigated include: Use of AeroDyn, use of FAST as a DLL, and use of ProteusDS as a DLL (i.e. ProteusDS in API mode).
Activity 2	Development of coupling code between ProteusDS and FAST and mooring test case	UPEI, DSA	DSA and UPEI will jointly develop and test the coupling code. NREL will be consulted as the process progresses on an as-needed basis. An example case will be produced that can form the basis of a tutorial on how to model turbines in FAST and floating platforms in ProteusDS. The coupling success will be evaluated based on a set of sample input files. The coupling code will be written in C++, which can interact with FAST and ProteusDS or the ProteusDS API. It is likely that the coupling code will be made available for download as an API example (if applicable) from DSA's website so that tidal energy research teams can use the code for their projects.
Activity 3	Development of sample mooring system model that is affected by FAST-enabled turbine model in ProteusDS	UPEI, DSA	A sample floating tidal turbine system will be decomposed into sub-models in FAST and ProteusDS. The mooring system will be modeled in ProteusDS. The tidal turbine will be modeled in FAST. A coupled simulation will be developed for testing and verification of the coupling. A validation case will be developed to ensure that all loads are appropriately passed between the software packages, including inertial and gyroscopic effects, torque, and thrust. UPEI's experience will be used to ensure that these effects are properly transmitted to ProteusDS from FAST (and vice-versa). A comparison will be completed between the standard ProteusDS turbine model output and the FAST BEMT model output for the moored platform simulation. Initial observations on the effect of the high-fidelity BEMT model on mooring line and cable loads will be observed. A tutorial will be developed that will enable the reproduction of the results for ProteusDS users in the tidal sector in Nova Scotia. This task will be completed as the code is being developed to facilitate code testing.
Activity 4	Development of final report	DSA	A final report will be created which documents the coupling strategy and sample mooring case developed in Activity 3. Observations on the efficacy of the BEMT modelling approach will be made. Any limitations or recommendations for future work and improvements will be made in the report

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

2.2 Description of coupling plan

As part of the project, a coupling plan was developed. The method for coupling ProteusDS to FAST had multiple potential paths. Developing a solid development plan will ensure the long term success of using ProteusDS with FAST for floating turbine systems.

The coupling methodology then follows a path of drawing on specific load effects from FAST modules. This is facilitated by the ongoing modular code development structure of FAST. This means that ProteusDS would drive the simulation process and utilize force and load effects from various FAST modules compiled and linked accordingly.

A three stage approach to developing the coupling between FAST modules and ProteusDS was determined, as described in the following sections.

2.2.1 Develop AeroDyn DLL

The primary FAST module of interest is AeroDyn, which is the module responsible for assessing fluid flow induction factors and the thrust and torque on the blades of the turbines. The first stage of the implementation plan is to integrate AeroDyn within ProteusDS via a DLL interface, with ProteusDS providing information on undisturbed environment velocities, platform position, orientation, and velocity, and turbine state, with the AeroDyn module then computing the resulting reaction forces and torques at that instant in time in the time domain simulation process. The coupling will be made possible by (1) reconfiguring AeroDyn to compile into a DLL that can be called by other codes and (2) designing an interface in ProteusDS to work with the AeroDyn DLL. To begin with, the rotor will be treated as rigid and constrained to a constant rotation speed. This facilitates simulation and analysis of a moored floating platform with an operating turbine following steady-flow behaviour. It does not account for the specific turbine control system or the effects of blade flexibility.

2.2.2 Develop ServoDyn coupling

The second implementation stage is an investigation of the ServoDyn FAST module to determine how existing methods in FAST for integrating turbine controllers can be incorporated into the coupled ProteusDS-FAST simulation. Being able to use external controllers will allow for more complex behavior from the tidal turbine, including variable blade pitch and variable rotor speed operation.

2.2.3 Develop BeamDyn coupling

The third implementation stage is an investigation in the use of the BeamDyn FAST module. This is used to determine the effects of blade flexibility on thrust and torque response and may be important at higher current velocity magnitudes and in turbulence. In addition to the inputs required by AeroDyn, the platform acceleration at the blade root is required, which may introduce complexities in the software time domain solution procedure, as platform accelerations are not known a priori to force and moment computation and may require an iterative solution procedure.

2.3 Code development

UPEI worked with OpenFAST to develop a wrapper in CPP that will facilitate ProteusDS calling AeroDyn.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

Meetings between DSA, UPEI, and NREL established the general approach to be taken, using the latest version of FAST, called OpenFAST, which is under ongoing development. The tidal turbine modifications to AeroDyn, which had been made to a previous version of FAST, needed to be brought into OpenFAST.

To begin with, the UPEI team successfully set up the OpenFAST source code for compiling with Cygwin following NREL instructions. Next, the AeroDyn code was updated with the modified AeroDyn code provided by NREL. The full OpenFAST code was again compiled successfully to check the completeness of the integration.

UPEI and DSA agreed to begin by incorporating only the AeroDyn module of FAST in the model that would be coupled with ProteusDS. This process began by taking the existing AeroDyn driver example in OpenFAST, which isolates the AeroDyn model in a form that can be run independently. A number of changes had to be made to make this stand-alone code suitable for coupled use. These include

- Setting of turbine rotor kinematics from an external source rather than pre-set values,
- Setting of stochastic water flow kinematics from an external source rather than uniform pre-set values,
- Inclusion of added mass forces and summation of added mass/inertia over the rotor,
- Output of hydrodynamic forces and added mass matrix.

Because the original AeroDyn driver was designed to run independently, additional features were required to provide coupling with a larger model like ProteusDS. An interfacing capability was created that provides communication between ProteusDS and the stand-alone AeroDyn code. This interface facilitates a dynamic loose coupling at the tidal turbine's rotor hub, with ProteusDS handling the modelling of the rotor shaft rotation, while the AeroDyn-based model handles the hydrodynamic forces and moments acting upon the rotor as well as the generator torque. Flow conditions are managed by ProteusDS and passed to the AeroDyn model at each coupling time step. Because AeroDyn requires a fixed time step size, the interfacing code contains an interpolation functionality that converts from the variable time step of ProteusDS. Lastly, the interfacing code also serves as a wrapper for the FORTRAN-based AeroDyn model, making it accessible to ProteusDS, which uses C++ data types.

The interfacing code includes functions that initialize the AeroDyn simulation, pass rotor shaft rotations to AeroDyn, return blade node positions to ProteusDS, pass flow velocity data to AeroDyn, and return net rotor hydrodynamic forces and moments to ProteusDS.

Changes were required to the AeroDyn input file structure to support the coupling with ProteusDS. Essentially, the AeroDyn-based model needs information about the turbine geometry and mass characteristics. These details are specified in a customized version of the original AeroDyn driver input file. The final AeroDyn-based model then requires this customized input file as well as all the normal AeroDyn input files. This approach provides a minimum of input file changes, making the ProteusDS-AeroDyn coupling as accessible as possible to users who are familiar with both models individually.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

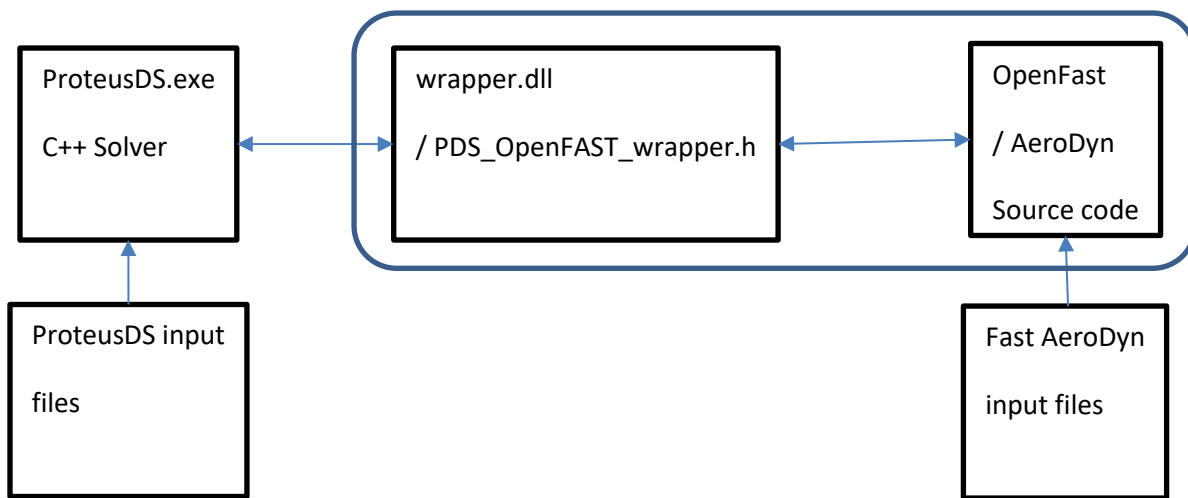


Figure 1 ProteusDS Interface with FAST

The wrapper is written in C++. The wrapper converts between ProteusDS vector data types and regular arrays.

2.3.1 Solution process

The solution process followed by ProteusDS which uses FAST is as follows:

1. User sets up ProteusDS model of floating platform, mooring system and sets environmental conditions.
2. User sets up FAST model of turbine using OpenFAST AeroDyn module input files.
3. ProteusDS inputs files and model is set to load AeroDyn input files
4. ProteusDS sets the turbine rotor rotation rate in OpenFAST; ProteusDS sets the fluid kinematics in OpenFAST.
5. Loads are evaluated in OpenFAST based on the set boundary conditions (turbine rotation rate, and fluid kinematics).
6. Turbine loads and moments and added mass and inertia of the turbine are retrieved from OpenFAST by ProteusDS
7. ProteusDS evaluates system dynamics based on all set parameters (including mooring line loads, global hydrodynamics and the retrieved loads/moments/added mass/inertia)
8. Time step advances based on explicit numerical integration and new rotation rate for turbine rotor is determined; new environmental conditions are known (based on simulation settings).
9. Repeat steps 4-8 to evaluate time-history of turbine loads and platform/mooring loads – and evolve system dynamics.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

2.4 Description of AeroDyn tests and sample simulation

A draft version of the AeroDyn-based model and interfacing code has been completed, including all of the described functionality. It has been successfully compiled and tested in isolation with steady-state input conditions. However, difficulties in compiling a portable version that does not depend on specific Cygwin libraries on the compiling computer have prevented testing of the coupling with ProteusDS. The immediate priority is to explore compiling options to solve this problem. Unfortunately, the supported compiling options for OpenFAST are very limited—purchasing a new compiler is most likely required.

The developed code is provided as a digital appendix to this report:

- PDS_AeroDyn_Coupling_2018-09-17.zip

This code has several components of note:

1. The PDS_OpenFAST_wrapper.h is the header file which contains the function definitions which ProteusDS uses to pass turbine loads to its rigid body (e.g. a floating or submerged tidal turbine)
2. Wrapper.dll is the compiled AeroDyn code which calls the AeroDyn input files. The input files for the example are located in the input directory:
 - a. ad_airfoil_example.inp
 - b. ad_blade_example.inp
 - c. ad_driver_example.ech
 - d. ad_driver_example.inp
 - e. ad_primary_example.inp
3. wrap_text.exe is the testing program written to test the functioning of the Wrapper.dll

The next steps which have been identified by DSA to succeed in coupling the code is to determine the Cygwin dependencies that are required, or to attempt to compile the wrapper using Intel's FORTRAN compiler which would remove the dependencies.

2.5 Tutorial on new ProteusDS FAST-enabled turbine model

A development branch has been created within ProteusDS code-named *Butterfish*. Code for testing the wrapper.dll has been built into this development branch.

The existing turbine user interface in ProteusDS makes use of a feature which is referenced from the \$Turbine property in the RigidBody DObject, as shown in Figure 2. This existing turbine model uses a look up table based on relatively fluid velocity at the nacelle (turbine hub). Tip speed ratios and thrust and torque coefficients are used to define the thrust and torque applied to the rigid body. Turbines can be added at any location or orientation on a rigid body.

Following the approach of the existing turbine model in ProteusDS, a FAST-based Turbine model can also be applied at any location on a RigidBody. Using the \$TurbineFAST property in ProteusDS, as shown in Figure 3, the turbine can be placed on the rigid body. A RigidBodyTurbineFAST feature is created in the library. The feature contains a property which indicates which input folder contains the FAST project files.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

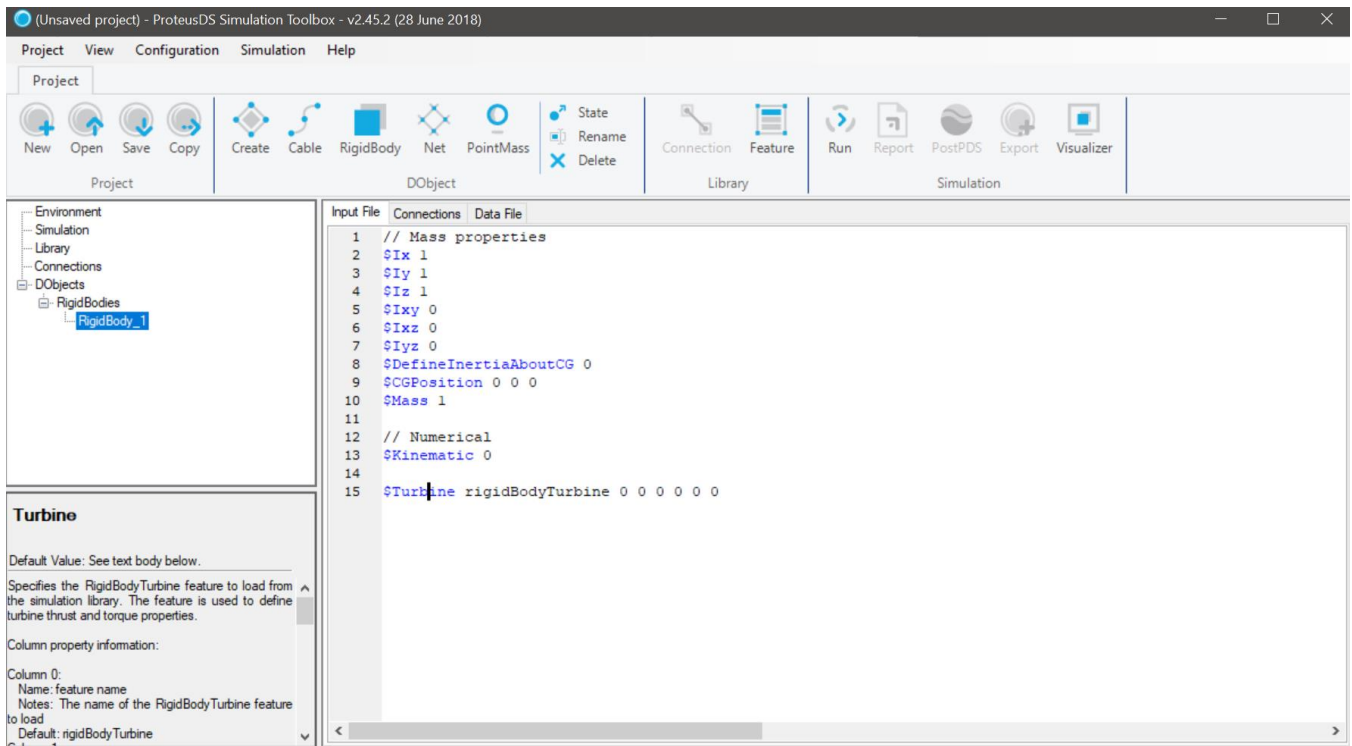


Figure 2 RigidBody input file in ProteusDS - with turbine added

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

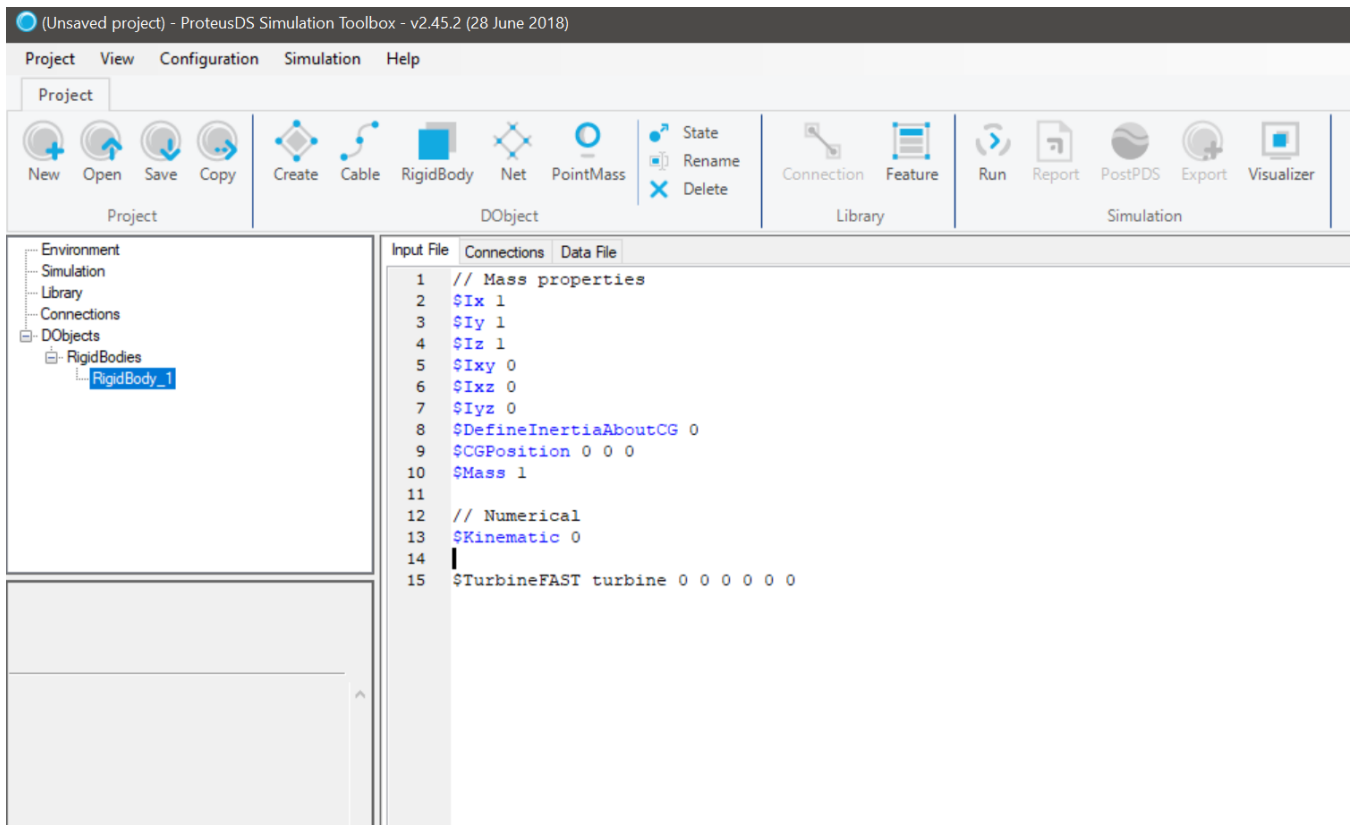


Figure 3 FAST based BEM turbine model is attached to a RigidBody in the same way as a look-up table turbine model, except the **\$TurbineFAST** property is used.

A user defines their turbine using the standard FAST input files. To use the turbine model defined by FAST input files, the FAST input files are put in a directory in the ProteusDS project folder, as shown in Figure 4. The folder in which the fast files are located, and the name of the main input file are noted in the feature within ProteusDS. This is done because the main FAST input file will reference external files.

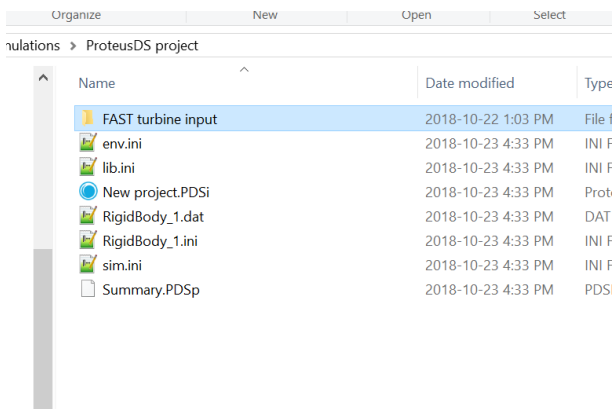


Figure 4 FAST input folder located in ProteusDS project folder

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

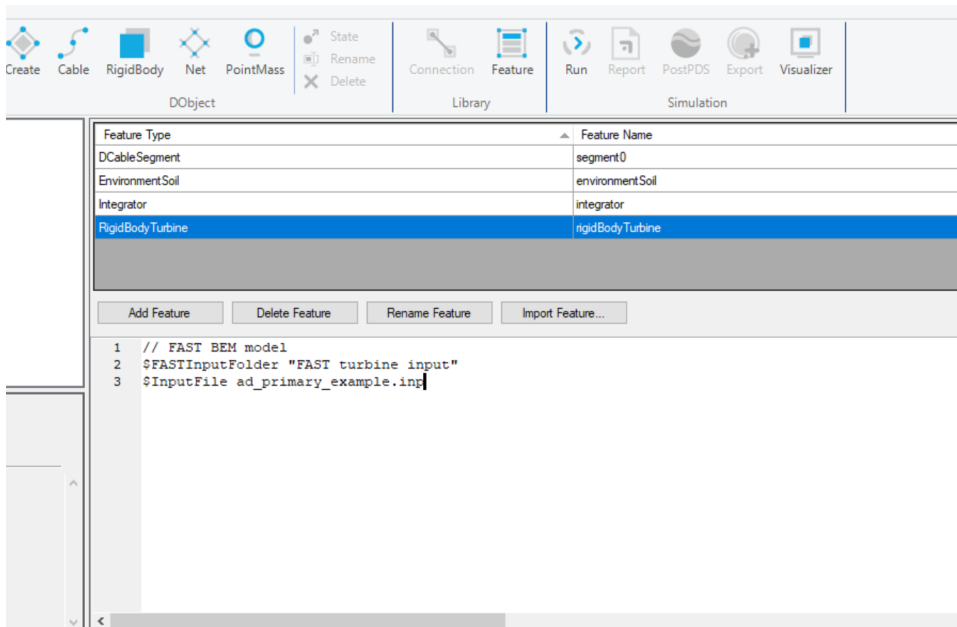


Figure 5 Input folder for FAST defined turbine input files, referenced from FAST turbine feature in ProteusDS

3 Deliverables

- The code and sample simulation have been provided as: *PDS_AeroDyn_Coupling_2018-09-17.zip*
- A tutorial on how to include the FAST turbine with ProteusDS is provided above.
- This report explains the nature of the coupling

4 Conclusions

The objectives, as listed in Section 1.2 are as follows:

Objective	Status
Develop a coupling method plan between ProteusDS and FAST	A detailed coupling plan was developed. This involves using the AeroDyn module embedded in ProteusDS, rather than having FAST call a ProteusDS DLL.
Implement the coupling plan	The code has been developed in OpenFAST to support coupling. The coupling was not fully achieved due to linking issues which require further time by DSA to address. DSA has implemented code to support AeroDyn.
Develop a sample simulation in ProteusDS that includes the FAST-enabled load effects	The sample simulation was not completed due to software library linking issues which need to be resolved first. DSA did provide an example, which shows how the simulation will work.

Title	Analysis of tidal turbine mooring systems in turbulent flows using FAST and ProteusDS - FINAL REPORT		
Revision	B	Date Last Revised	2018-07-10
DSA Project	OERA-FAST	Client Project / Reference	
DSA Document	2018-10-05-DSA-OERA-FAST-Final-Report.docx	Report Status	Draft

The work carried out by UPEI has enabled DSA to take a major step towards support for FAST within its software. The next step is to solve the compilation issues on DSA's side, potentially using the Intel FORTRAN compiler.

Future work, once compiling issues are resolved, includes testing the draft AeroDyn-based model in coupled form with ProteusDS, checking time step interpolation capabilities, and then performing an initial test run with a known scenario to allow for initial verification.

Following this, a more complete interface within ProteusDS to handle this coupling will be built.

5 Recommendations & Next Steps

Based on feedback from industry, integration approach developed appears to be robust and would be welcomed by potential ProteusDS users. DSA would also be able to better model mooring and cabling system loading using this new capability.