FAST-tutorial Documentation

Bryce Ingersoll

Aug 09, 2018

Contents

FAST is a horizontal-axis wind turbine (HAWT) dynamic analysis code developed by the National Renewable Energy Laboratory (NREL) that simulates the deflections, motions, and loading of a defined wind turbine design in specified wind conditions. FAST can be incredibly useful in the design and analysis of many aspects of the wind turbine, such as the blade, tower, nacelle, platform, and generator, to name a few.

The purpose of this tutorial is to help those unacquainted with FAST quickly become familiar with it and how it can benefit their research related to wind turbine design. This tutorial is specifically targeted for FASTv7, though some material included here may be helpful for those using other versions, such as FASTv8 and OpenFAST. In addition, this tutorial is directed toward Mac and Linux users.

CHAPTER 1

Getting Started

To begin, request an account to access and download software in the NREL/NWTC portal. The link to the page can be found here. Your account will need to be approved, but in general this should be done fairly quickly (within one or two days).

Once your account has been granted, go to the FASTv7 webpage and download *FAST Archive for Linux (tar.gz 3.1 MB)*. To compile FAST, a number of programs need to be downloaded. To see a list of these codes, unzip the downloaded FAST folder, and open the makefile in the Compiling directory. At the beginning of the folder should be a list of the other codes, which should read:

- FAST (v7.02.00d-bjj, 20-Feb-2013)
- AeroDyn (v13.00.02a-bjj, 20-Feb-2013)
- InflowWind (v1.01.00b-bjj, 10-Aug-2012)
- NWTC Subroutine Library (v1.07.00b-mlb, 10-Jan-2013)

We've already downloaded the FAST directory, but still need to download the other packages. Included are hyperlinks for the specific versions for Aerodyn, InflowWind, and the NWTC subroutine library. These versions don't match exactly with what is specified in the FAST makefile, but still successfully compile FAST.

CHAPTER 2

Compiling FAST

Once the packages need to compile FAST are downloaded, we can compile FAST using the makefile provided in the Compiling subdirectory of the FAST directory. You will first probably want to change:

```
BITS = 32
```

to:

```
BITS = 64
```

We then need to specify the location of source files for FAST, AeroDyn, InflowWind, and the NWTC Library. The easiest way to do this is to comment out:

```
ifeq ($(OS),Windows_NT)
   NWTC_LIB_DIR= C:/Users/bjonkman/Documents/DATA/DesignCodes/miscellaneous/nwtc_subs/
↪SVNdirectory/trunk/source
   AERODYN_DIR = C:/Users/bjonkman/Documents/DATA/DesignCodes/simulators/AeroDyn/
↪SVNdirectory/trunk/Source
   WIND_DIR    = C:/Users/bjonkman/Documents/DATA/DesignCodes/simulators/InflowWind/
↪SVNdirectory/trunk/Source
   FAST_DIR    = C:/Users/bjonkman/Documents/DATA/DesignCodes/simulators/FAST/
↪SVNdirectory/trunk/Source
else
   NWTC_LIB_DIR = $(HOME)/PC/CAEtools/Miscellaneous/NWTC_Library/trunk/source
   AERODYN_DIR  = $(HOME)/PC/CAEtools/simulators/AeroDyn/SVNdirectory/trunk/Source
   WIND_DIR     = $(HOME)/PC/CAEtools/simulators/InflowWind/SVNdirectory/trunk/Source
   FAST_DIR     = $(HOME)/PC/CAEtools/simulators/FAST/SVNdirectory/trunk/Source
endif
```

and replace the directory paths with the locations of the downloaded packages on your local system. For example, this might look like:

```
NWTC_LIB_DIR = /Users/username/Documents/NWTC_Lib_v1.07.00b-mlb/source
AERODYN_DIR = /Users/username/Documents/AD_v13.00.02a-bjj/Source
```

(continues on next page)

(continued from previous page)

```
WIND_DIR = /Users/username/Documents/InflowWind_v1.02.00c-bjj/Source
FAST_DIR = /Users/username/Documents/FAST_v7.02.00d-bjj/Source
```

In addition, it can be helpful to add the flag:

```
-fdefault-real-8
```

to FFLAGS. You also will probably want to change:

```
PREC = SingPrec
```

to:

```
PREC = DoubPrec
```

Finally, change the OUTPUT_NAME to whatever you want the FAST unix executable to be called, and change DEST_DIR to where you want it be created.

We're now ready to compile the FAST executable. Go to the Compiling subdirectory in the FAST directory and run:

```
make
```

This should create the FAST executable, with the name and location specified using OUTPUT_NAME and DEST_DIR.

Note: This FAST executable uses a specific pitch control routine. This pitch routine is in CertTest and called Pitch.ipt, and is specific for the WindPACT 15A1001 model wind turbine. We'll discuss later what to do for other wind turbine designs.

CHAPTER 3

Running FAST

Running FAST on the terminal is straightforward. Simply specify the path to the compiled FAST unix executable and the FAST input file (we'll discuss this file in a bit, which typically has a .fst extension). In the CertTest folder are a number of certification tests that should be run to check that everything was compiled correctly (also discussed later), but we can run one quickly now with the line:

```
./FAST_64 CertTest/Test01.fst
```

CHAPTER 4

Useful Simulation Parameters

All of the input parameters and files are discussed in detail in the FAST user manual (at this address), but we discuss briefly here some of the parameters that are initially most helpful to be familiar with.

4.1 Simulation Time

FAST performs a time based simulation, and the parameters that control in what time frame this is performed and recorded are:

- TMax
- DT
- TStart
- DecFact

The simulation time begins at 0.0 seconds, and ends at TMax. DT is the time step. At the beginning of a FAST simulation, the data is often artificially noisy. To not record this data, set TStart to the simulation time when you want to start recording data. Finally, DecFact can be set so that FAST will output data only once each DecFact integration time steps.

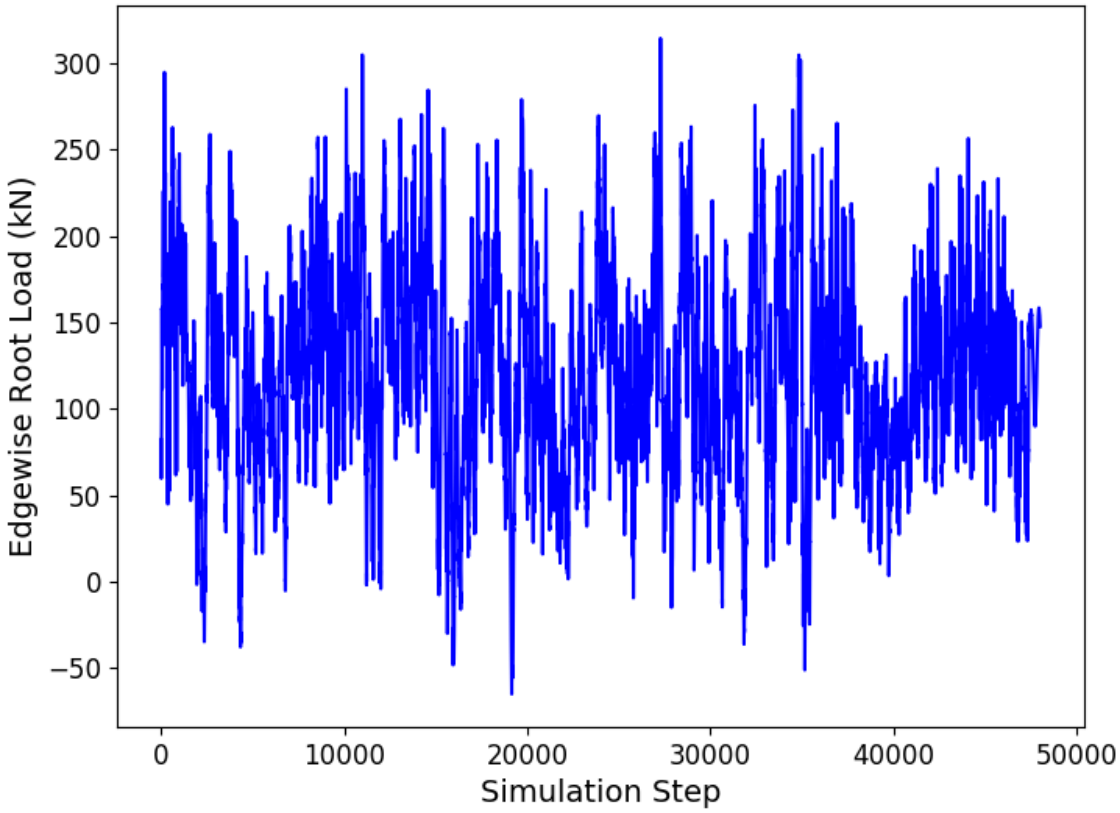
Note: You should be careful to choose an appropriate value for DT because if DT is too small or too large, the numerical solution will become unstable.

4.2 Defining the Turbine Design

Different components of the wind turbine are defined in (among others) the main FAST input file, tower file, blade file, and Aerodyn file. It would be quite the task to generate your own FAST input files, but fortunately there are some reference designs available. Consider using the NREL5MW or the WindPACT reference turbines in your studies, as well as the sample input files included in the FAST directory.

4.3 Defining Outputs

There are a number of outputs that can be specified and recorded. The desired outputs are listed at the end of the FAST .fst file. For example, we can specify that the force at the root in the edgewise direction be recorded, and a plot of this information is shown below.



Detailed descriptions of all possible outputs are given in the FAST user manual, but we briefly describe some useful tips to specify the outputs of the simulation. An example of an output is:

```
Spn2MLxb1
```

This is the local edgewise moment of the first blade (1 at end of parameter name) at the second span station (2 in middle of name). To specify the locations of the span positions where we want to record data, we use the parameter:

```
BldGagNd
```

In conjuction with:

```
RNodes
```

These two parameters determine the locations of the virtual strain gages where loading and deflection data is recorded. BldGagNd is a parameter in the main FAST input file. It is a group of indices, where the values correspond to the locations listed in RNodes, a parameter in the Aerodyn input file. For example, if we set the following:

```
BldGagNd - 1,3
RNodes - 3,8,13,18
```

Then loading and aerodynamic will be available at 3m and 13m along the length of the blade. As an example, if we wanted to record the flapwise moment at 13m along the length of the first blade, we would include:

```
Spn2MLyb1
```

in the output section of the main FAST input file.

The size of the BldGagNd array is limited by NBlGages, which is also a parameter in the main FAST input file. NBlGages is simply the length of the array. However, in version 7.0 of FAST, NBlGages cannot be greater than 7.

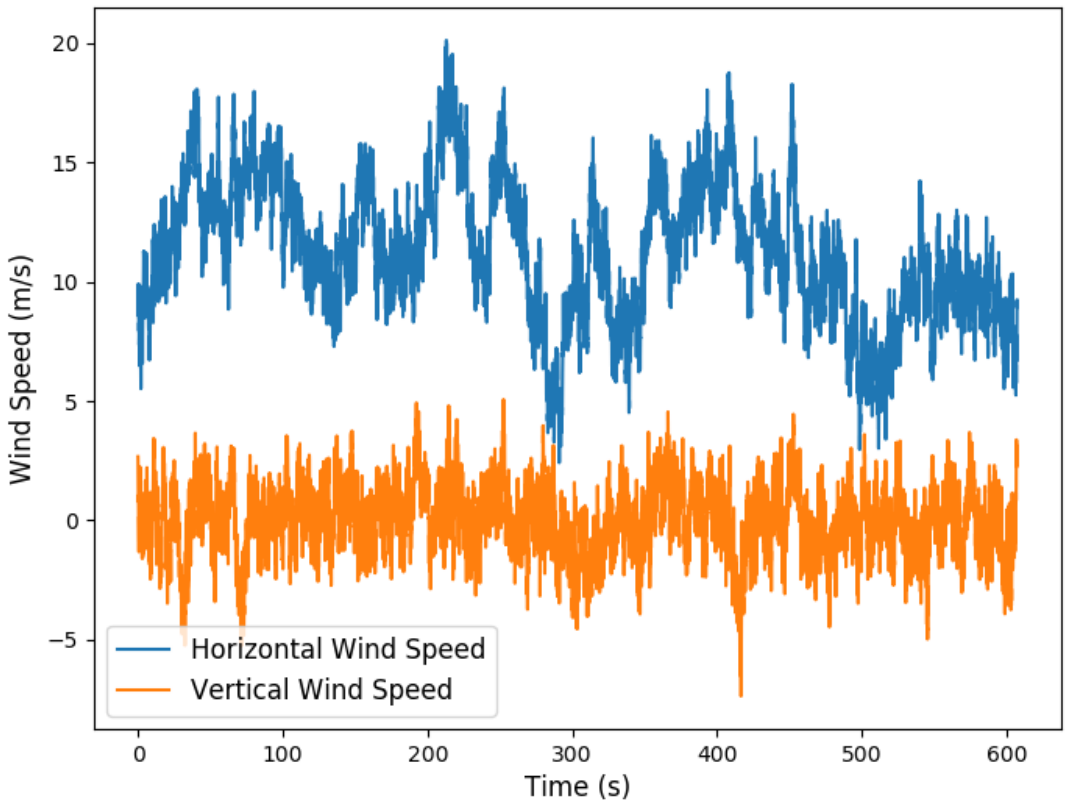
Note: BldGagNd is one-based, not zero-based, with respect to the location

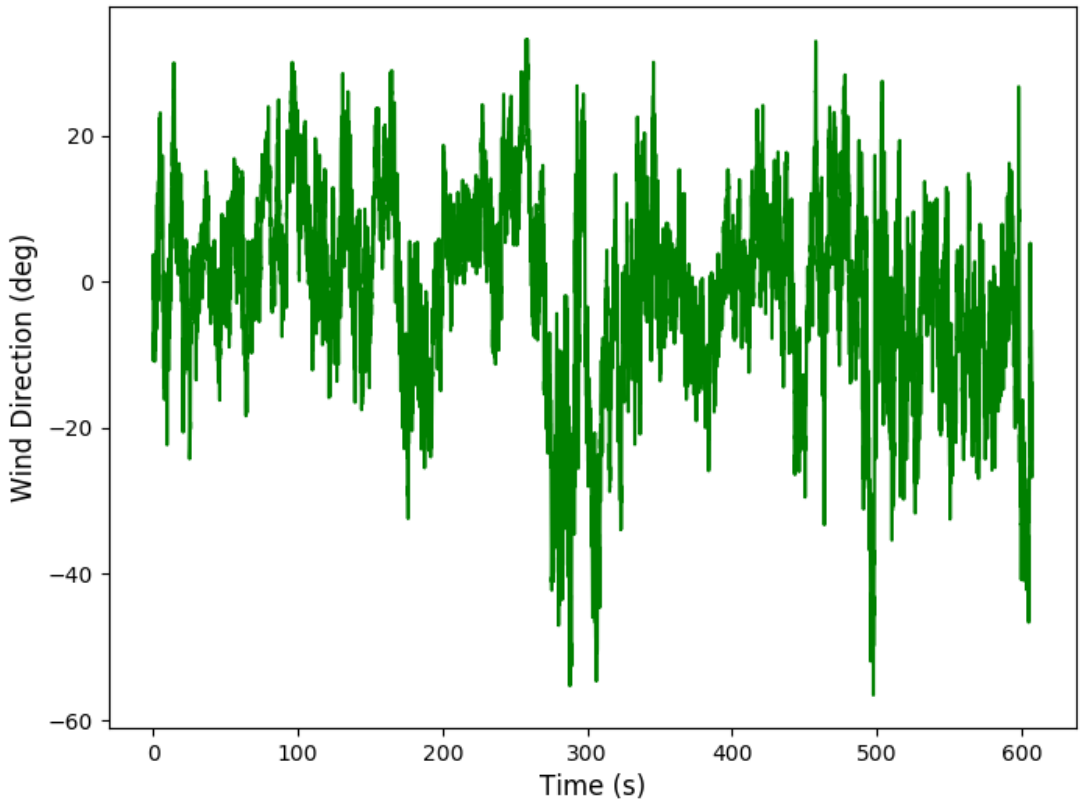
---

values in RNodes.

4.4 Wind Files

We also should specify the wind conditions of the simulation. Examples of possible turbulent conditions are shown below.





Turbulent wind input files can be generated using NREL's tool TurbSim, and non-turbulent wind input files can be generated using NREL's tool IECWind.

4.5 Next Steps

This tutorial will hopefully help you be able to do some initial turbine design analysis. For use in design optimization, please see our other tutorial.

CHAPTER 5

Indices and tables

- genindex
- modindex
- search