

# FAST - HAWC2 Model Conversion MATLAB Toolbox – Manual

*By Knud A. Kragh and Paul Fleming*

*Risø-DTU and NREL – July 2011*

modified by Bonnie Jonkman, April 2013

## Introduction

The FAST - HAWC2 Model Conversion MATLAB Toolbox is a modular toolbox for Importing FAST and Hawc2 models (including structural and aerodynamic data) into MATLAB, manipulating the model data from within MATLAB and exporting either Hawc2 or FAST models. This manual contains a short description of each of the functions included in the toolbox and charts explaining how to use the different functions in the context of converting and manipulating models. The actual conversion of FAST data to Hawc2 data is not covered in this manual, but can be seen in data conversion functions.

Import of HAWC2 models is at this point not part of the toolbox.

## Functions

The functions in the toolbox are categorized in the following categories: FAST2MATLAB, MATLAB2FAST, MATLAB2HAWC2, and utilities. In the following table a short description of each of the functions is given. For full details on the function see the function headers and the two demos.

All functions return a similar structure containing at least a list of parameters and labels. When a table or list is included in the input data file, this is a separate component.

### FAST2MATLAB

<i>Fast2Matlab.m</i>	Function that imports parameters from .fst FAST input file and stores values and parameter names in a structure. It can also import data from tower data files and blade data files, populating appropriate structures.	<b>.Label and .Val</b> , list of parameters and associated values <b>.HdrLines</b> , array of header lines <b>.OutList and .OutListComments</b> , list of variables to output and associated comments <b>.TowProp and .TowPropHdr</b> , table of tower properties and associated headers <b>.BldProp and .BldPropHdr</b> , table of blade properties and associated headers
<i>FastLin2Matlab.m</i>	Function that imports parameters from FAST linearization file and stores values and parameter names in a structure	<b>.Label and .Val</b> , list of parameters and associated values
<i>AeroDyn2Matlab.m</i>	Function for reading AD inputs into a	<b>.Label and .Val</b> , list of parameters and

		associated values <b>.FoilNm</b> , list of associated airfoil names <b>.BldNodes</b> and <b>.PrnElm</b> , the node table, with PrnElm separate
<i>Fast_PC2Matlab.m</i>	Function for importing profile coefficients from the FAST profile coefficient file. Stores data for all foils as well as average values of the profile coefficients.	Structure of airfoil data

### **MATLAB2FAST**

<i>Matlab2FAST.m</i>	Function for creating a FAST model from FAST parameter stored in MATLAB structure and a Template file. The function will replace parameters in the template file with parameters in the MATLAB structure. The MATLAB structure can be created using <i>Fast2Matlab.m</i> and manipulated with <i>get</i> and <i>set</i> utilities.
<i>Matlab2FASTLin.m</i>	Same as above, just for the linearization file
<i>MATLAB2AD.m</i>	Same as above, just for AD file
<i>GenConstWindFile.m</i>	This generates a new constant speed wind file (useful for wind speed sweeps), given a wind speed and a base file name (the wind speed is appended as part of the file name)
<i>GenStepWindFile.m</i>	Given a filename, and a list of wind speeds, generates a wind file where the wind speed is changed according to the input array every 100s

### **Suggestions for additional functions:**

- Function for exporting profile coefficient files. These should be made to enable Hawc2 to FAST model conversion.

### **Utilities**

<i>GetFastPar.m</i>	Function for getting FAST parameter value from FAST parameter structure (works on the .Label and .Val of all "FAST" structures (returned from .fst, .ipt or tower or blade))
<i>SetFastPar.m</i>	Function for setting FAST parameter value in FAST parameter structure (works on the .Label and .Val of all "FAST" structures (returned from .fst, .ipt or tower or blade))

### **Suggestions for additional functions:**

- Tool for plotting modes spaces for result files from linearization

### **Linearization**

The linearization tools use the above mentioned tools to create linearization sweeps. The scripts use the above tools to generate wind files, AD files, .fst files and call FAST, linearize and produce sweep plots of modal

frequencies changing with wind speed. The scripts can be used as is, or as a basis for creating sweeps across any arbitrary variable

<i>LinSweep.m</i>	User provides the input model files (FST, AD, Lin), a wind speed list, and some basic details (rated wind speed, fine blade pitch) and the sweep generates a .Lin file (linearized periodic model) using FAST's linearization tool at each wind speed specified
<i>SweepPlot.m</i>	Using the results of the above script (saved variables and outputted files), run Bir's MBC3 on each linearization and plot the mode frequencies vs. wind speed

### MATLAB2HAWC2

<i>Fast_st2Hawc2_st.m</i>	Function for converting structural data from FAST to Hawc2 format (blade and tower) and for defining dummy data for shaft and hub. Apart from data extracted from FAST this function requires additional input supplied in a data structure. See section ??
<i>st2Hawc2.m</i>	Function for writing Hawc2 .st file from structural data in Hawc2 format stored in MATLAB structure
<i>Aero2Hawc2.m</i>	Function for writing HAWC2 AE and PC files from Aerodyne data extracted from FAST using <i>AeroDyn2Matlab.m</i> , <i>Fast_PC2Matlab.m</i> and <i>Fast2Matlab.m</i>
<i>SimPar2Hawc2.m</i>	Function for writing simulation block in HAWC2 .htc input file
<i>Structure2Hawc2.m</i>	Script for writing structure block in HAWC2 htc file from FAST data stored in MATLAB. Uses following functions: <i>DefTower.m</i> , <i>DefShaft.m</i> , <i>DefHub.m</i> , <i>DefBlade.m</i> , <i>DefOrient.m</i> , <i>DefConstraint.m</i>
<i>Fast_Ae2Hawc2_Ae.m</i>	Function for converting Fast format aerodynamic inputs and to Hawc2 aerodynamic input format.
<i>DefTower.m</i>	Function for defining and writing tower main body in Hawc2 .htc input file
<i>DefShaft.m</i>	Function for defining and writing shaft main body in Hawc2 .htc input file
<i>DefHub.m</i>	Function for defining and writing hub main body in Hawc2 .htc input file
<i>DefBlade.m</i>	Function for defining and writing blade main body in Hawc2 .htc input file
<i>DefOrient.m</i>	Function for defining and writing orientation block in Hawc2 .htc input file
<i>DefConstraint.m</i>	Function for defining and writing constraint block in Hawc2 .htc input file
<i>AeroBlock2Hawc2.m</i>	Function for defining and writing aerodynamic block in Hawc2 .htc input file

### Parameter structure needed for FAST2Hawc2 conversion

The additional parameters needed are given below and the data structure should be structured and named as below (see example):

Hawc2Set.Tower.nbodies	- Discretization of tower main body, no. of elements
Hawc2Set.Tower.E	- Youngs moduls, used in estimation of inertia from stiffness given by FAST.

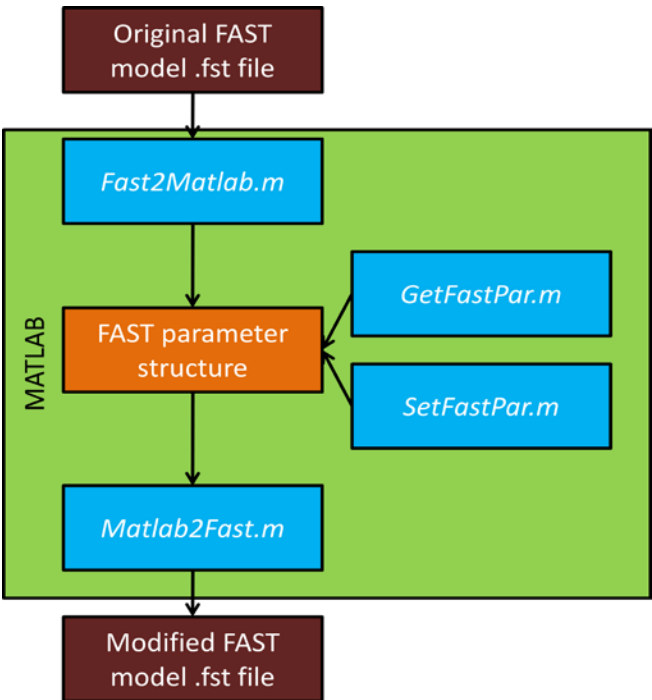
Hawc2Set.Tower.rho	- Density of tower, used for estimating cross sectional area from mass distribution
Hawc2Set.Tower.G	- Shear Modulus
Hawc2Set.Tower.J	- Torsional stiffness constant , dummy if torsional stiffness is provided in tower file
Hawc2Set.Tower.kx	- Shear factor for force in principal bending xe direction
Hawc2Set.Tower.ky	- Shear factor for force in principal bending ye direction
Hawc2Set.Shaft.nbodies	- Discretization of shaft main body, no. of elements
Hawc2Set.Shaft.nSec	- Number of sections in definition of half chord position
Hawc2Set.Shaft.E	- Youngs moduls
Hawc2Set.Shaft.G	- Shear Modulus
Hawc2Set.Shaft.Ix	- Area moment of inertia with respect to principal bending xe axis
Hawc2Set.Shaft.Iy	- Area moment of inertia with respect to principal bending ye axis
Hawc2Set.Shaft.A	- Cross sectional area
Hawc2Set.Shaft.kx	- Shear factor for force in principal bending xe direction
Hawc2Set.Shaft.ky	- Shear factor for force in principal bending ye direction
Hawc2Set.Hub.nbodies	- Discretization of hub main body, no. of elements
Hawc2Set.Hub.nSec	- Number of sections in definition of half chord position
Hawc2Set.Blade.nbodies	- Discretization of blade main body, no. of elements
Hawc2Set.Blade.E	- Youngs moduls, used in estimation of inertia from stiffness given by FAST.
Hawc2Set.Blade.rho	- Density of tower, used for estimating cross sectional area from mass distribution
Hawc2Set.Blade.G	- Shear Modulus
Hawc2Set.Blade.J	- Torsional stiffness constant , dummy if torsional stiffness is provided in tower file
Hawc2Set.Blade.kx	- Shear factor for force in principal bending xe direction
Hawc2Set.Blade.ky	- Shear factor for force in principal bending ye direction
Hawc2Set.Aero.nsec	- Number of aerodynamic calculation points

### *Using tool box*

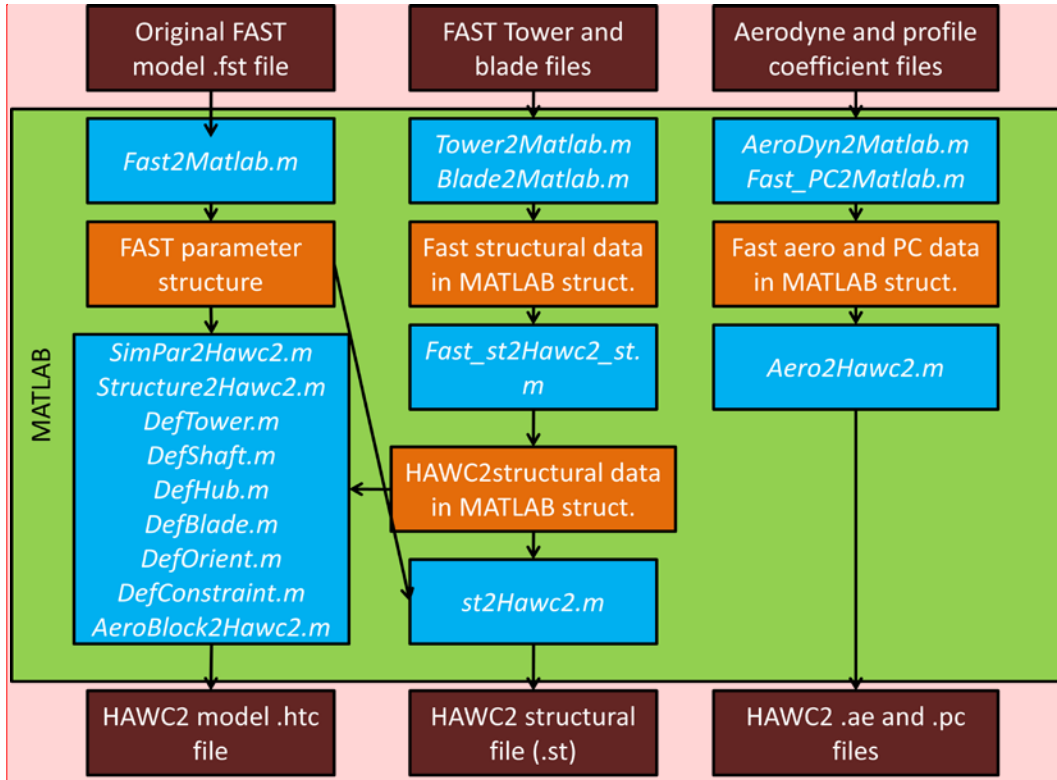
Demonstration scripts of how to use the tool box is supplied with the tool box (see the demos folder). This is the best way to gain a quick understanding of how can use the supplied tools.

Included in the demos are examples of manipulating a FAST model using the toolbox and one for converting a FAST model to a Hawc2 model. The context in which the toolbox functions are used to perform these task are illustrated in the figures below.

**Manipulating FAST model using toolbox**



**Converting FAST model using toolbox**



**Comment [BJJ1]:** Tower2Matlab.m and Blade2Matlab.m have now been replaced with Fast2Matlab.m