Author(s): Lars Christian Henriksen, Knud Abildgaard Kragh and Carlo Tibaldi
Title: The linear Aero-Servo-Elastic code HAWCStab2: User Manual
Institute: Department of Wind Energy

Summary (max. 2000 char.):
This report is a user manual for HAWCStab2.

Report Number: DTU Wind Energy E-0022
Publication Date: January 2013

Contract Number: -

Project Number: 46-44015-I-1

Sponsorship: EU project STABCON, EUDP project AeroOpt, and DSF project CASED

ISBN: 978-87-92896-32-2

Cover: -

Pages:
Tables:
References: 2

Technical University of Denmark
DTU Wind Energy
Frederiksborgvej 399
4000 Roskilde
Denmark
Telephone +45 4677 5085

1. Preface

This report is meant as a user manual to HAWCStab2. HAWCStab2 was originally developed by Morten Hartvig Hansen. HAWCStab2 is a frequency based aero-servo-elastic code useful for e.g. stability analysis of wind turbines. It shares some code with the time based aero-servo-elastic code HAWC2 and is to some extent able to read the same input files as HAWC2. HAWCStab2 is available in two versions: HAWCStab2, which is graphical user interface based program, and HAWC2S, which is a command line based program suitable for e.g. optimization. HAWCStab2 is so far, only able to handle 3 bladed wind turbines.

Additionally, HS2pid, which is another command line program, is available with reduced functionality. HS2pid is only able to calculate tuning parameters for a PI controller assuming torsional stiff blades.

Contents

1. Preface **3**

2. Introduction **5**

3. The htc file **6**
3.1. HAWC2 specific commands 6
3.2. HAWCStab2 specific commands 7
3.3. Damping 10

4. Getting started **11**
4.1. Examples with the GUI: HAWCStab2.exe 11
4.1.1. Calculating operational points 11
4.1.2. Calculating steady state and induction 12
4.1.3. Performing open-loop aeroelastic modal analysis 13
4.1.4. Tuning of PI controller 14
4.1.5. Performing closed-loop aeroelastic modal analysis 15
4.2. Examples with the command line program: HAWC2S.exe 16
4.3. Examples with the command line program: HS2pid.exe 18

5. Keyboard shortcuts **19**

A. GnuPlot files **20**

B. HAWCStab2 Changelog **23**

2. Introduction

In Chap. 3 the basic structure of the htc file is explained.

In Chap. 4 a few examples on how to use the program are shown.

The examples shown in this document are based on:

- HAWC2 (version 11.5)
- HAWCStab2 (version 2.7)
- DTU 10MW RWT (version 2.0)

3. The htc file

The input to HS2 is htc-files, which are also used by H2. The file used by HS2 has the normal H2 specific commands as well as some HS2 specific commands.

3.1. HAWC2 specific commands

The HAWC2 user manual[1] should be consulted a more detailed description of the commands.

The **new_htc_structure** block defines the structural setup of the wind turbine. Herein, it defines the various main bodies e.g. *tower*, *towertop*, *shaft*, *hub* and *blade* in the **main_body** sub block. The main orientation of the main bodies is then defined in then **orientation** sub block. The interconnection of the main bodies is defined in the **constraints** sub block.

```
begin new_htc_structure;
  begin main_body;
  ...
  end main_body;
;
  begin orientation;
  ...
  end orientation;
;
  begin constraint;
  ...
  end constraint;
end new_htc_structure;
```

For a standstill modal analysis calculated by HAWC2, the following lines can be added to the **new_htc_structure** block.

```
  body_eigenanalysis_file_name ./eigenfrq/body_eigen.dat ;
  structure_eigenanalysis_file_name ./eigenfrq/strc_eigen.dat ;
```

where the *nbodies* option in all the **main_body** blocks should be set to 1, otherwise the body-specific eigenvalue analysis is performed on sub-bodies of the main body.

The **wind** block contains information about *density* of air, which is used by HS2.

```
begin wind ;
  density                 1.225 ;
  wsp                     11   ;
  tint                    0.201454545454545   ;
  horizontal_input        1     ;            0=false, 1=true
  windfield_rotations     0 0.0 0.0 ;    yaw, tilt, rotation
  center_pos0             0.0 0.0 -119.00 ; hub height
  shear_format            3 0.2 ;
  turb_format             0     ;  0=none, 1=mann,2=flex
  tower_shadow_method     0    ;  0=none, 1=potential flow, 2=jet
end wind;
```

The **aero** block contains information about aerodynamic properties for the blade such drag and lift coefficients. Furthermore, *induction_method* and *tiploss_method* are used by HS2.

```
begin aero ;
  nblades  3;
  hub_vec shaft -3 ;
  link 1 mbdy_c2_def blade1;
  link 2 mbdy_c2_def blade2;
  link 3 mbdy_c2_def blade3;
  ae_filename        ./data/DTU_10MW_RWT_ae.dat ;
  pc_filename        ./data/DTU_10MW_RWT_pc.dat ;
  induction_method   1 ;     0=none, 1=normal
  aerocalc_method    1 ;     0=ingen aerodynamic, 1=med aerodynamic
  aerosections       50 ;
  ae_sets            1 1 1;
  tiploss_method     1 ;     0=none, 1=prandtl
  dynstall_method    2 ;     0=none, 1=stig oye method,2=mhh method
end aero ;
```

Other blocks such as e.g. **simulation**, **aerodrag**,**force**, **hydro**, **soil**, **dll** and outputs are not used by HS2.

3.2. HAWCStab2 specific commands

HS2 also uses a HS2 specific block **hawcstab2** wherein the main bodies in the

ground_fixed_substructure

sub block are used for the tower. The main bodies in the

rotating_axissym_substructure

sub block are used as the shaft. The main bodies in the

rotating_threebladed_substructure

sub block are used for the blade where the aerodynamic forces are assumed to be applied on the last main body.

```
begin hawcstab2 ;
  begin ground_fixed_substructure ;
    main_body tower ;
    main_body towertop ;
  end ground_fixed_substructure ;
  begin rotating_axissym_substructure ;
    main_body shaft ;
  end rotating_axissym_substructure ;
  begin rotating_threebladed_substructure ;
    main_body hub1 ;
    main_body blade1 ;
    second_order_actuator pitch1 100.0 0.7 ;
  end rotating_threebladed_substructure ;
  operational_data_filename ./operational_data.opt ;
  begin operational_data ;
    ...
  end operational_data ;
  begin controller_tuning ;
    ...
  end controller_tuning ;
  begin controller ;
    ...
  end controller ;
end hawcstab2 ;
```

The **operational_data** block is optional. It contains information, which is used in Sec. 4.1.1

The **controller_tuning** block is optional. It contains information, which is used in Sec. 4.1.4

The *operational_data_filename* file contains information about the operational points for selected wind speeds.

```
22 Wind speed [m/s]          Pitch [deg]      Rot. speed [rpm]        Aero power [kW]      Aero thrust [kN]
           4.000000             2.889748              6.000000             287.319260            224.286816
           5.000000             2.115800              6.000000             805.573745            352.209828
           6.000000             1.109058              6.000000            1543.002742            500.388658
           7.000000             0.000048              6.000000            2525.245528            658.232557
           8.000000             0.000055              6.424607            3770.277010            816.795864
           9.000000             0.000019              7.226938            5374.530562           1034.430140
          10.000000             0.000056              8.031337            7378.855371           1277.059791
          11.000000             0.000048              8.839966            9826.489718           1544.121977
          12.000000             4.807932              9.600000           10636.875545           1262.557036
          13.000000             7.388350              9.600000           10640.312112           1080.883000
          14.000000             9.289680              9.600000           10634.865878            970.253169
          15.000000            10.887191              9.600000           10652.538640            892.739336
          16.000000            12.346992              9.600000           10618.567809            828.619150
          17.000000            13.672693              9.600000           10631.933899            780.318935
          18.000000            14.926127              9.600000           10646.833268            740.590975
          19.000000            16.120324              9.600000           10640.981055            706.090035
          20.000000            17.268079              9.600000           10646.834596            677.263923
          21.000000            18.374175              9.600000           10632.078861            651.469908
          22.000000            19.443877              9.600000           10648.917163            630.689690
          23.000000            20.484638              9.600000           10622.019335            610.636055
          24.000000            21.496839              9.600000           10628.972396            594.608862
          25.000000            22.485124              9.600000           10638.692060            580.889756
```

3.3. Damping

If *log_decrements* is present in either:

ground_fixed_substructure,

rotating_axissym_substructure or

rotating_threebladed_substructure

then the H2 specific damping commands will be overwritten by this command and a spectral damping model will be used to calculate the damping properties.

Otherwise, a damping model similar to H2 (Rayleigh) will be used. The damping properties will be calculated for the unloaded, standstill wind turbine. N.B. use only stiffness proportional terms. If mass proportional terms are used, the damping for H2 and HS2 will not be the same.

Consult Hansen [2] for more information about the mixed mass/stiffness damping model.

4. Getting started

In this chapter a few examples on how to use the program are shown.

4.1. Examples with the GUI: HAWCStab2.exe

In this section a small example on how to use HS2 is shown.

Assuming that no prior calculations are performed, the first thing to calculate is operational points for different wind speeds. When opening the desired htc file under

`File->Open HAWC2 model file...`

HS2 will produce an error because the *operational_data_filename* file does not exist. This should be ignored by pressing ok on the error dialog box.

4.1.1. Calculating operational points

The first step is to create the *operational_data_filename*. This is done under

`Computation->Optimal operational data`

A dialogue box will appear where the user is required to fill various information. If the htc file contains the following

```
begin operational_data ;
  windspeed 4.0 25.0 22 ; cut-in [m/s], cut-out [m/s], points [-]
  genspeed 300.0 480.0 ; gen. speed. min. [rpm], gen. speed. max. [rpm]
  gearratio 50.0 ; [-]
  minpitch 0.0 ; [deg.]
  opt_lambda 7.5 ; [-]
  maxpow 10638.3 ; [kW]
  prvs_turbine 1 ; [-]
  include_torsiondeform 1 ; [-]
end operational_data ;
```

then the default values in the dialogue box are replaced by the values given by the htc file.

Once the computations have been performed the user should save the computed data. This is done under

`File->Save optimal power data`

The saved data file should be named to match the file name specified by *operational_data_filename*.

The Gnuplot code found in Listings A.1 has been used to generate Fig. 4.1.

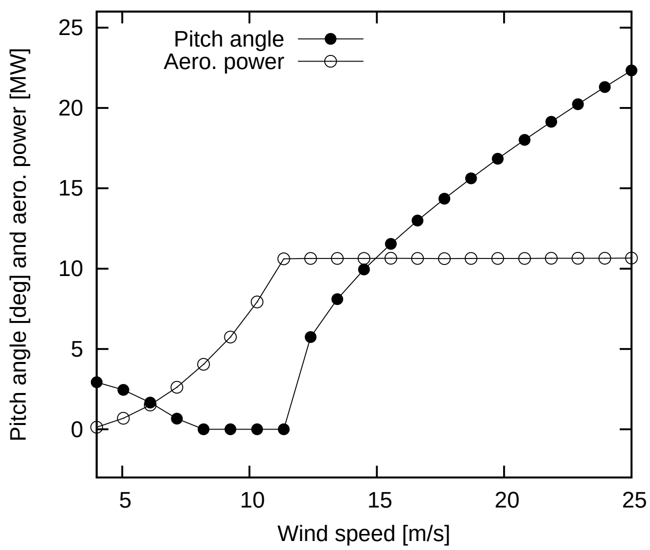

Figure 4.1.: Steady state power and pitch angle values.

4.1.2. Calculating steady state and induction

First ensure that the steps found in Sec. 4.1.1 have been performed. Then

`Compute->Steady state and induction`

should be chosen. Afterwards further analysis can be performed.

Using

`File->Save power...`

to produce *def.pwr* provides steady state value for power, pitch angle, blade tip deflections etc. The Gnuplot code found in Listings A.2 has been used to generate Fig. 4.2, where flapwise and edgewise tip deflections are shown.

Using

`File->Save steady state...`

to produce multiple files *opt_u*.ind*, preferably in a dedicated folder, for various wind speeds provide an extended number of steady state values. The Gnuplot code found in Listings A.3 has been used to generate Fig. 4.3, where the torsion of the blade along the blade span for various wind speed is seen. Steady state pitch values has been added to the total torsion of the blade to get the shown plots.

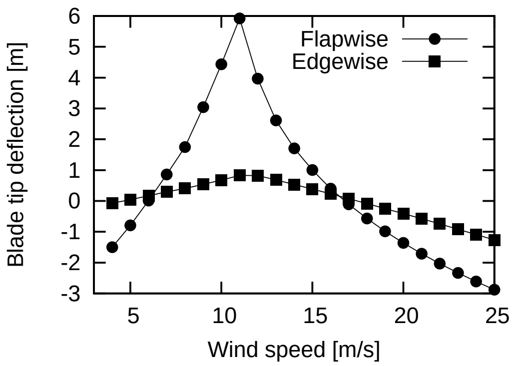

Figure 4.2.: Steady state blade tip deflections.

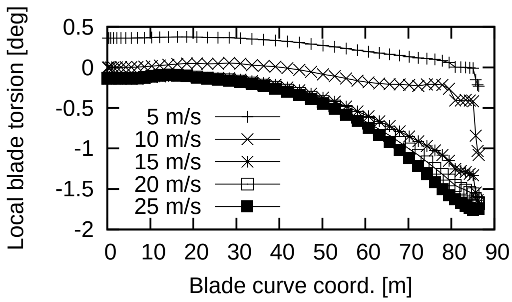

Figure 4.3.: Steady state blade torsion.

4.1.3. Performing open-loop aeroelastic modal analysis

First, ensure that the steps found in Sec. 4.1.2 have been performed.

Selecting

`Compute->Structural modal analysis->Entire turbine`

will compute the structural modes. This calculation is required to perform the

`Compute->Aeroelastic modal analysis->Entire turbine`

The sort the modes, the following values was used: (0.01,0.30,0.50Hz,0.1,8,sort after mode shapes)

Results obtained from the analysis can be saved under

`File->Save modal amplitudes`

as e.g. *turbine_ae.cmb*.

The Gnuplot code found in Listings A.4 and A.5 has been used to generate Fig. 4.4.

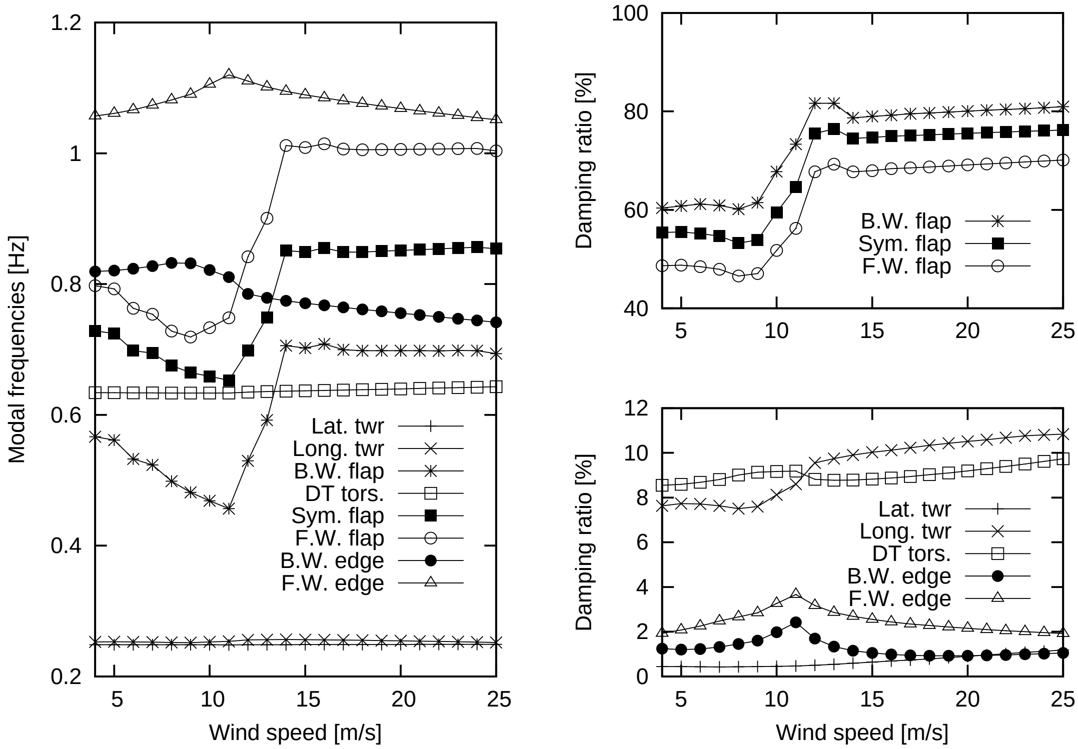

Figure 4.4.: Open-loop modal frequencies and damping ratios.

4.1.4. Tuning of PI controller

Selecting

`Compute->Tune pitch controller by DTU Wind Energy`

A dialogue box will appear where the user is required to fill various information. If the htc file contains the following

```
begin controller_tuning ;
  partial_load 0.05 0.7; fn [hz], zeta [-]
  full_load 0.06 0.7 ; fn [hz], zeta [-]
  gain_scheduling 1 ; 1 linear, 2 quadratic
  constant_power 1 ; 0 constant torque, 1 constant power
end controller_tuning ;
```

then the default values in the dialogue box are replaced by the values given by the htc file.

The computations produces *controller_input.txt*, which can be used with the Basic DTU Wind Energy controller [3].

4.1.5. Performing closed-loop aeroelastic modal analysis

To perform a closed-loop analysis several approaches can be used:

The first approach is to use built-in hard coded PI controller

```
pi_pitch_controller P_rated Omega_rated  Kp  Ki  K1 K2
 ... omega_filt  csi_filt   pitch_time  type;
```

e.g.

```
pi_pitch_controller 5200 1.2671 0.771100 0.319309 102.68665 754.18745
... 0.6 0.7  10  1;
```

Where: `P_rated` is the rated power express in kW.

`Omega_rated` is the rated rotor rotational speed express in rad/s.

`Kp` and `Ki` are the proportional and the integral gain of PI pitch controller. The values can be computed with HAWCStab2 and are in the controller ”input” file.

`K1` and `K2` are two constant for the gain scheduling of the PI gains. The values can be computed with HAWCStab2 and are in the controller ”input” file.

`omega_filt` and `csi_filt` are the frequency and the damping ratio of the rotor rotational speed feed-back.

`pitch_time` is the time constant of the pitch actuators. They are described as a first order filter.

`type` should be 1 if the control is for constant power and 0 if it is for constant torque.

Furthermore, the following should be included

```
begin controller ;
  begin input ;
    constraint bearing1 shaft_rot ;
    constraint bearing2 pitch1 collective ;
    constraint bearing2 pitch1 cosine ;
    constraint bearing2 pitch1 sine ;
  end input ;
  begin output ;
    constraint bearing1 shaft_rot 1 only 2 ; 1
    constraint bearing2 pitch1 1 only 1 collective ; 2
    constraint bearing2 pitch1 1 only 1 cosine ; 3
    constraint bearing2 pitch1 1 only 1 sine ; 4
  end output ;
end controller ;
```

The inputs are defining how the wind turbine is controlled. The outputs are defining which sensors the controller is using. The cosine and sine pitch actuators/sensors can be used by an individual pitch controller in the Coleman coordinates.

Additional outputs can be added to the output vector. Those will not be used by the current controller but they can be used to examine e.g. their transfer functions.

The second approach is to export the full system matrices and do the closed-loop calculations in another program e.g. Matlab, where any controller can be linearized and analyzed.

4.2. Examples with the command line program: HAWC2S.exe

When using HAWC2S the commands that in HAWCStab2 are selected through the GUI interface must me included in the htc file as command lines. These will be executed as a workflow.

Using

```
compute_optimal_pitch_angle use_operational_data
```

where the values from the **operational_data** block are used, alternatively using

```
compute_optimal_pitch_angle nobladedeform notipcorrect noinduction
  minwind maxwind noptiwinds mingen maxgen
  gearratio opt_lambda maxpow minpitch
```

where

```
minwind maxwind noptiwinds mingen maxgen gearratio opt_lambda maxpow minpitch
```

are numbers, can be used instead of the steps in Sec. 4.1.1.

Other commands available are:

- `compute_steady_states` - (Sec. 4.1.2)
  Parameters:
  - `nobladedeform`
  - `notipcorrect`
  - `noinduction`
  - `gradients`

  Unlike the previous version `compute_steadystate` it does not save any output. The command `compute_steadystate` is still functional.

- `compute_stability_analysis` - (Sec. 4.1.3)
  Parameters:
  - `matrixwriteout`
  - `eigenvaluewriteout`
  - number of modes
  - maximum damping ratio
  - minimum frequency
  - zero pole threshold
  - aerodynamic and deflection ratio
  - `frequencysorting`
- `save_ol_matrices` - Writes out the A,B,C,D matrices to text files
- `save_ol_matrices_full` - Writes out the M,D,K matrices to text files
- `save_ol_matrices_all` - Writes out both A,B,... and M,... matrices to text files
- `compute_controller_input` (Sec. 4.1.4)
  This command needs the block `controller`
- `save_beam_data`
- `save_blade_geometry`
- `save_aero_point_data`
- `save_profile_coeffs`
- `save_power`
- `save_induction`

4.3. Examples with the command line program: HS2pid.exe

This program is free but has reduced functionality. Its sole purpose is to provide tuning parameters for a PI controller for the wind turbine. The program is hard coded with blade torsion disabled. If blade torsion is to be included in the analysis HAWC2S.exe is to be used instead.

Procedure for using HS2pid.exe to tune the Basic DTU Wind Energy controller [3].

- The operational parameters should be added to the htc file (sec. 4.1.1).
- The controller data parameters should be added to the htc file (sec. 4.1.4).
- Execute "HS2pid.exe xxx.htc" in a MS-DOS command prompt.
- Use the calculated values from *controller_input.txt* to tune the controller in the htc file.

The closed loop frequencies should be below the first tower mode. Thus for a floating wind turbine, very low frequencies has to be selected.

5. Keyboard shortcuts

Keys	Action
Shift + x	Rotation about tilt axis
Shift + y	Rotation about yaw axis
Shift + s	Zoom out
Shift + w	Zoom in
Arrow up	Move turbine up (only in turbine view)
Arrow down	Move turbine down (only in turbine view)
Shift + h	Recenter the view
Shift + b	Toggle between blade turbine views
Shift + v	Transparent view
Shift + n	Toggle drawing of nacelle
Shift + a	Decrease amplitude of modal vibration
Shift + q	Increase amplitude of modal vibration
Shift + f	Animate forces due to vibration
Shift + k	Decrease speed of modal vibration
Shift + i	Increase speed of modal vibration
Shift + c	Draw aerodyn. choord. sys.
Shift + e	Draw struct. choord. sys.

Appendix A.

GnuPlot files

Listing A.1: Gnuplot commands used to power and pitch figure.

```
reset
set term post eps soli mono 12
set out 'power_and_pitch.eps
set key at 15,25
set size 0.5,0.6
set xr [4:25]
set yr [-3:26]
set format y '%3g'
set xlabel 'Wind speed [m/s]'
set ylabel 'Pitch angle [deg] and aero. power [MW]'
plot 'operational_data.opt' us 1:2 t 'Pitch angle' w lp pt 7 lt 7, \
     'operational_data.opt' us 1:($4*1e-3) t 'Aero. power' w lp pt 6 lt 7
set term wxt
set out
```

Listing A.2: Gnuplot commands used to deflection figure.

```
reset
set term post eps soli mono 12
set out 'deflec.eps
set key right
set size 0.4,0.4
set xr [3:25]
#set yr [0:11]
set format y '%3g'
set xlabel 'Wind speed [m/s]'
set ylabel 'Blade tip deflection [m]'
plot 'def.pwr' us 1:($12+3.766) t 'Flapwise' w lp pt 7 lt 7, \
     'def.pwr' us 1:($11) t 'Edgewise' w lp pt 5 lt 7
set term wxt
set out
```

Listing A.3: Gnuplot commands used to power and pitch figure.

```
reset
set term post eps soli mono 12
set out 'torsion.eps'
set key left bottom
set size 0.4,0.32
set xr [0:90]
#set yr [-1.5:2]
set format y '%3g'
set xlabel 'Blade curve coord. [m]'
```

```
set ylabel 'Local blade torsion [deg]'
plot 'opt_u5000.ind' us 1:($29*180/pi+0.2477776000E+01) t '5 m/s' w lp pt 1
    lt 7, \
      'opt_u10000.ind' us 1:($29*180/pi+0.4800000000E-04) t '10 m/s' w lp pt
         2 lt 7, \
      'opt_u15000.ind' us 1:($29*180/pi+0.1074508600E+02) t '15 m/s' w lp pt
         3 lt 7, \
      'opt_u20000.ind' us 1:($29*180/pi+0.1713331300E+02) t '20 m/s' w lp pt
         4 lt 7, \
      'opt_u25000.ind' us 1:($29*180/pi+0.2234911500E+02) t '25 m/s' w lp pt
         5 lt 7
set term wxt
set out
```

Listing A.4: Gnuplot commands used to generate modal frequencies figure.

```
reset
set term post eps soli mono 12
set out 'turbine_frq.eps'
set key at 24,0.6
set size 0.4,0.8
set xr [4:25]
set yr [0.2:1.2]
set format y '%3g'
set xlabel 'Wind speed [m/s]'
set ylabel 'Modal frequencies [Hz]'
plot 'turbine_ae.cmb' us 1:2 t 'Lat. twr' w lp pt 1 lt 7, \
      'turbine_ae.cmb' us 1:3 t 'Long. twr' w lp pt 2 lt 7, \
    'turbine_ae.cmb' us 1:4 t 'B.W. flap' w lp pt 3 lt 7, \
      'turbine_ae.cmb' us 1:5 t 'DT tors.' w lp pt 4 lt 7, \
      'turbine_ae.cmb' us 1:6 t 'Sym. flap' w lp pt 5 lt 7, \
      'turbine_ae.cmb' us 1:7 t 'F.W. flap' w lp pt 6 lt 7, \
      'turbine_ae.cmb' us 1:8 t 'B.W. edge' w lp pt 7 lt 7, \
      'turbine_ae.cmb' us 1:9 t 'F.W. edge' w lp pt 8 lt 7
set term wxt
set out
```

Listing A.5: Gnuplot commands used to generate modal damping ratios figure.

```
reset
set term post eps soli mono 12
set out 'turbine_dmp.eps'
set xr [4:25]
set multiplot
set size 0.4,0.39
set orig 0,0.42
set format x '%g'
set format y '%3.0f'
set xlabel ''
set yr [40:100]
set ytics 40,20,100
set ylabel 'Damping ratio [%]'
set key at 24,60
plot 'turbine_ae.cmb' us 1:12 t 'B.W. flap' w lp pt 3 lt 7, \
      'turbine_ae.cmb' us 1:14 t 'Sym. flap' w lp pt 5 lt 7, \
      'turbine_ae.cmb' us 1:15 t 'F.W. flap' w lp pt 6 lt 7
set size 0.4,0.39
```

```
set orig 0,0
set format x '%g'
set format y '%3.0f'
set xlabel 'Wind speed [m/s]'
set ylabel 'Damping ratio [%]'
set key at 24,8
set ytics 0,2,12
set yr [0:12]
plot 'turbine_ae.cmb' us 1:10 t 'Lat. twr' w lp pt 1 lt 7, \
     'turbine_ae.cmb' us 1:11 t 'Long. twr' w lp pt 2 lt 7, \
     'turbine_ae.cmb' us 1:13 t 'DT tors.' w lp pt 4 lt 7, \
     'turbine_ae.cmb' us 1:16 t 'B.W. edge' w lp pt 7 lt 7, \
     'turbine_ae.cmb' us 1:17 t 'F.W. edge' w lp pt 8 lt 7
unset multiplot
set term wxt
set out
```

Appendix B.

HAWCStab2 Changelog

```
**********************************************************************
HAWCStab2 changelog
**********************************************************************
Version 2.8 (rev. xxx) - xx/xxx/2013 - LARH
**********************************************************************
- Soon to be implemented: Dynamic inflow.
- Soon to be implemented: More flexible handling of external systems.
- Improved aero-servo-elastic modal analysis and sorting.
- N-bladed wind turbine stability analysis.
**********************************************************************
Version 2.7 (rev. 734) - 21/OCT/2013 - LARH
**********************************************************************
- Fixed bug introduced in version 2.5 that gave minor steady state
  errors.
- Fixed bug introduced in version 2.3 that gave wrong low frequency
  Bode plots and controller tuning due to wrong aerodynamic gradients.
- Fixed bug introduced in version 2.3 that divided B-matrix w.r.t
  pitch with 3.
- Added 'constant_power' line in 'controller_tuning' block:
  0 = constant torque and 1 = constant power.
- Changed calculation of inflow angle written to result files. Now it
  the same as the one used by the internal calculations.
- Fixed bug with ct that was not fixed in V.2.6.
- When a new controller tuning is calculated, this will be used by the
  closed-loop analysis instead of the values given by the command line
  'pi_pitch_controller'
- Steady state values for N-bladed wind turbines can now be calcu-
  lated. Controller tuning for N-bladed collective pitch controllers
  is also possible now.
- Write-out of closed-loop matrices is now possible from the menu.
- Added new criteria for sorting modes.
- Added new controller gain scheduling method
- Changed dialog box when performing aeroelastic and aero-servo-
  elastic modal analysis.
- Minor tweak in implementation of damping_posdef routine.
- Added HAWC2S commands:
  'compute_structural_modal_analysis',
```

```
  'compute_steady_states': it is the same as 'steady_state' but it
    doesn't save anything.
  'compute_stability_analysis', 'compute_aeroservoelastic',
  'compute_controller_input', 'save_ol_matrices',
  'save_ol_matrices_all', 'save_beam_data', 'save_blade_geometry',
  'save_aero_point_data', 'save_profile_coeffs', 'save_power',
  'save_induction'
*********************************************************************
Version 2.6 (rev. 569) - 23/AUG/2013 - LARH
*********************************************************************
- Changed write-out of "Save modal amplitudes" for blade only. Now a
  phase angle close to 0 deg. is ensured and the sign of the amplitude
  is corrected accordingly. I.e. amplitudes can now be negative.
- Fixed bug where torsional mode shape was not written correctly to
  text file
- Fixed bug where gear ratio was not read in the htc file.
- Started development of code to handle 2,3,4,... bladed turbines.
- [N.B. See V.2.7] Fixed bug with ct in BEM calculations, this should
  only have minor implications at above rated wind speeds.
- Added support for external systems. So far they are attached to the
  bottom node of the tower. This can be used for e.g. jackets modeled
  as super elements.
*********************************************************************
Version 2.5 (rev. 512) - 08/AUG/2013 - LARH
*********************************************************************
- Critical bug fix. Previous release version removed from website and
  replaced by current one.
*********************************************************************
Version 2.5 (rev. 499) - 07/AUG/2013 - LARH
*********************************************************************
- Added new save open-loop matrices options. Also added steady state
  output values of selected sensors.
- In  the operational_data block the prvs_turbine option has reversed
  logics, e.g. 0 = stall regulated, 1 = pitch regulated. This is
  opposite of previous version, but more logical.
- Changed calcalation of inflow angle written to result files.
- Various bug fixes.
*********************************************************************
Version 2.4 (rev. 380) - 10/JUN/2013 - LARH
*********************************************************************
- Added 64 bit version of HS2 to enable memory expensive calculations.
- Added Rayleigh damping model similar to HAWC2 with mass and stiff-
  ness proportional damping terms 'mx my mz kx ky kz'. N.B. 'damp'
  and 'damp_pos' might be disagreeing on the order of 'kx' and 'ky'
  in HAWC2. HAWCStab2 uses the 'damp' convention.
- Added command line program H2Spid.exe which is a command line
  version of HAWCStab2.exe with reduced functionality only able to
```

```
  calculate PI controller gains assuming rigid blades.
- Log files are now saved as 'filename'.htc
**********************************************************************
Version 2.3 (rev. 203) - 29/JAN/2013 - MHHA
**********************************************************************
- Added command line program HAWC2S.exe which is a command line
  version of HAWCStab2.exe.
**********************************************************************
Version 2.2 (rev. 181) -  6/JUL/2012 - MHHA
**********************************************************************
- ...
**********************************************************************
Version 2.1 (rev. 131) -  2/FEB/2012 - MHHA
**********************************************************************
- ...
```

Bibliography

[1] T. J. Larsen and A. M. Hansen. *How 2 HAWC2, the user's manual.* DTU Wind Energy, Roskilde, Denmark, version 4.3 edition, April 2012. Risø-R-1597.

[2] M. H. Hansen. Anisotropic damping of Timoshenko beam elements. Technical Report Risø–R–1267(EN), Risø National Laboratory, Denmark, 2001. (Available from www.risoe.dk).

[3] M. H. Hansen and L. C. Henriksen. Basic DTU Wind Energy controller. Technical Report DTU Wind Energy E-0018, Dept. of Wind Energy, Technical University of Denmark, January 2013.