Report #3: (Due: Sun 27-Oct-2024 23:00) Controller tuning

Assignment format

One person from your group submits a **PDF** of your report. The report should include descriptions and the required tables and figures and should be no more than 10 pages (excluding appendices). Other plots or tables that you deemed relevant to the assignment are in the appendix of the report. The format is up to you, but please list the names of who worked on the report.

On Wednesday (30-Oct-2024) in class, you will **orally present the report for 7 minutes** as a group, by scrolling through the report. After the presentation, there will be **8 minutes of questions from the instructors**. Not everyone in the group needs to present during the 7 minutes, but everyone should be ready to answer questions afterwards.

Please be ready to explain any trends in the plots, the reasons behind values in tables, etc. **For your group to pass the assignment**, your report should not only contain the requested tables/figures, but you should also orally demonstrate an adequate understanding of the reported results.

Report objectives

Your report should accomplish the following objectives:

- Derive the units of each controller parameter
- Calculate controller parameters using theoretical equations and compare them with HAWC2S
- Tune the pitch controller for different frequencies and damping values of the closed-loop regulator mode
- Simulate and analyse the step-wind response with different controller design
- Choose a final frequency and damping value for the closed-loop regulator mode for your design and evaluate your design's final step-wind response

Required figures and tables

This is a list of figure/tables that are required to pass. Other plots/tables that support your analysis are of course welcome.

- **Table**: Unit and values for controller tuning parameters (see Table 1).
- **Table**: Values for blade pitch controllers k_P and k_I for C1-C6. (See Table 2).
- **Plots**: Two figures of step-wind simulations, one for C1-C3 and one for C4-C6.
- **Table**: Frequency, damping, and controller values for your own controller design C7.
- **Plot**: Step-wind of your controller design C7 compared with C1.

Step-by-step guide

Parts 1 and 2 should be completed after Week 6. You will present the results to your peers in the beginning of Week 7. Part 3 can be completed after Week 7.

Part 1: Theoretical calculations and comparison with HAWC2S

To be done after Week 6.

Derive units and calculate theoretical values

Derive and insert the units of each variable in Table 1; and calculate theoretical values. Include your derivations in your report. Notice that the units for pitch angle unit are assumed to be **rad**.

Calculate values with HAWC2S

Update make_htc_files.py and myteampack to generate an htc file for controller tuning:

- Add a new method to MyHTC called make_hawc2s_ctrltune()
  - Start by copying the existing code for make_hawc2s()
  - At the end of the function, right before _update_name_and_save, add a new line that will add the controller tuning block to the htc file
    - ***Hint***: Use the existing method _add_ctrltune_block().
  - Determine how you will pass the needed keyword arguments through make_hawc2s_ctrltune() into _add_ctrltune_block().
- Update make_htc_files.py so that it generates a new htc file for tuning the controller parameters.
  - For now, use the same natural frequency and damping ratio of the speed regulator mode as the DTU 10 MW: 0.05 Hz and 70% for the torque controller ("partial load") and 0.06 Hz and 70% for the pitch controller ("full load").
- Compare the generated htc file with the DTU 10 MW controller-tuning htc file to make sure the settings look right (Learn / Miscellaneous / HAWC files for DTU 10 MW).
- Update/Use the `regions` command in your master file to reflect your turbine's opt file if the automatic region detection algorithm in HAWC2S cannot find the regions correctly.
- Run HAWC2S on your generated HTC file.
- Fill out the HAWC2S column in Table 1.

Compare with theoretical values

Use the equations in class to evaluate the theoretical values for the controller-tuning parameters. Note that the rotor inertia and aerodynamic gains are printed in the _ctrl_tuning.txt file generated by HAWC2S in the step above (**pay attention to the unit of aerodynamic gains in the "_ctrl_tuning.txt" file**).

Table 1: controller tuning parameters

Parameter	Units	Theoretical	HAWC2S	Natural frequency	Damping
Optimal Cp tracking K factor				N/A	N/A
Proportional gain of torque controller					
Integral gain of torque controller					
Proportional gain of pitch controller					
Integral gain of pitch controller					

Coefficient of linear term in aerodynamic gain scheduling, KK1				N/A	N/A
Coefficient of quadratic term in aerodynamic gain scheduling, KK2				N/A	N/A
Aerodynamic gain at pitch = 0 deg				N/A	N/A

Changes to make_htc_files.py/myteampack

- New method to myteampack.MyHTC: . make_hawc2s_ctrltune()
- New htc file: HAWC2S to tune your controller with a flexible rotor and DTU 10 MW controller poles (the natural frequency and damping ratio of the closed-loop regulation mode)

Part 2: Pitch controller tunings for various frequencies

To be done after Week 6.

Use the torque-controller values from Task 1. Update make_htc_files.py so that it generates a new control-tuning htc file for the **pitch**-controller poles listed in Table 2. *Be sure to give each file a distinct name!* Run HAWC2S on all the files to generate control-tuning parameters for each pole and fill out the table.

In your report, please explain: for the same ω_Ω, why is the k_P different but k_I is the same when using Constant Power and Constant Torque?

Table 2: Pitch controller tuning

	Generator Torque control type	ω_Ω (pitch controller)	ζ_Ω (pitch controller)	k_P	k_I
C1	Constant Power	0.05 Hz	0.7	?	?
C2	Constant Power	0.01 Hz	0.7	?	?
C3	Constant Power	0.10 Hz	0.7	?	?
C4	Constant Torque	0.05 Hz	0.7	?	?
C5	Constant Torque	0.01 Hz	0.7	?	?
C6	Constant Torque	0.10 Hz	0.7	?	?
C7	?	?	?	?	?

Please provide a written explanation to the question: For the same ω_Ω, why is the k_P different but k_I is the same when using Constant Power and Constant Torque?

Changes to make_htc_files.py/myteampack

- New htc files: 6 HAWC2S files to tune controllers for the natural frequencies and damping ratios of the closed-loop regulation mode C1 through C6.

Part 3: Step-wind responses

To be done after Week 7.

Generate step-wind htc files

The wind speed should go from 4 to 25 m/s with steps of 1 m/s. Each wind speed must be kept at least 40 seconds, and you should simulate and then discard 100 extra seconds of steady wind in the beginning to get rid of transience.

- Given the specifications above, how long is the total simulation? When should HAWC2 start recording data?
- Modify MyHTC so it has a new method make_step(). You must determine what inputs are necessary to make_step(). This method should:
    - Update the relevant controller parameters
        - Note there is a lacbox function to load values from ctrl_tuning.txt into a dictionary.
    - Delete the hawcstab2 block (see example in _del_not_h2s_blocks())
    - Set the simulation time to the desired start and stop time
    - Set wind speed to the initial wind speed
    - Set the turbulence intensity to 0
    - Set the shear profile to [3, 0]
    - Set the turbulence format to 0
    - Set the tower shadow to 0
    - Add the necessary wind_ramp_abs values to create your desired step wind
        - An example of how to add lines is in _add_ctrltune_block()
    - Update the filename and saves the file
- Modify make_htc_files so it calls make_step() for a SINGLE pole. **NOTE**: the HAWC2 file should be saved in a subfolder called htc/!
- Compare the generated file with the DTU 10 MW step-wind example (Learn / Miscellaneous / DTU 10 MW example files) and make sure all settings look correct.
- Modify make_htc_files.py to generate the step-wind files for **C1** to **C6**, saving them in htc/.

Run and analyse the results

Run HAWC2 on the 6 generate step-wind files. **Note** this will take some time. Step-winds are long.

Provide two figures (C1-C3 on one figure, C4-C6 on another figure) where wind speed, pitch angle, rotation speed, and **electrical power**[1] vs. time must be presented in the same plot. Please provide some brief comments on your observations (e.g., a bullet-point list).

Design and evaluate your controller

For C7, design your own controller, update make_htc_files.py to generate the 7th HAWC2 file, and simulate the step wind.

Can your design C7 perform better than C1 in terms of overshoot and settling time? Compare your final C7 step-wind response with your C1 step-wind response and analyse the result.

Changes to make_htc_files.py/myteampack

- New method in MyHTC: make_step()
- New htc files: 7 HAWC2 step-wind files, for **C1** to **C7**, saved in htc/ folder

---

[1] Electrical power output in the generator servos DLL output (the 2nd output of this DLL)