

Comp3402 Wind Turbine Control

C.B.Price November 2022

Purpose

(i) To learn about Wind Turbine control strategies for Regions 1-4 with a focus on Region 2 and Region 3. (ii) To conduct a series of **investigations** and subject the observations to analysis.

Files Required

Unreal4 resources and Octave scripts.

ILO Contribution

LO 4

Send to Me

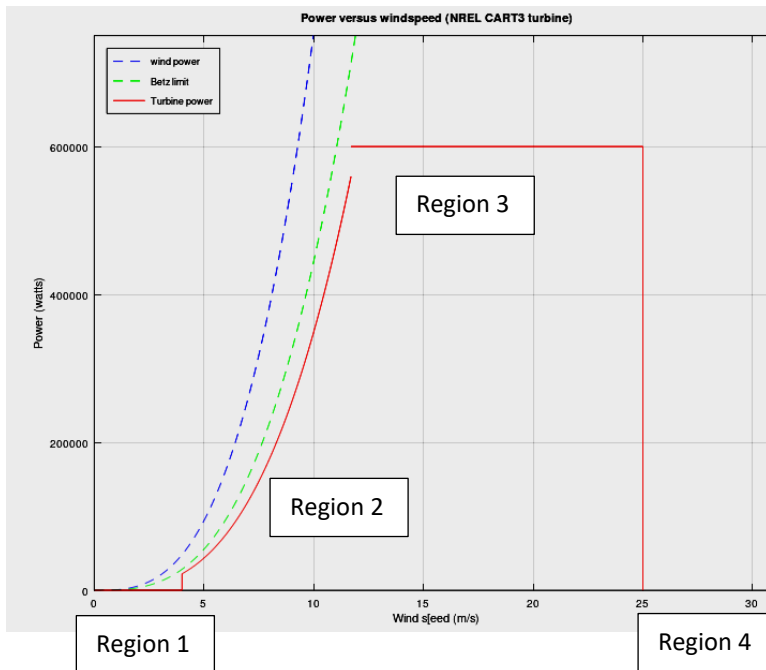
nix

Homework

Read chapter 4

1 Parameters for the NREL-CART3 600kW turbine

Below are the parameters for this wind turbine, defined in Unreal and also Octave. On the left is the power curve showing the four regions, the energy in the wind and the Betz limit. These curves are all proportional to wind-speed cubed, i.e., v^3 .



R	Rotor radius	20m
ρ	Air density	1.2 kg/m^3
J	Inertia	613,301
λ_{opt}	Optimal tip speed ratio	6.6
C_{pmax}	Maximum power coefficient	0.4633
ω_{rated}	rated angular velocity	3.9521
v_{cutIn}	Cut-in wind speed	4 m/s
v_{cutOut}	Cut-out wind speed	25m/s
v_{Rated}	Rated wind speed	11.7 m/s

Important Stuff

- 1) Power is proportional to angular speed cubed $P \propto \omega^3$ in Region II
- 2) In Region II, the turbine blades are 'pitched' (angle **beta**) so they extract less power from the wind.

2 Overview of the Turbine response to the full range of wind speeds.

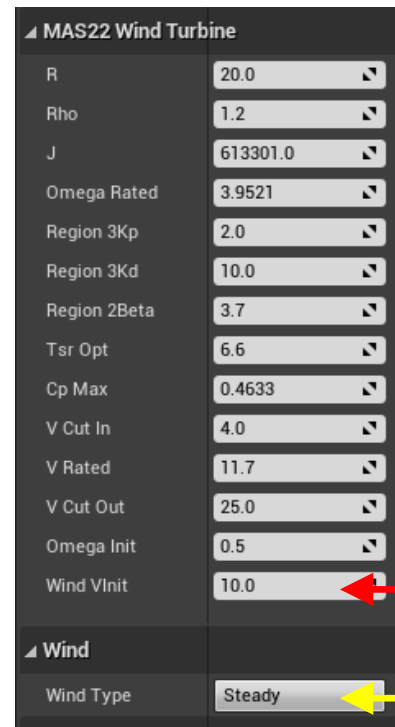
Here we shall run the turbine with various wind speeds which we shall set manually.

(a) Open up the level **MAS22_SciencePark** in the Editor as shown in class, find the wind turbine and left click to bring up its Details panel shown on the right.

(b) Make sure the parameters agree with what is shown and make sure the **Wind Type** is set to **Steady**.

(c) Choose a series of windspeeds to cover the full range of Regions II to IV (say 10 values). For each wind speed set it in the parameter panel (red arrow) and run (Play) the level. You can divide this work with folk around you.

Watch the turbine, observing and change in speed. Look at the HUD values for power and omega. Run your level until power does not change (to the left of the decimal point). Press F1 to pause.



Read off the following values and put them into a table:

windV	power	omega	beta	tsr	Cp

(d) Now, let's see what we can get from this table.

- (i) In which region does the power increase with wind speed? Is this what you expect?
- (ii) In which region does the power stay the same as wind speed increases? Is this what you expect?
- (iii) In which region does the blade pitch (beta) stay the same as wind speed is increased? As expected?
- (iv) In which region does the blade pitch increase as wind speed is increased? As expected?

(e) Now download and unzip the folder **Octave Assets** and using Octave, open the script **Plot_Power_WindV**. Add your windV and power values into matrix **A** as rows, windV then power. At command line type the name of this script, and your data will appear on the theoretical plot.

3 Automatic Collection of data for the full range of wind speeds

Here we shall configure the engine to step through a range of wind speeds from 3.0 m/s to just above cut-out (25 m/s).

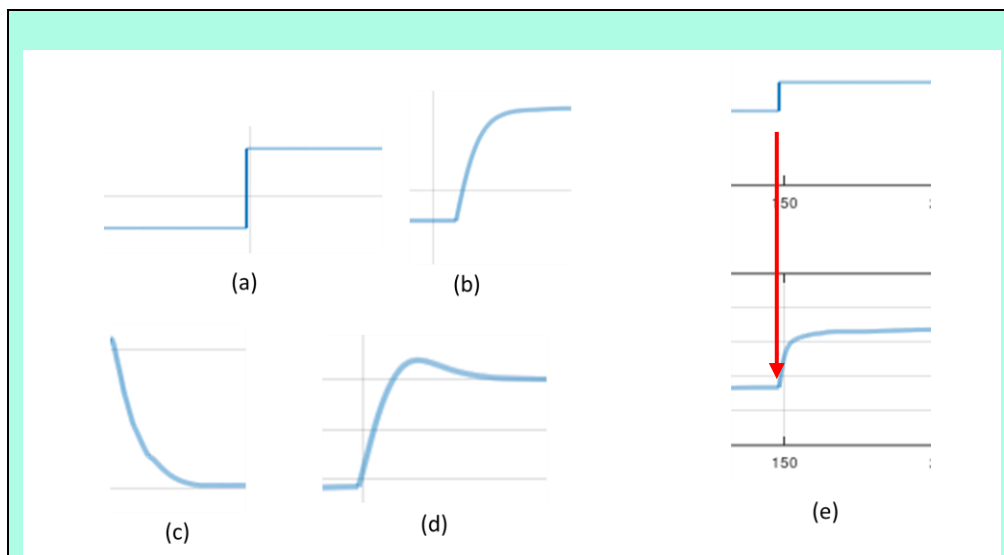
(a) Make sure the **Wind Type** is selected as shown and the **Wind Vinit** is set to 3.0



(b) Run the engine until you hit Region IV. (How will you know you are in this region by observation?)

(c) Navigate to the Unreal SciencePark folder, and open up **OctaveFiles**. Run the script **WT** (the name of this MAS22 Actor and it will plot a load of graphs.

These look quite complex when viewed together, but there are several patterns which you can see. Let's have a look at these.



Plot (a) is a **step** where something changes instantaneously with time. This is almost always an *input* to a system, rather than a *response*. An example is wind speed which is specified.

Plot (b) is a **rising response** where something *real* is changing. Real things can't change suddenly, as in (a), but need *time* to respond. This curve shows how most real (physical) things increase with time.

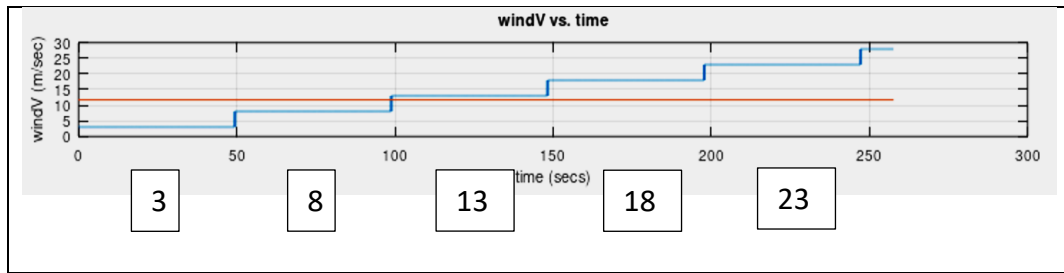
Plot (c) is a **falling response**. It is the equivalent of (b) but here something is decreasing with time.

Plot (d) shows a **rising then slightly falling response**. Here, something takes time in rising, but it overshoots its 'target', then takes a little more time to come down to its target value.

Note that in all the above, the values all bottom out to a steady state which does not change with time.

Plot (e) shows a 'cause and effect', or an 'input and output', or a 'stimulus and response'. The top shows the input and the bottom shows the response; the input step causes the response to grow as in (b).

Here's a reminder of the wind speeds associated with each segment of the **windV vs. time** plot.



(d) Identify the times when the following transitions occur. HINT: Look at the plots of beta or omega
(i) Region I to II, (ii) Region II to III, (iii) Region III to IV

(e) At each step it takes time for the controller to reach its target value. Look at the power curve and compare the values reached with the target values you tabulated in Activity 2.

(f) When there is a step in wind, why does omega not increase as a step, but shows a 'rising response'?

(g) Look at the omega-time curve. Which other curve closely resembles this? Why?

(h) Overall, how well does the controller perform in extracting power from wind over regions II and III?

(i) Why does beta stay constant up to 250 secs then increase?

(j) How would you describe the increase in beta e.g. is it 'proportional' to wind speed, or something else (qualitative answer, please)?

(k) What makes omega and power fairly constant after 250 secs?

(l) Is there anything else that grabs you, looking at this plot as a whole?
