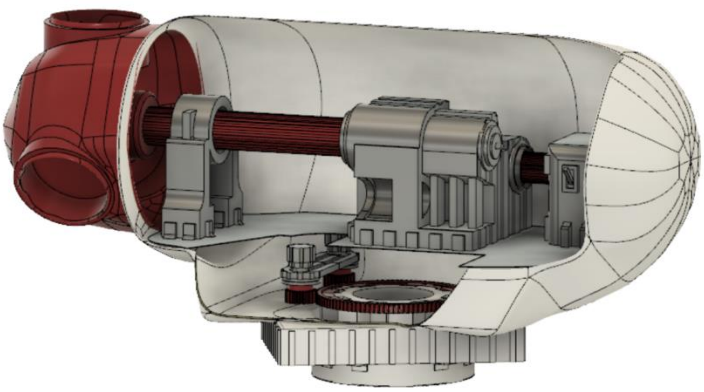
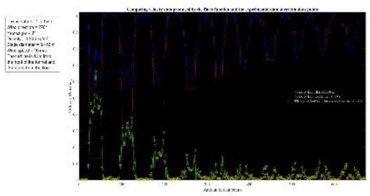
Design and Computing first year project

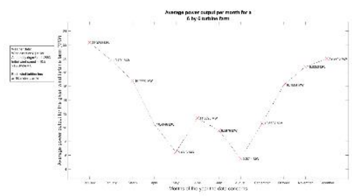
Norse wind turbine project

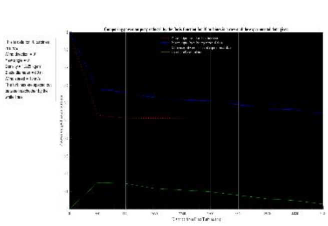
Reuben Marland

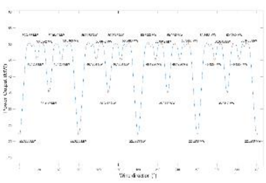
Mechanical Engineering

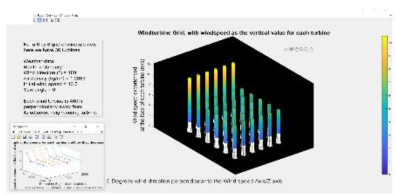












Content List

General Introduction

The Orkland Islands resides in north west Scotland as a sub-Antarctic archipelago, where wind turbines farms have had popular occupancy. Wind turbines are not the only entity on the island as there is also an indigenous puffin population as well as cultured locals. These locals would not favour the placement of wind turbine farms in their homeland, and consequently, being the planner, this is one of numerous factors that need to be carefully considered when establishing a new farm. From rotor diameters to appropriate weather conditions, this report explores how a wind turbine farm could be built within this region.

As a Mechanical Engineer, I will be exploring a detailed nacelle design, which includes a gear system to produce power, as well as a yaw system so the kinetic to electrical energy conversion is working at maximum efficiency. Another part of the project is to analyse various parts of data concerning the wind turbines, such as power output in various scenarios and how much wake can affect the windspeeds at consecutive wind turbines.

Design

Introduction

Within this project I was tasked with generating experimental turbines designs and further-more designing in more detail which one I thought suited best. The tool I used to provide much more detail was Autodesk Fusion 360. This involved generating a 3D construct of the nacelle of a HAWT wind turbine. Though before designing this, I had to decide on several factors: What type of turbine? What mechanical system will it use within the nacelle? What type of yaw system is in use? The first header of the Design section of this report will explore these ideas, followed by analysing techniques on what to prioritise.

Sketches

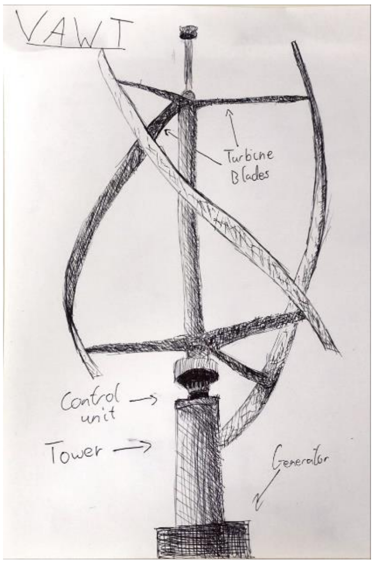


Figure 1 – VAWT

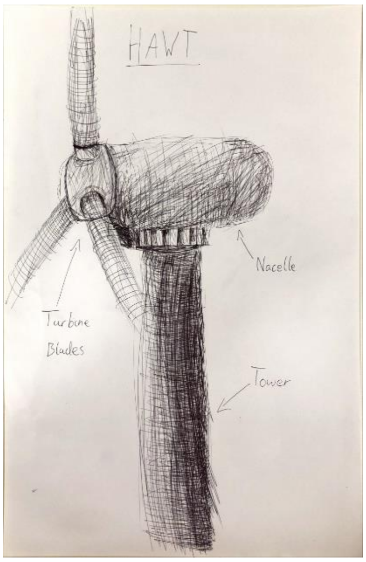


Figure 2 – HAWT

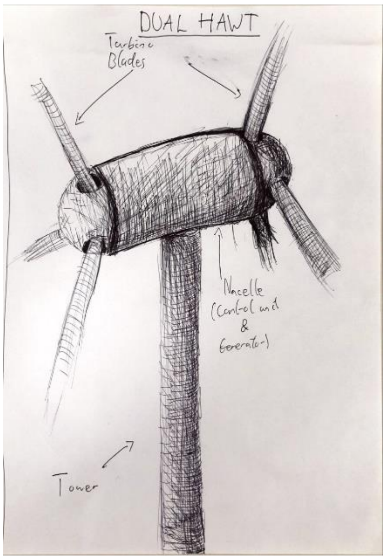


Figure 3 – Dual HAWT

Before I started, I knew I was going to use a HAWT as my design for my wind turbine nacelle though I decided to investigate two other designs first. Figure 1 describes a VAWT. This turbine has vertical spinning blades in comparison to most other turbines. I decided not to use this as the power generation is far less than a normal HAWT. The structure is overall more complex and requires more structural cost. The only plus side to this turbine in my opinion is the aesthetic. Figure 3 is a complete concept design of a turbine that realistically does not work in real life. Wind only blows from one direction and hence having a turbine on both sides would not make sense unless the blades on the rear were forward acting. In that case the wind would be able to turn the rear blades but so much of the wind would get caught in the wake so the power generation would not be as efficient. Overall, this design would cost a lot more for how much more power it would produce. Whilst costs are saved in the tower, it is no where near enough. Figure 2 is the ultimate design I ended up using as a HAWT shape is the most cost effective and standardised. In the real world, the parts to build these turbines come at lower rates because they are much more commonly built than any other turbine.

Even though I knew what I was going to use, I implemented the use of Pughs conversion matrix to understand just how much worse the alternate designs were:

Criteria	**Weight**	**HAWT**	**VAWT**	**DUAL HAWT**
Cost	4	0	-1	-1
Durability	2	0	0	-1
Power Generation	4	0	-1	1
Ease of build	1	0	-1	-1
Environmental impact	2	0	1	0
repair cost	2	0	-1	0
Totals			**-9**	**-3**

Figure 4

As you can see both designs come out on the negative side though surprisingly the Dual HAWT was not as bad as I considered it to be.

More decisions had to be made to make a choice on priority when considering my turbine design. To perform this procedure, I chose to use pairwise comparison tables. One for the components of the wind turbine and one for the factors of building a wind turbine.

Components		**[1]**	**[2]**	**[3]**	**[4]**	**[6]**	**[7]**	**[9]**		
	X	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Total**	**%**
Gliding Yaw	**1**	**X**	0.33	0.11	0.11	0.33	1.00	1.00	**2.89**	**6**
Roller Yaw	**2**	3.00	**X**	0.33	0.33	1.00	1.00	1.00	**6.67**	**14**
Direct Drive	**3**	9.00	3.00	**X**	1.00	1.00	3.00	3.00	**20.00**	**41**
Gearbox	**4**	9.00	3.00	1.00	**X**	1.00	3.00	3.00	**20.00**	**41**
Steel	**5**	3.00	1.00	1.00	1.00	**X**	3.00	1.00	**10.00**	**20**
Fibreglass	**6**	1.00	1.00	0.33	0.33	0.33	**X**	0.33	**3.33**	**7**
Cost	**7**	1.00	1.00	0.33	0.33	1.00	3.00	**X**	**6.66**	**14**

Figure 5 – Pairwise components table

Factors		**[1]**	**[2]**	**[3]**	**[4]**	**[6]**	**[7]**	**[9]**		
	X	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Total**	**%**
Cost	**1**	**X**	**3.00**	**1.00**	**3.00**	**9.00**	**3.00**	**3.00**	**22.00**	**45**
Durability	**2**	**0.33**	**X**	**0.33**	**1.00**	**3.00**	**3.00**	**3.00**	**10.66**	**22**
Power Gen	**3**	**1.00**	**3.00**	**X**	**3.00**	**3.00**	**9.00**	**3.00**	**22.00**	**45**
Environmental	**4**	**0.33**	**1.00**	**0.33**	**X**	**1.00**	**1.00**	**1.00**	**4.67**	**9**
Access	**5**	**0.11**	**0.33**	**0.33**	**1.00**	**X**	**0.33**	**0.33**	**2.44**	**5**
Repair Cost	**6**	**3.00**	**0.33**	**0.11**	**1.00**	**3.00**	**X**	**3.00**	**10.44**	**21**
Ease of build	**7**	**1.00**	**0.33**	**0.33**	**1.00**	**3.00**	**0.33**	**X**	**6.00**	**12**

Figure 6 – Pairwise factors table

From viewing the pairwise table of figure 5, I was able to deduce that the component that was the most important was the system within the nacelle. I ended up going for a standard gearbox system as I considered the price being lower as well as it being more manufacturable currently. Though as a note, a direct drive turbine can still be improved and are most likely going to be the popular system in the future. To decide between the two, I had to consider multiple factors. I performed this comparison using another pairwise table in figure 6. This showed that the most important factors to me were the cost and the power generation. This agrees with my decision of choosing a HAWT as my turbine as well as focusing on using a standard gearbox.

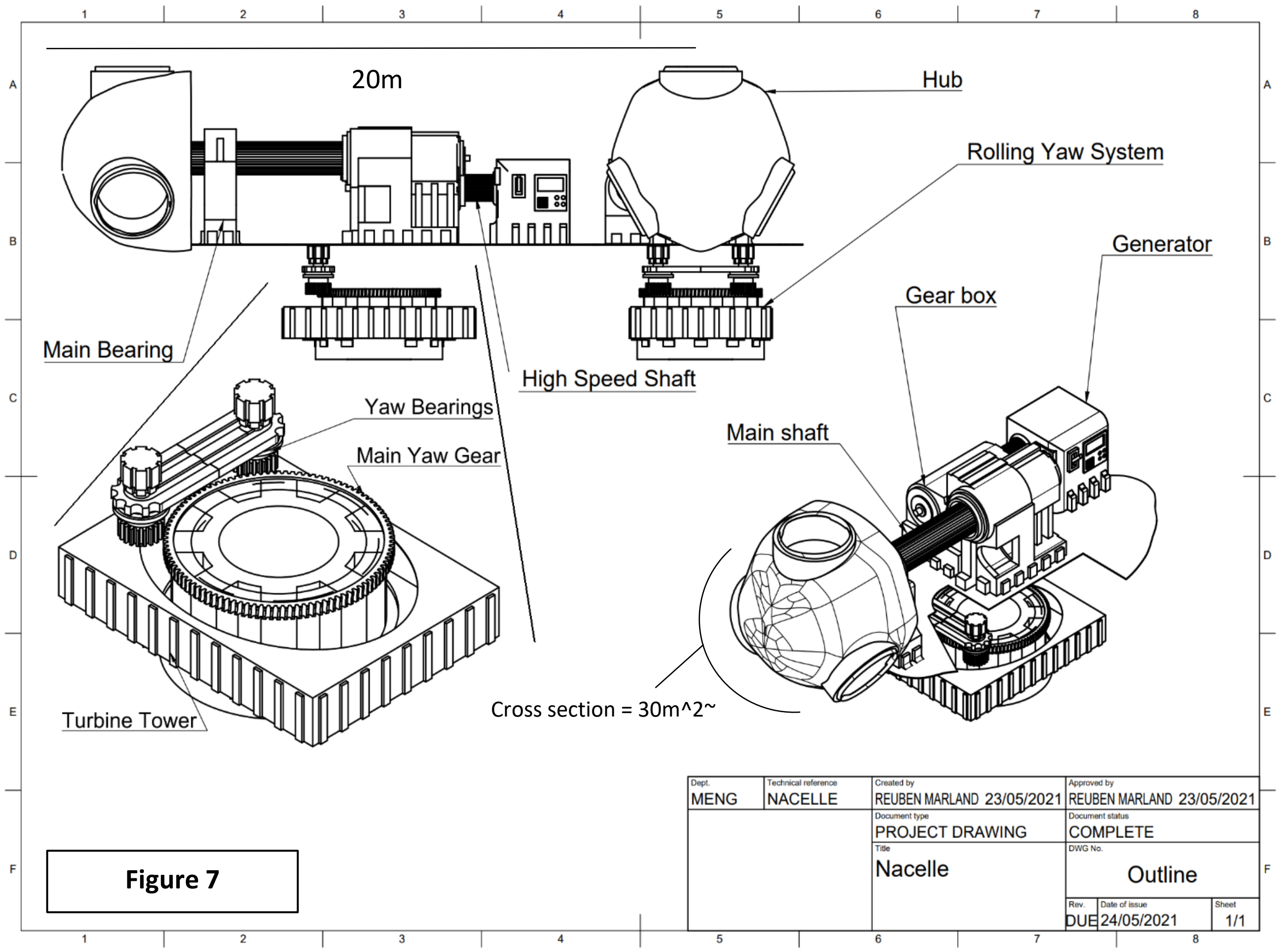


Figure 7

This is the CAD drawing of my turbine nacelle design. When the chassis is over the internal system it has a length of 20m and a cross section of roughly 30m^2. Within the fusion 360 file the systems have been attempted to be simulated to be like that of a real turbine. There are functioning gear ratios of a real-life turbine between the main shaft and the high-speed shaft, as well as the gears around the yaw system turning correctly with the correct number of teeth intersecting. Thus, the nacelle can spin 360 ° and the hub can also rotate with theoretical blades and this in turn rotates into the gear box and later the generator. Note the 'Tower Top' component needs to be grounded for nacelle rotation and the 'Nacelle Body' needs to be grounded for hub rotation

Conclusion

In general, I would have liked to be able to go more in depth with my 3D design. Being accurately able to build the smaller parts of the power generation system in detail would be a big step, as the accuracy of its real-life function ability would improve. From using comparison matrices, I was able to validate the choice of wind turbine and how varying factors would prioritise within the construction and full design process of a turbine.

<u>Coding</u>

Introduction

The aim for the coding side of this project was to effectively utilise the Floris function given within Matlab. This varies from validating the code, to using it to produce graphs relating to power generation and wake studies. As a summary, two validation graphs are produced to check if the Floris function is accurate enough to simulate a real-life environment. These graphs are to do with the power produced from turbines in a line and then how much more accurate the Floris function becomes with velocity data as the number of data points increases.

Validation

The first validation is a plot of normalised velocity against number of data points. Within this graph are 3 lines defined on the legend that illustrate the accuracy of the Floris function

This is data for 1 turbine
Wind direction = 270°
Yaw angle = 0°
Density = 1.225 kg/m³
Blade diameter = 0.416 m
Wind speed = 10 m/s
The turbine is 10m from the front of the tunnel and 3 metres from the floor

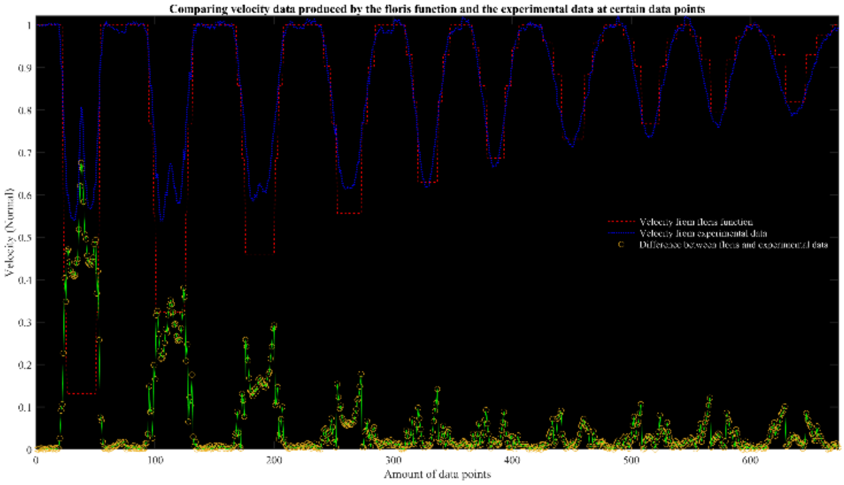


Figure – 8 (Normalised Velocity against No. Data Points)

From this data we can see how the velocity produced from the Floris function becomes a lot more accurate as time goes on. Due to the nature of the function, it makes sense that increasing the number of data points produces a better curve as the average works out more accurately to a precise number. The green line with the yellow markers describes the difference between the experimental data and the Floris data and is effectively a data bar. It is clear that the error is very great at a lower number of data points and gradually decreases over the increasing amount of data points to until it normalises at around 350 (extrapolation). There-fore we can deduce that the Floris function becomes far more accurate at a high number of data points and we should utilise this in future Floris usage.

This is data for 10 turbines in a row
Wind direction = 0°
Yaw angle = 0°
Density = 1.225 kg/m³
Blade diameter = 80 m
Wind speed = 10 m/s
The turbines are spaced out between eachother by the white lines

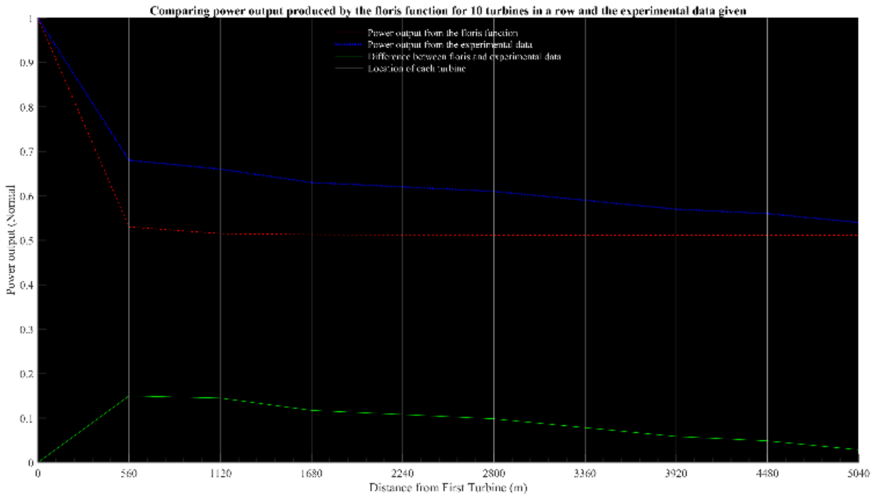


Figure – 9 (Normalised Power Output against the Distance from the First Turbine)

This data shows how the power output changes at different wind turbines that are lined up in a row. This data shows the normalised power at each turbine, which are at the locations of the vertical white lines. This validation shows that the Floris function remains with a medium sized error that slowly decreases throughout. This would suggest that when using the Floris function for future plots, it should be considered that there may be an error, more predominantly at the start of the plot.

Main body code and analysis

The main section of Coding within this project was to produce multiple graphs studying power and wind-speed values relating to various other variables and factors.

This code was split into 4 main sections. The first determined all the variables, which varied from user inputs, to external data packs, to preassigned variables. The user input variables that needed to be extensively coded include asking the user for specified grid dimensions for a square wind turbine farm, as well as asking what month the user would like to analyse. The relevant data from the months was extracted from the weather info pdf provided on blackboard and converted into a .csv file. The second Section produces two plots which show the location of the wind turbines in a farm and the data for the wind-speed experienced at each turbine location.

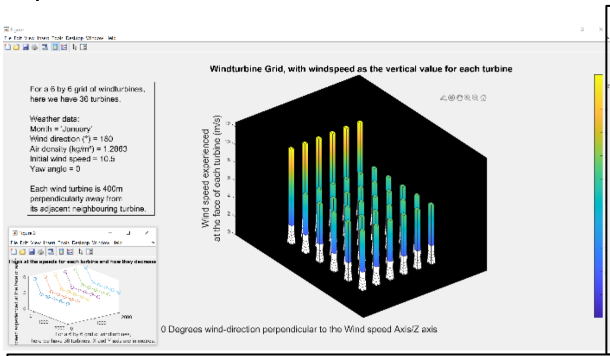


Figure 10

This is an example of what the second section of my code produces. This combines to showing the location of the individual wind turbines in a farm. The lower white section of each bar represents where the individual turbines are and the vertical colour bars emerging from the top represent the wind-speed experienced at the face of each turbine.

The figure also comes equipped with an annotation box that displays the variables which have been selected. For example, in this plot, the user has chosen a 6 x 6 plot of turbines, along with using the data from the month of January. There is also a custom data option. An extra plot is also included in the bottom left which remove the bars and include only the wind-speed points for a clearer look at the data.

In conclusion the data shown in this graph exhibits that when the wind direction is parallel to the wind turbine grid, the wind-speed occurring at each wind turbine on average is far lower than if the wind was coming at an angle. With the code setup, the user can explore different wind direction angles using the custom data option and work out what would be optimal.

The third section of my code displays a plot that shows the average power output at any given moment for a certain windfarm farm for each month of the year. And relevantly compares which months are more suitable for power generation

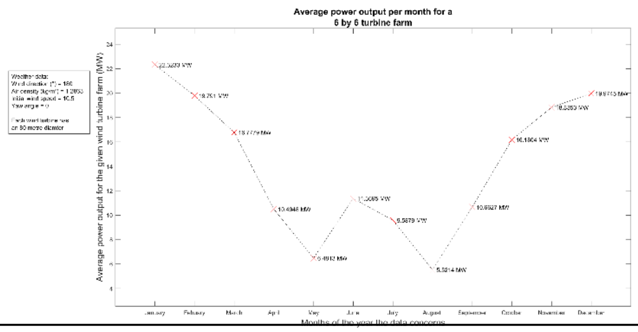


Figure 11

This graph clearly shows that the power generation changes greatly per month, the optimal being January and December or more notably the Autumn and Winter months. This plot also comes equipped with an annotation box and title that display the active variables for this data.

The data shown used to display the points for each month come from the weather info data pack, the same used for the user month selection data.

We can easily conclude from this data that the wind turbine farm can only operate optimally in certain months, though on average for it to be running well for around half the year, the costs would ultimately be covered as repairs could be made in the less active months.

The fourth section of code produces a plot of the power output for degree increments of wind direction. Analysing this data will let the user identify what is the optimal and worst wind direction for the farm.

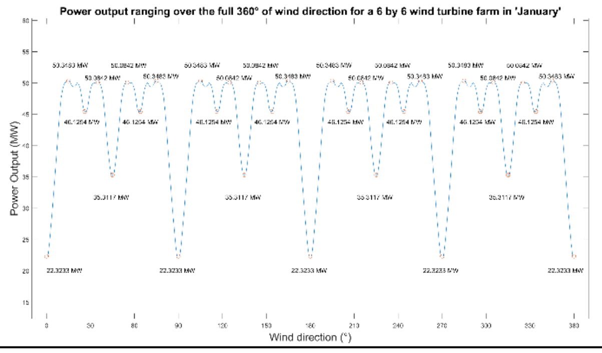


Figure 12

The title of this graph acts to show the variables selected for this data. The data for this figure and the previous two have come from the same values of variables. As this data is again for a 6 x 6 farm in the month of January. The power output distinctly varies for certain angles, by as much difference as a 60% decrease from max.

The reason for this data being so interchangeable is due to the wind direction being perpendicular or parallel to the square wind farm grid. If the wind is in this direction, such as 90 degrees, the wake of the front row turbines will be directly in front of those behind it, and those behind it lie in a row so even more wakes occur. This contrasts to a rough direction of 55 degrees where the wind must be acting at such an angle that there is a minimum amount of wake affect from each turbine, thus the turbines must not be lying in a row as much.

Conclusion

To conclude the coding side of this project, there is a lot more detail I would have like to go into for some of the certain graphs, e.g., the 3D plot but my current knowledge of Matlabs held me back slightly. I can tell that there are areas of my code which could be easily shortened, for example using for commands to create a loop to more easily set up my variables. This is distinctly noticeable in the line space that my month selection code in section one took up, along with my power sum month code in section 3.

Overall, the code works as it should and produces the desired user outputs. The custom data input allows the user to try alternative data for variables which can produce some interesting graphs, though they become more hypothetical.

From combining the validation code and studying where the errors are with the Floris function, being able to identify the areas of my code that look slightly off, make sense, and that using more data points will make my data more accurate. This is mostly noticeable in figure 11 and 12 where the curve becomes very well defined.

General Conclusion

Overall, this project has greatly improved both my design and coding skills and I have been able to learn about how they can intertwine with each other and how both need to be utilised in a design environment before the idea is implemented in the real world. Professional comparison standards such as Pughs matrix and using pairwise have also acted as a useful tool in deciding how to prioritise my focus, deciding what would be best for the design. There are a lot more factors than highlighted in this project that would play a part in establishing a wind turbine farm, especially in the Orkland islands. However, as a simple design and setup for a farm, we can begin to build an idea of what it may be like.

References

No external references used, only what was provided on blackboard and in the assessment files.