Final Degree Project (TFG)

Engineering in Industrial Technologies

WIND TURBINE MODEL WITH IMPLEMENTATION OF A PITCH CONTROLLER

MEMORY

Author: Carlos Domenici Massanet

Supervisor: Marc Cheah Mañé

Date: January 2022

ETSEIB

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

ETSEIB

Abstract

This work is inspired by the motivation to give more importance to the difficult task of mathematical modeling in the engineering industry by providing an own wind turbine model design. Model design is a complex and very important work which lacks attribution in most engineering projects. Moreover, current global problems such as climate change have prompted the decision to design a model of a technology that is helping to alleviate the effects of this major problem; wind turbines, capable of generating energy through renewable sources.

This paper shows the importance of model design and focuses on the process of modeling the most essential parts of a wind turbine. The aerodynamic design that captures the kinetic energy of the wind and the gearbox which transforms that energy into generated power are the most relevant aspects. In the latter, two types of gearbox models are analyzed: one-mass and two-mass models. Both are designed and subsequently compared. In addition, a pitch controller is also implemented to automate this turbine in the case of variable winds which may damage the parts that make up the wind turbine.

Finally, to prove the correct functionality of the model designed in this work, a variable wind profile has been created, so that the dynamic behavior in different situations can be analyzed.

After validating the design of this work it is concluded that the realization of mathematical models can greatly help in many engineering projects and are able to avoid serious errors during design and fabrication stages. The simulation of these models provides useful data and allows for the testing of different ideas that can help with the creation of more efficient and beneficial technologies for both companies and society in general.

CONTENTS

ETSEIB

Acknowledgements

I would like to express my very great appreciation to my supervisor and director Marc Cheah Mañé for his support, encouragement, and guidance during this work. You have my gratitude for the help provided and the opportunity to learn and enable me to research on this topic which I love.

My deepest gratitude to my parents, Pilar and Gianluca, who have supported me throughout my journey and have been able to give me the possibility of studying what I always dreamed of. Without them, this would have not been possible. They have been a guidance in my life and an example to follow for me.

List of Abbreviations

WT	Wind Turbine
MPPT	Maximum Power Point Tracking
DT	Drivetrain
FODE	First Order Differential Equation
MUX	Multiplexer
DEMUX	Demultiplexer

List of Figures

List of Tables

Preface

1.1. Origin of the project

Constant improvement and learning have always been aspects that define me as a person. My interest in technology and trying to find solutions for the disaster that we are starting to live with climate change have made me want to introduce myself into the world of renewable energies. I have always been fascinated by the wind turbines that I have seen physically, for their magnitude and apparent simplicity on the outside. They are very important tools and very useful nowadays. Therefore, the origin of this project comes from many years ago when I saw for the first time one of these turbines in operation and I realized that I would like to dedicate my professional career to it at some point in my life.

1.2. Motivation

This project has been inspired by my passion for renewable energies, with the intention of expanding my knowledge in the field and improving my skills when designing a model. I intend to understand more precisely how one of the technologies that today provides us with electricity, something we always take for granted. I also think that a deep understanding of this topic can help me understand other types of renewable energies and help me improve as an engineer in this industry.

1.3. Previous requirements

To tackle this project, it is important to know the basics of several branches of engineering, which I have been acquiring throughout my undergraduate studies.

It is necessary to have a basic understanding of aerodynamics and fluid mechanics to know how turbine blades move. It is also necessary knowledge on electrical

engineering and electrical machines as they are the basis of how it transforms and transports the energy captured from the wind. Knowledge on how to use mathematical programs, such as *MATLAB*, and more specifically *Simulink* to be able to design the turbine model and thus perform different simulations to study it in different environmental situations. It should be noted that without the help of a knowledge of system dynamics and control, it would not be feasible to make this wind turbine model.

2. Introduction

2.1. Aim and objectives of this project

The aim of this project is to create a wind turbine model on *Simulink*, in which only by introducing the velocity of the wind, it is able to determine the total power generated. The idea is to also implement an angle pitch controller to the turbine which acts on the turbine under certain conditions to ensure that at no time it is working in conditions that could damage the integrity of the wind turbine itself.

To achieve this purpose, it is necessary to fulfill a series of objectives:

- Deep research must be done on the operation of a wind turbine, how its components work and all the physics that governs their behavior.
- Learn how to make a model from scratch using a program as powerful as *MATLAB*; from the most basic functions to those with certain complexity.
- Be able to extract information of the dynamic operation of the turbine, as it would be done in a real case.

2.2. Extent of this project

To make the turbine model, three main aspects which are directly related to the aerodynamic part will be discussed in depth. These are the aerodynamic model of the system (wind incidence used to create a torque), the drive train and the pitch angle controller. Each of these will be explained in detail, from their fundamentals to their independent analysis once designed. At the end of the project, a wind model will be produced that will allow to analyze various aspects of the finished model (once the main models are attached) at various wind speeds and use it to study the behavior of the wind turbine.

The electrical part of the turbine will not be studied because it is irrelevant to the objective of this project. It would be an implementation that would not provide any relevant information to the subject of study, which in our case is the aerodynamic behavior of the wind turbine.

2.3. Importance of modeling

Models are very important in the world of engineering as they can provide various opportunities when designing or optimizing a process, tool, or machine to name a few. The model represents a schematization of the characteristics of reality, which greatly facilitates its investigation. It is essential that the model has a certain degree of structural and functional similarity with reality so that the studies carried out on the model can be extrapolated to the object or phenomenon under study.

Something very important to keep in mind is that the model being created should be simpler to study than the real object or phenomenon in question, as well as being operative. In this way, it can be studied with different disturbances to observe the consequences that it would have in the real world and thus be able to learn in depth new characteristics of the object itself. All this, without forgetting that the ability to make models helps economically to research and development, which, to this day, is crucial in all industries in the world.

It is very interesting to think that the same real phenomenon has many ways of being represented in the form of a model, which means that there is not only one way of investigating and it also helps us to understand other aspects that we might not have considered in the first place. In fact, it is important that there is a set of different models when representing reality in a theoretical way. This is because in each model we could be leaving out some features that are not necessary in some of the models to represent some functionality, but this feature may be an important factor in the real phenomenon. Therefore, using several models is always a good idea.

The model is interpreted in the scientific theory, and this theory establishes the very meaning of the variables, relationships, and constants of the analytical model. In

addition, this theory develops all the implicit properties and relationships of the model, hence they can be represented in an ideal and systematic way so that the dynamics and structure of the phenomenon can be studied.

All this leads us to the conclusion that modeling is of great importance since it allows us to construct, illustrate and optimize human theoretical-practical and evaluative activities. As a whole, it is an effective and economical resource to predict events and anticipate facts not observed to date.

2.4. Wind power generation

What are wind turbines? These devices vary in size and can transform the kinetic energy of the wind into electrical energy for any type of use. Turbines are increasingly used due to technological progress and cost reduction. However, it is not the only source of renewable energy, as it competes with solar panels which are becoming cheaper to manufacture due to the reduction in the price of the materials needed to manufacture them. The most distinctive feature between the two is that the energy created by wind turbines is normally used to cover peaks of demand in the electrical grid, while solar is usually a regular source of energy regardless of demand.

The installed wind power capacity worldwide is growing year by year, in fact in 2020 some 93 GW was added to the global capacity [2]. *Figure 2.1* [2] shows the global capacity:

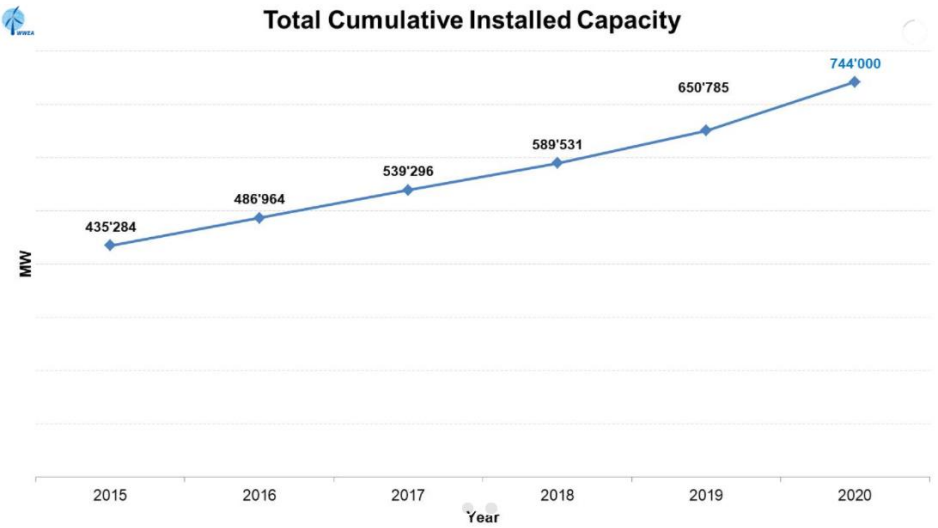

Figure 2.1: Worldwide installed wind power capacity

We can see in the previous figure how the total power capacity has been increasing in the last 6 years. In the year 2021 there will be a very large capacity, but how much does Spain represent in this amount?

Figure 2.2 [2] shows the 11 countries with the highest wind power capacity. Spain is in the top 5 with approximately 27,5 GW of power, which is very remarkable if we compare our population with those ahead of us in the table. Countries with more than 30 times the population of Spain, such as India or China, have only 10 times more capacity in the case of China. This is a remarkable fact as it demonstrates Spain's commitment to renewable energy sources. However, we must state that the growth of the Chinese with wind farms installation is exponential, and they represent alone a 39% of global capacity.

Country/Region	2020	New Capacity 2020	2019	2018
China*	290'000	52'000	237'029	209'529
United States	122'328	16'895	105'433	96'363
Germany	62'784	1'427	61'357	59'313
India	38'625	1'096	37'529	35'129
Spain	27'446	1'638	25'808	23'494
United Kingdom	24'167	652	23'515	20'743
France*	17'949	1303	16'646	15'313
Brazil	18'010	2'558	15'452	14'707
Canada	13'588	175	13'413	12'816
Italy*	10'850	280	10'512	9'958
Turkey	9'305	1'249	8'056	7'369
Rest of the World*	110'000	14'000	96'035	84'814
Total*	**744'000**	**93'000**	**650'785**	**589'547**

Figure 2.2: Top 11 countries with most wind power capacity in the last 3 years

The ratio between the energy generated in a wind farm during a period of time, and the energy that would have been generated by the wind farm operating at full load during that period, is known as the load factor, and it does not usually exceed 20% in wind farms. This means this technology cannot supply with guarantees a certain constant demand (base load). The energy produced by wind turbines is normally used to supply peaks of demand in the electrical grid.

If we now compare the electricity demand in 2019 in Spain, we will observe that almost half of the electrical energy (40%) comes from renewable sources and 22.6% of the total comes from wind turbine production alone. The data for 2019 are reflected in *Figure 2.3* [3]:

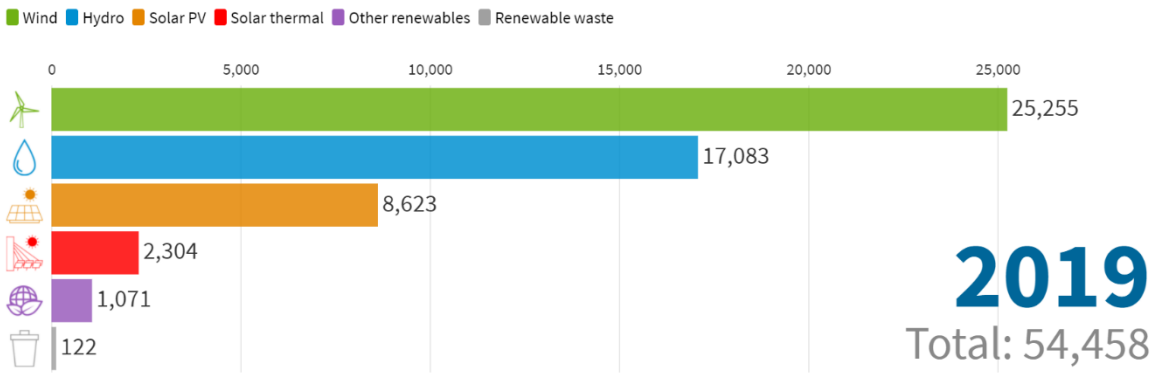

Figure 2.3: Contribution of renewable energies in Spain during the year 2019

With all this information we can get an idea of the importance of this technology. There is no doubt that they are very good sources of energy and will play a very important role in the near future to cover part of the world's demand for electricity. The evolution of the turbines will allow them to be manufactured at a lower cost and will provide us with energy sources that are much more beneficial to the environment and thus be able to stop using fossil fuels.

3. Aerodynamic Model

This chapter deals with the modeling and programming of a conventional wind turbine with adjustable blade pitch. The aim is to achieve a model that only needs 3 input variables, the wind speed (v_w), the angular velocity of the turbine shaft (ω_t) and the blade pitch angle (β). As outputs, the turbine torque (T_t) and the recalculation of the turbine angular velocity (ω_t) are obtained. It is important to note that as the model progresses, input variables such as ω_t and β will cease to be variables that we are able to control since they will be automated. This is due to the fact that we intend to create a model where, when a specific wind speed is received by the turbine, it adapts to the conditions to be able to offer electrical power without ever exceeding the nominal value.

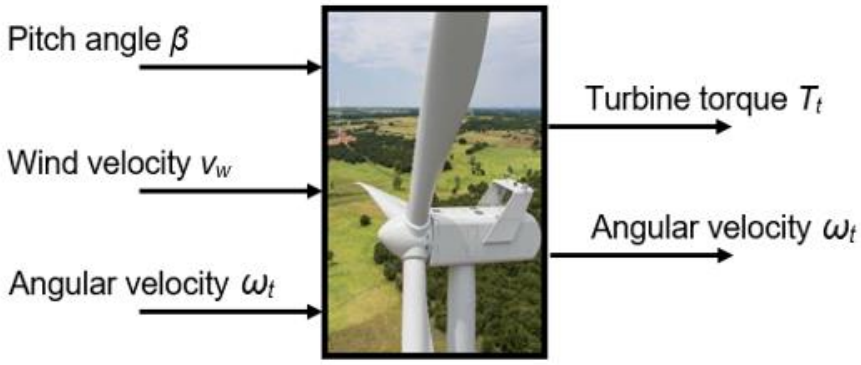

Figure 3.1: Block diagram for the aerodynamic model

3.1. Fundamentals of a wind turbine

Fundamentally, nowadays, the most common wind turbines are built with 3 blades fixed on a structure called hub, and the latter is connected to what is called nacelle. Inside the nacelle we find the mechanical and electrical parts such as the gearbox, multipliers, a generator, brakes... All this is held by the tower.

3.1.1. Classification

There are 2 different types of wind turbines: vertical axis and horizontal axis. In the past, wind energy was converted directly into mechanical energy and in some cases, there are still turbines that do this. However, due to the high demand for electricity in all the world populations, it is common to find wind turbines that directly convert this wind energy into electricity by means of a generator.

Rotors with vertical axis

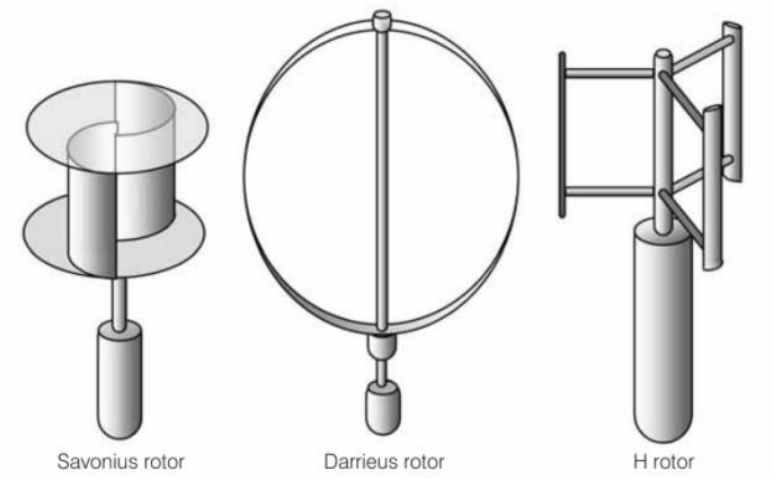

Figure 3.2: Rotors with vertical axis

These rotors are ordered from oldest to newest. They are rotors that work especially well with abrupt changes in wind direction and do not have to face the wind direction. This is a great advantage, but the problem with these systems is their low efficiency compared to more modern models of horizontal axis rotors. The best of these three types of vertical rotors is the H-rotor, which in fact was designed for extreme climates existing in high mountains or in Antarctica [4]. It is for these reasons that this type of rotor is only used on special occasions, and its manufacture is very costly in terms of material requirements. It should be noted that they still have certain advantages compared to horizontal rotors. Their structure and assembly are simple, and the generator and its electrical components can be placed on the ground, which greatly simplifies the maintenance of the turbine.

Rotors with horizontal axis

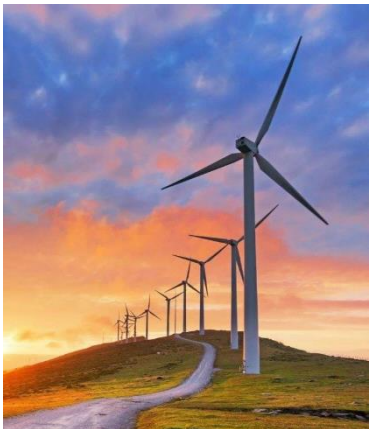

Figure 3.3: Rotors with horizontal axis

Despite certain advantages of vertical axis rotors, they have not played a major role in being used as standard wind turbines in the industry. Most wind turbines are horizontal axis due to crucial factors such as their higher efficiency and lower resource utilization in manufacturing [4].

Over the years, wind turbines have been manufactured larger and larger, which allows to generate a greater amount of electrical energy. We can see the evolution in *Figure 3.4* (M. Cheah, personal communication, September 29, 2021):

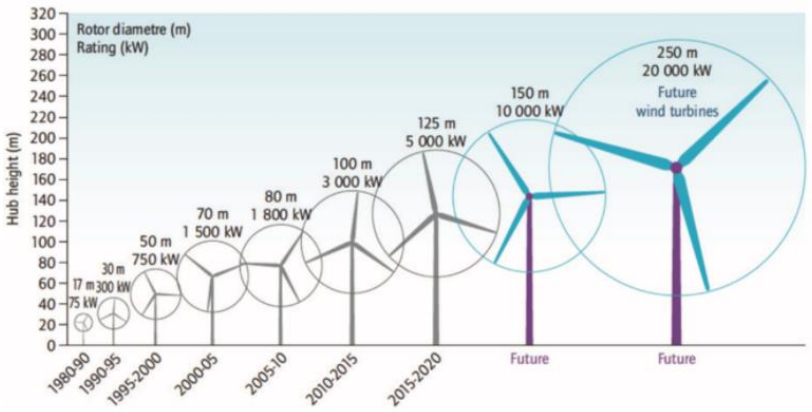

Figure 3.4: Evolution of wind turbine power generation

Most modern turbines are built with one, two or three blades. Due to the amount of material needed to manufacture the blades, horizontal rotors with more than 3 blades are not usually used. But, if that is the case, then why not use a single blade? The

problem with single-bladed rotors is that they need a counterweight on the opposite side to have a smooth movement and therefore suffer a very high stress on the material.

The optimum power coefficient (C_p), which we will talk about later, is higher in three-bladed rotors than in two-bladed rotors. In addition, they integrate visually better with the landscape as they appear to have a smoother motion than those with two blades.

3.1.2. Parts of a turbine

As mentioned above, the wind turbine is made up of four basic components. The blades, the hub, the nacelle and the tower as we can see in the following figure [4]:

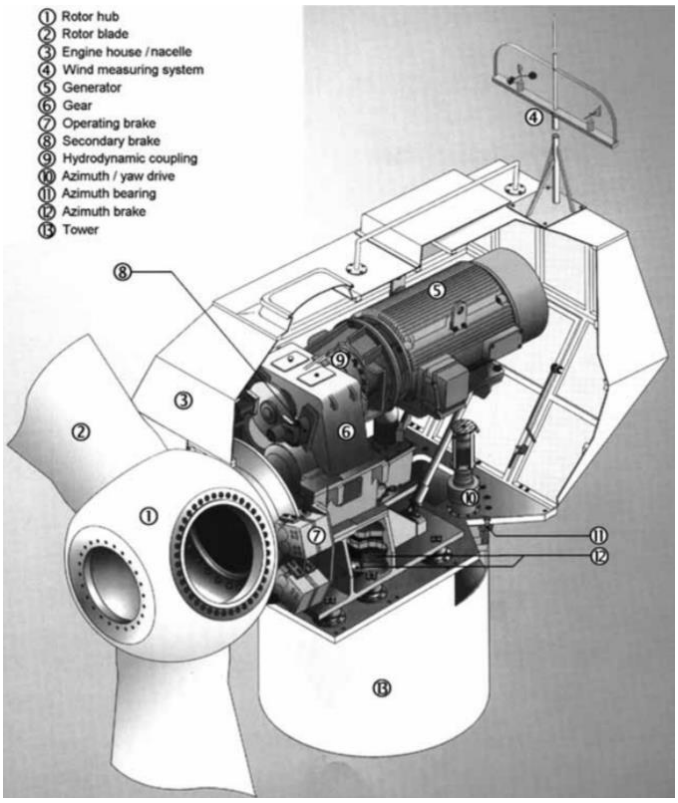

Figure 3.5: Assembly of a standard wind turbine

The hub supports the blades and the blade pitch change system, which will be modeled later in this paper. The hub is supported by the nacelle which contains most of the parts of a wind turbine. Inside you can see the gearbox, the multiplier, a pair of brakes and the electric generator among other things. The electric generator will not

be modeled in this work, but it is important to note that there must be a match between it and the other components. That is, it must be a generator according to the performance of the other components and the torque that the turbine blades will generate. The blade pitch change will play a very important role since it will be the system that will allow us to vary the electrical power generated by the turbine. This is crucial both for not producing too much energy that is not needed and for not damaging the turbine (for example, if there is a storm, the turbine must be able to stop its operation in order not to be damaged by the strong winds if it is not prepared for them).

3.2. Turbine modelling

3.2.1. Equations and concepts

According to [4], the generation of a wind turbine is based on the extraction of power from the kinetic energy of the wind. Therefore, we can express that the power generated by the turbine P_t, as the kinetic power available in the gust of air passing through the area swept by its blades (P_w), all multiplied by a dimensionless constant called power coefficient C_p.

C_p is then defined as the ratio of the power used by the turbine P_t to the power contained in the wind P_w. It is commonly understood that C_p is a measure of the aerodynamic efficiency of the turbine. This efficiency depends on several factors such as the average air speed across the area covered by the blades, the angular velocity, and the geometry of the turbine. The latter is highly dependent on the blade pitch angle. The expression for the power output of the turbine is then:

$$P_t = C_p * \frac{1}{2} * \rho * A * v_w{}^3 \quad (1)$$

Where:

ρ is the air density (assumed constant at all times)

A is the area swept by the turbine blades

v_w is the average speed of the wind

There are many ways to represent C_p as a model, from the consideration that it is a steady state constant to representing it with tables based on measured data. However, in this paper we will make use of a common analytical expression suggested in [5]:

$$C_p(\Lambda,\beta) = c_1\left(c_2\frac{1}{\Lambda} - c_3\beta - c_4\beta^{c_5} - c_6\right)e^{-c_7\frac{1}{\Lambda}} \quad (2)$$

Where [c_1....c_9] are characteristic parameters of the turbine. These parameters are obtained by statistical analysis of measured data from a real turbine and finite element simulations. To focus on the modeling of the work, we will take these parameters from [6], but more information about this will be displayed later in the work. The symbol β blade pitch angle, and Λ is defined as:

$$\frac{1}{\Lambda} \triangleq \frac{1}{\lambda + c_8\beta} - \frac{c_9}{1+\beta^3} \quad (3)$$

Where λ is the tip speed ratio defined as:

$$\lambda \triangleq \frac{\omega_t R}{v_w} \quad (4)$$

Where w_t is the turbine speed and R is the radius of the turbine (length of a blade).

The tip speed ratio is extremely important for the design of a wind turbine. It is the ratio between wind speed and the blade tip speed [7]. This concept is key because if the turbine rotor were to rotate too slow, for example, most of the wind would pass through the space between the blades and it would not be able to generate much power. On the other hand, if the rotor were to rotate too fast, the turbine would act as if it were a solid wall for the wind. In addition, the turbine blades create turbulence as they spin,

so if the next blade comes in too fast, it will be hitting turbulent air which is neither good for the purpose of power creation nor for the turbine structure and materials which could be damaged. Therefore, turbines must be designed with optimum tip speed ratios to obtain the maximum possible power from the wind.

To better understand the concept of the power coefficient C_p, it will be further explored by analyzing whether there can be a maximum power coefficient. As explained above, C_p is a measure of the power extraction efficiency of the wind. Therefore, we know that its value should be between 0 and 1. It is clear that a C_p of 1 would be impossible since we would be saying that we would have a perfect turbine, without any losses, however, theoretically, that range of values would be possible.

Betz calculated in 1926 [4] the maximum possible power coefficient, which was defined as the ideal or Betz power coefficient $C_{p,Betz}$. The wind speed ratio is defined as the wind speed after passing through the blades (v_2) divided by the speed before passing through the blades (v_1):

$$\xi = \frac{v_2}{v_1} \tag{5}$$

Through calculations, it is obtained that the ideal wind speed ratio is 1/3, that is to say, two thirds of the wind speed would be lost as it passes through the turbine. This suggests that the turbine could ideally harness the kinetic energy of 2/3 of the wind passing through it. In fact, it is demonstrated [4]:

$$C_p = \frac{P_t}{P_w} = \frac{(v_1+v_2)*({v_1}^2-{v_2}^2)}{2*{v_1}^3} = \frac{1}{2} * \left(1 + \frac{v_2}{v_1}\right) * \left(1 - \frac{{v_2}^2}{{v_1}^2}\right) \tag{6}$$

Hence, if we introduce the ideal wind speed ratio (1/3):

$$C_{p,Betz} = \frac{16}{27} \approx 0{,}593 \tag{7}$$

It indicates that the theoretical maximum power that can be extracted from the wind is about 60%. In *Figure 3.6* it is observed how the C_p varies against the ratio of wind velocities and the optimum coefficient is displayed [8]:

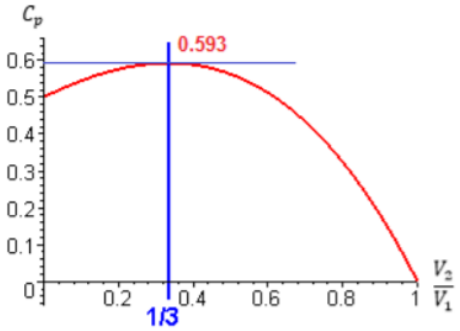

Figure 3.6: Power coefficient against ratio of wind velocities.

Although real wind turbines do not reach the theoretical optimum calculated by Betz, actual systems can achieve power coefficients between 0,4 and 0,5. The efficiency of a wind turbine system can be defined as the division between its C_p and the $C_{p,\text{Betz}}$:

$$\eta = \frac{C_p}{C_{p,Betz}} \tag{8}$$

Equations (2) and (3) mentioned the appearance of the power coefficients. These coefficients come from several finite element simulations and experimental data with real turbines, however an interesting but not complex graph that can be shown is how the C_p behaves against λ (for pitch angle 0) [*Appendix A*]:

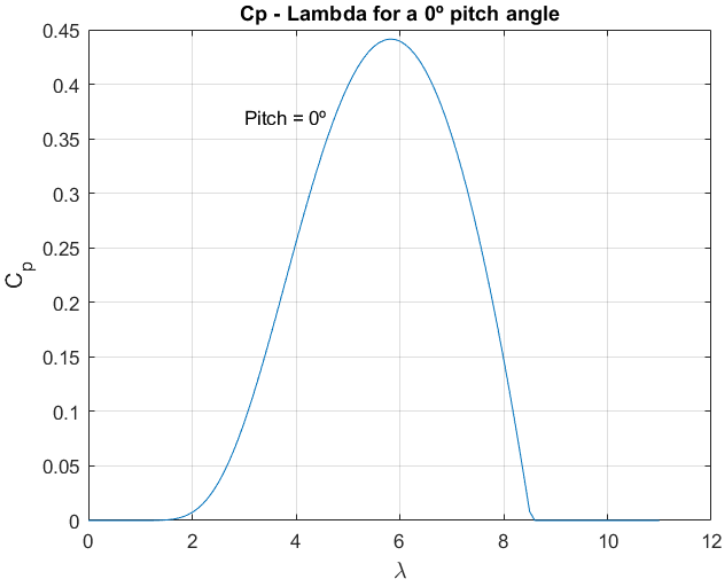

Figure 3.7: Evolution of the power coefficient against tip speed ratio for a 0 degree pitch angle

A representation of the same graph but with different pitch angles [Appendix A]:

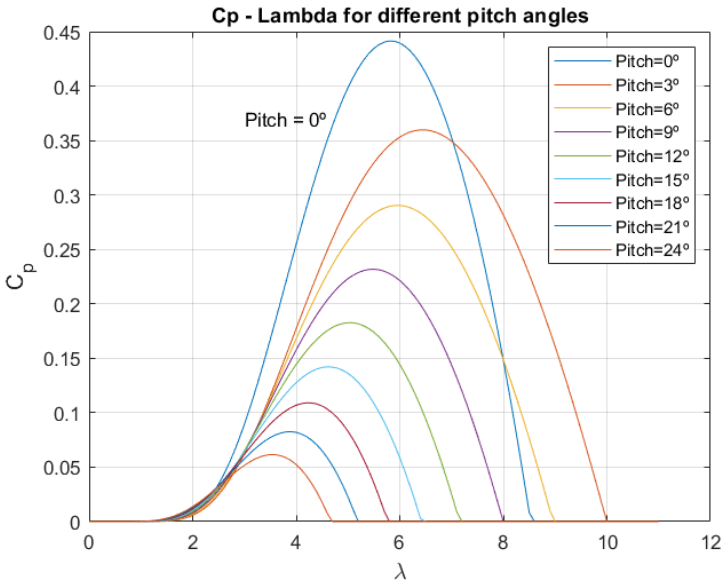

Figure 3.8: Representation of Figure 3.7 with pitch angles from 0º to 21º

Finally, there is need for an equation relating all the previous concepts with torque, which is the outbound variable of the aerodynamic model. Equation (1) related all the concepts previously explained to provide the power extracted. From basic physics, it is known that torque is power divided by an angular velocity, hence, if we divide equation (1) by the angular velocity of the wind turbine axis, we obtain the torque generated by the turbine:

$$\Gamma_t = C_p * \frac{1}{2} * \rho * A * {v_w}^3 * \frac{1}{\omega_t} \quad (9)$$

With equation (9) and (1), we can plot graphs relating turbine torque and power, to the angular velocity. Using all the data from our own turbine and observing the evolution with various wind speeds, we obtain *Figures 3.9* and *3.10 [Appendix B]*. The trend is clear, the higher the wind, the higher the torque and power, and it is observed that the maximum point in higher winds is found in larger omegas. The turbine power at different wind speeds can also be plotted (*Figure 3.11 [Appendix C]*). It is limited in the nominal power value of our turbine (727,3 kW) which is represented by the continuation of a horizontal line in the graph itself.

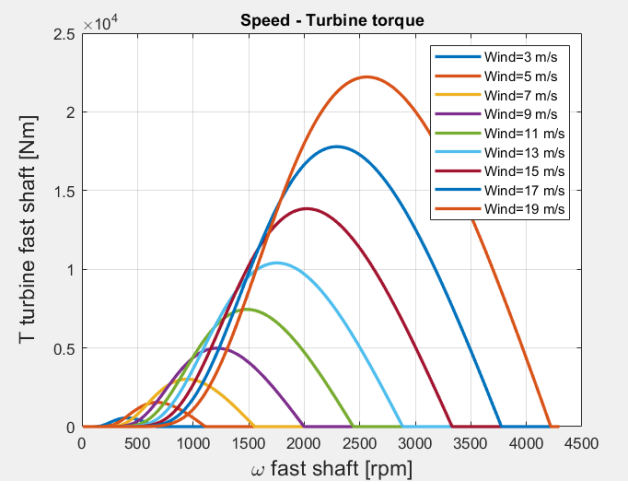

Figure 3.9: Turbine torque against its omega at different wind velocities

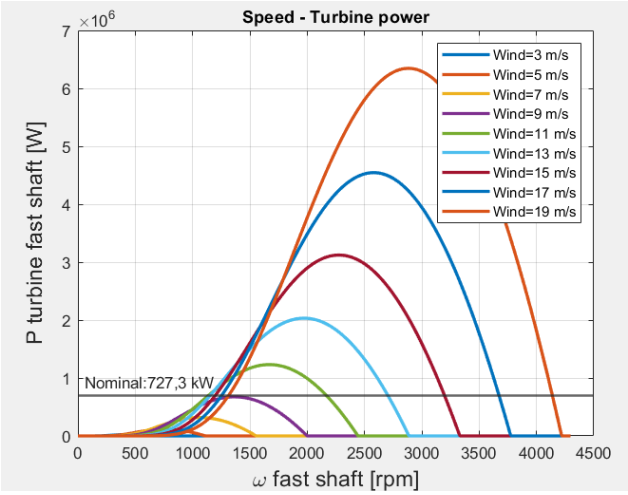

Figure 3.10: Turbine power against its omega at different wind velocities

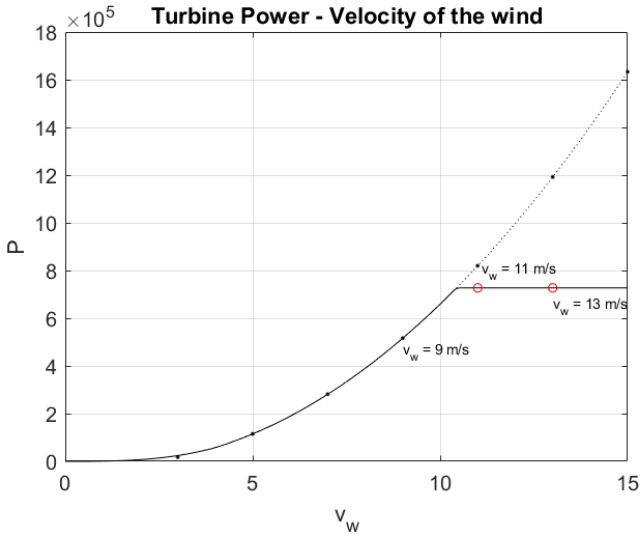

Figure 3.11: Turbine power against different wind velocities and limited by the nominal value

3.2.2. Simulink model

This section shows the implementation to Simulink of all the concepts previously explained about the aerodynamic model. First, all the data of the turbine to be modeled in this work must be detailed in *Table 1* [5]:

Table 1: Wind turbine parameters

Parameter	Value	Units	Description
R	33	m	Turbine radius
v_w^N	9,55	m/s	Nominal wind speed
ρ	1,225	Kg/m^3	Air density
c_1	1		
c_2	110		
c_3	0,4		
c_4	0,002		
c_5	2,2		Cp function parameters
c_6	9,6		
c_7	18,4		
c_8	0,02		
c_9	0,03		
λ^{opt}	5,83	1	
C_p^{opt}	0,442	1	
ω_t^N	22,83	min^{-1}	Wind turbine nominal speed
Γ_t^N	0,320e6	Nm	Wind turbine nominal torque
P_t^N	727,3	kW	Wind turbine nominal power

The maximum efficiency of the turbine is calculated through the differentiation of the equation (2) as a function of λ and then finding the roots of the equation. To simplify calculations, the pitch angle will be taken as zero. Taking beta as zero can be done since an increase of this latter one will always decrease the force that the wind applies on the turbine blades. Therefore, λ^{opt} *and* C_p^{opt} will be:

$$C_p{}^{opt} = \frac{c_1 c_2}{c_7} e^{-\frac{c_2 + c_6 c_7}{c_2}} \tag{10}$$

$$\lambda^{opt} = \frac{1}{c_9 + \frac{c_6}{c_2} + \frac{1}{c_7}} \tag{11}$$

At first, the value of the pitch angle will be taken as zero:

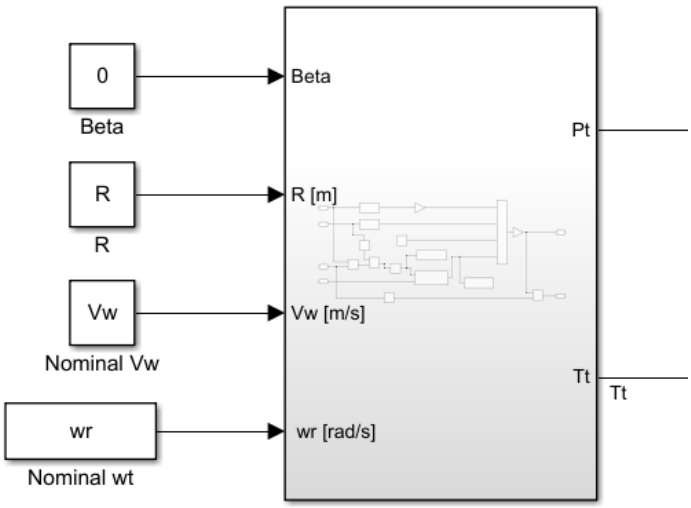

Figure 3.12: General view of the aerodynamic model in Simulink

In *Figure 3.12* there is an overview of the aerodynamic model which had been also displayed conceptually at the beginning of this chapter in *Figure 3.1*. In the next figure there is a detailed visual of the inside *Simulink* model.

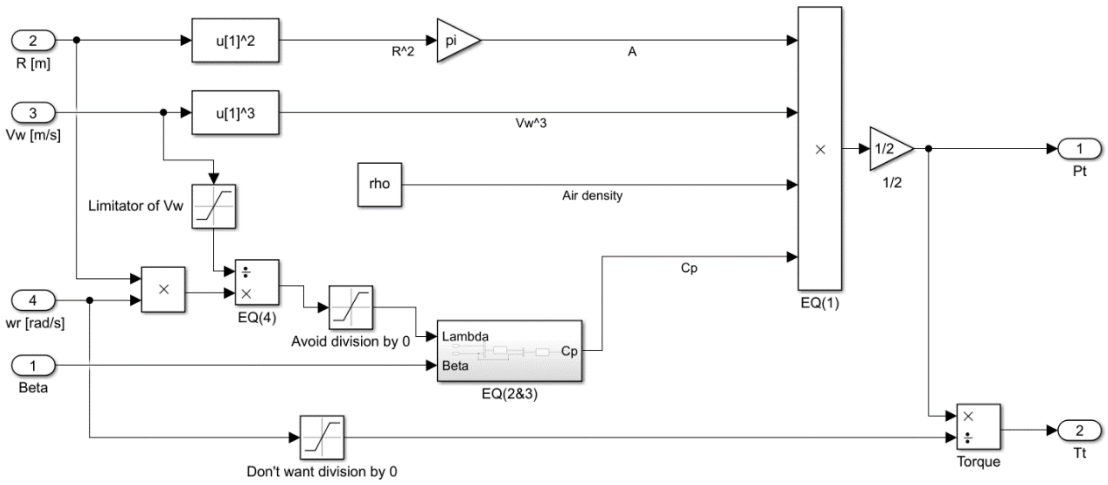

Figure 3.13: Inside the aerodynamic model in Simulink

All the equations that have been discussed in this chapter are written in the form of blocks and can be seen in *Figure 3.13*. It is important to note that some saturators have been placed in some parts to make sure no divisions by 0 are done in the calculations. There is also a limit for the wind velocity, which could be interpreted as a brake for the wind turbine since they should not be working at high speeds (to avoid

any damages). There is also a subsystem which is displayed in detail in *Figure 3.14.* It is the block diagram for the equations (3) and (2) which give out the value of C_p. There has also been placed a saturation of C_p to avoid negative values.

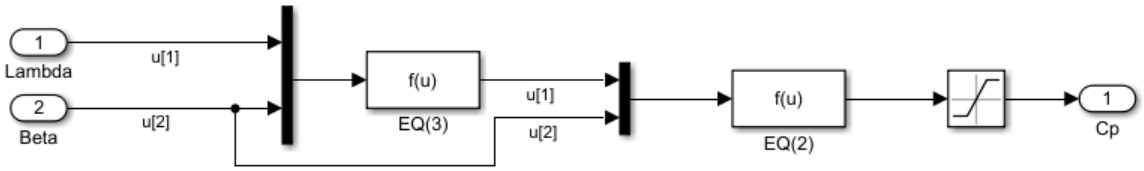

Figure 3.14: Details of the subsystem which calculates the power coefficient

With this model design, it is possible to obtain different turbine powers by varying base variables such as the velocity of the wind and pitch angle of the blades. The angular velocity of the turbine is, at the moment, a variable, however, later on by adding the rest of the model, this variable will be intrinsic and will be dependent on the other variables. The blade pitch is a variable for now, even though, with the design of the pitch controller later in the work this will be also an internal value instead of a variable. The idea is to set an amount of electric power needed to be generated by the turbine, and the turbine will adapt to achieve that power by changing the blade pitch angle (always with a constant velocity of the wind).

3.3. Proof of the aerodynamic model

In this section we will try to understand if the aerodynamic part of the wind turbine has been correctly modeled in *Simulink*. For this purpose, the evolution of different values that exist in the model will be observed, always with different wind speeds. To do this, we have used a block of *Simulink* called *Repeating Sequence Stair* (*Figure 3.15*). It simply creates a signal that simulates an ascending stair. These are step signals that operate in sequence and where the next step always starts above its predecessor.

Specifically, steps are made from a speed of 2 m/s up to 32 m/s with 3 m/s increments. The step time is of 15 seconds. The result can be seen in the graph in *Figure 3.16*.

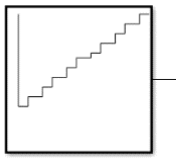

Figure 3.15: Simulink block named Repeating Sequence Stair

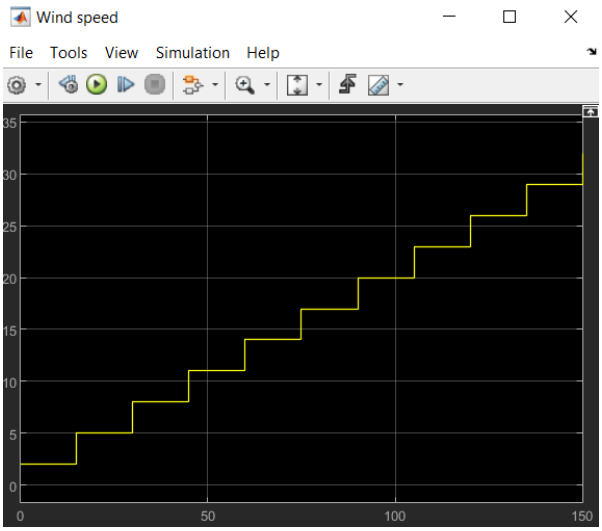

Figure 3.16: The increments in the velocity of the wind

The variables to be analyzed are the turbine torque, the power generated by the turbine, the value of C_p and the value of the tip speed ratio. The results are shown in *Figure 3.17*. It is observed that for low wind velocities, the value of lambda is very high, which makes sense considering equation (4). The lower the wind speed, the larger the division result. Moreover, it is observed that when the nominal value of v_w (9,55 m/s) starts, the λ reaches stationary value faster (at its optimal value (*Table 1*)). In fact, it hardly varies its trajectory for velocities higher than the nominal one, therefore, this factor behaves correctly.

If we now observe the value of the power coefficient, we can also understand that for low speeds (lower than the nominal speed), its value takes a long time to stabilize and increases enormously. On the other hand, at values of v_w greater than nominal, the C_p manages to reach the stationary value (the optimum value (*Table 1*)) faster and with smaller jumps. Therefore, it can also be said that this value is correct.

As for the power and mechanical torque values of the turbine, their trajectories are correct. As the wind speed increases, their values increase and reach the steady state smoothly.

With all this information, it is then possible to say that the aerodynamic model designed in *Simulink* is validated and correct.

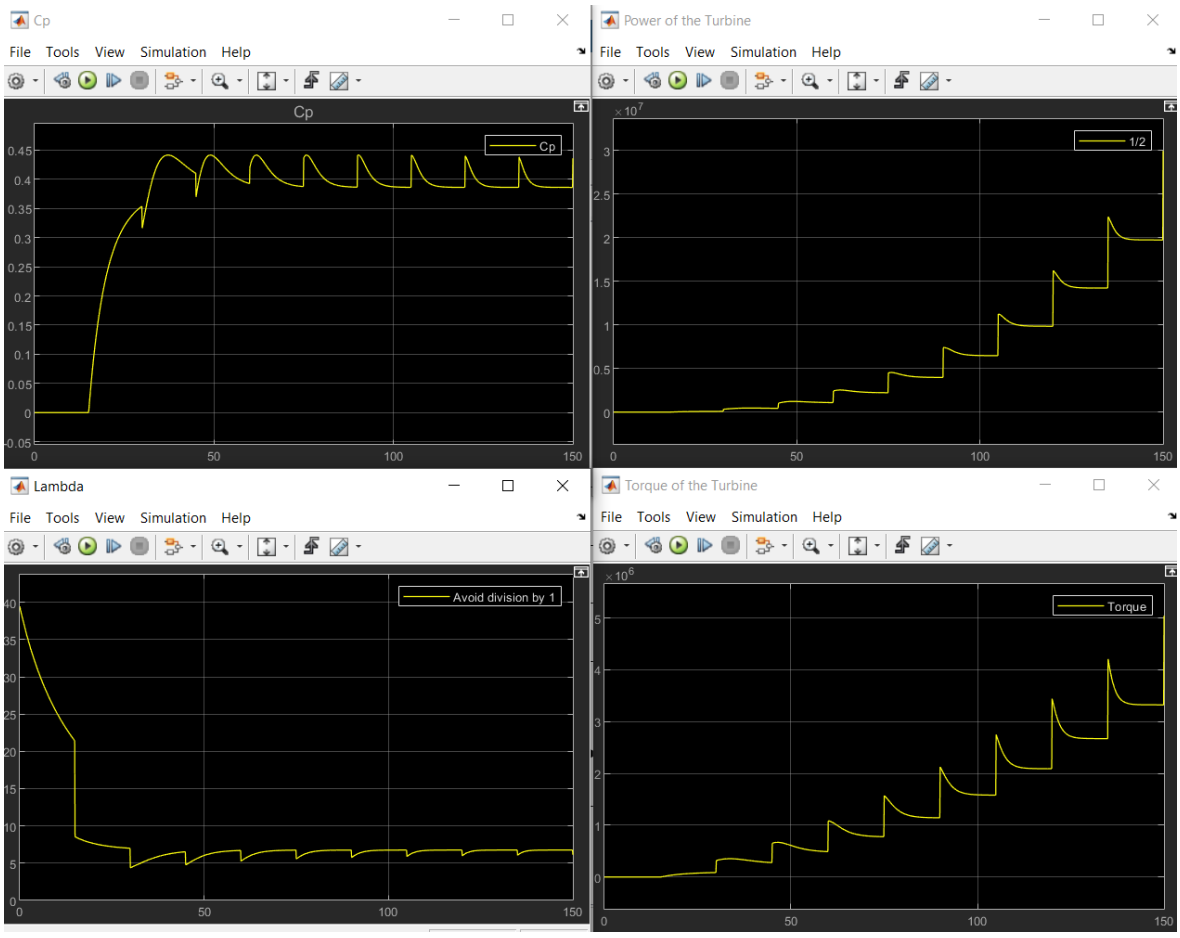

Figure 3.17: Analysis of C_p, λ, P_t and T_t of the aerodynamic model for different values of the velocity of the wind

4. Drivetrain of the wind turbine

The term mechanical drivetrain encompasses everything that has rotating parts, that includes parts from the rotor hub to the electrical generator. The drivetrain (DT) is an indispensable part of a wind turbine as it is responsible for converting all the rotational energy of the three turbine blades into electrical power.

4.1. Mechanical transmission system

It is a very complex part not only in wind turbines but in all systems that contain one, or even more than one. It should be noted that they do not always have to be designed in the same way, since depending on the system we are modelling there will be different factors to consider.

Classical models of concentrated parameters are insufficient to describe precisely and accurately all the transient phenomena that are produced in the most modern wind turbines; however, they are often helpful to obtain first results in design analysis [8]. Why is it so complicated? Basically, speaking in terms of wind turbines, significant influences such as elasticity and flexibility of the blades and other elements, for example rotating shafts, should be brought to light. If you want to include these influences, you need to represent them with much more detailed models, which are extremely difficult in many cases.

One way to simplify, is usually to replace the classical model by several sub models that are coupled together. In this way it is possible to decompose the very complex system into its main structural components and represent each one with its own model, but always using concentrated parameters. A typical wind turbine DT is shown in detail in *Figure 4.1* [10]:

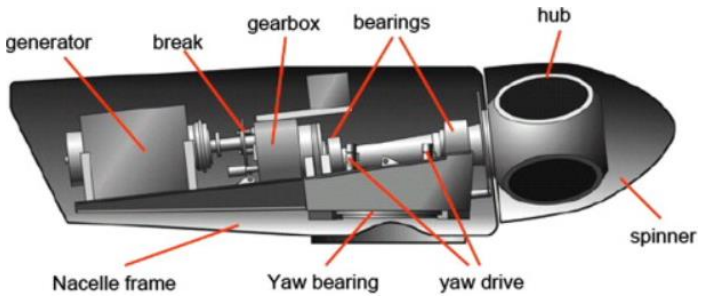

Figure 4.1: Detailed general drivetrain of a wind turbine

The drivetrain is composed of several elements, all of which have a specific job, and they must work in harmony because if one of them fails, the rest of the components will not be able to continue working; each part is important. Except for the direct DT, all of them have a gearbox. This element allows us to transfer torque loads from a primary motion to a rotational output, and usually, this is done with a different angular velocity ratio [11].

Horizontal shaft DTs also contain components such as the hub, the main shaft and main bearing, brakes, generator shaft and a generator. All these components should always be considered together as they form a single functional unit. From here, there are several ways to configure drivetrains.

4.1.1. Drivetrain model classification

It should be noted that between some of the models presented below the difference in complexity is very high and the results obtained between them hardly differ. That is why it is not always necessary to use very detailed and accurate models; simpler models provide us with results very similar to reality, which is demonstrated in several published articles [12]. Not all existing models are represented in this work.

Six-mass model

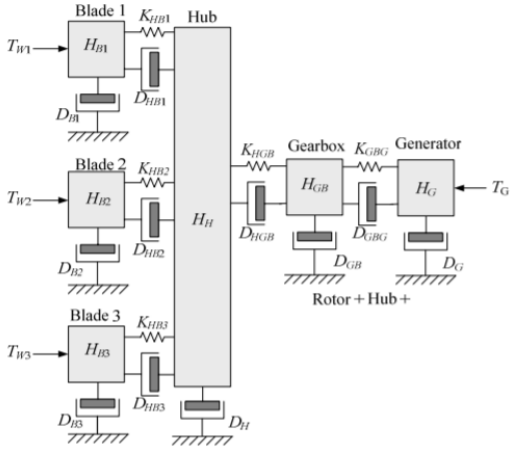

Figure 4.2: Schematic diagram of a 6-mass model drivetrain

The diagram shown in *Figure 4.2* [11] shows a basic 6-mass drivetrain model. In this model it can be distinguished that the three turbine blades, the hub, the gearbox, and the generator are defined as masses: six elements. Each of these elements has its own inertia and therefore also its own angular velocity. In addition, it should be noted that stiffnesses are defined at each joint as well as damping, both at the joints and at their own masses.

This model is hardly applicable in most power systems due to the complexity involved in defining each mass separately and all its independent phenomena.

Three-mass model

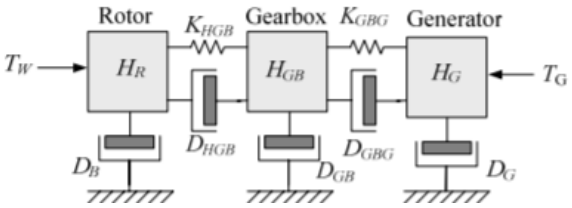

Figure 4.3: Schematic diagram of a 3-mass model drivetrain

On this occasion, it is clearly visible on *Figure 4.3* [11] that, compared to the six-mass model, this model is considerably simpler. Three masses are defined: a rotor, a

gearbox, and a generator, all elastically coupled. Normally the low-speed shaft (the coupling between the rotor and the gearbox) is the most flexible and elastic component (it has a small stiffness component). On the other hand, the blades and the generator are at the opposite extreme as their stiffness components are the highest and also have the largest inertia [8].

Two-mass model

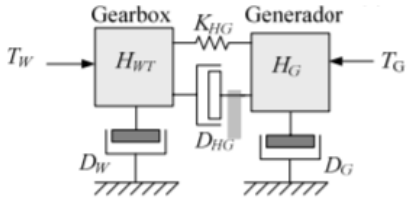

Figure 4.4: Schematic diagram of a 3-mass model drivetrain

This model is usually the most used in research papers on WT modeling, mainly due to its good behavior and similarity to the three-mass model. In this case, the model parameters are concentrated on the entire mechanical system attached to the low-speed shaft. The components represented are the gearbox and the generator. We could interpret the mass on the left of *Figure 4.4* [11] as representing the blades and hub sub models, while the mass on the right would represent the high-speed shaft, the gearbox, and the generator rotor.

Turbine blades are usually made of composite materials which are so light in weight that they make them feasible choice for building large turbines. Due to the strong forces applied to the blades by the wind, they tend to deform, thus introducing dynamic effects that should be considered in the transmission of power through the turbine. A common way to model this phenomenon is by solving differential equations, however, it is so computationally intensive that it is common to use approximate parameters for the simulation of these systems. One way to approximate is to treat the mechanical system as masses in series connected through elastic couplings with linear stiffness, a damping ratio, and a multiplication ratio [5]. The solution can be seen visually in *Figure 4.5*.

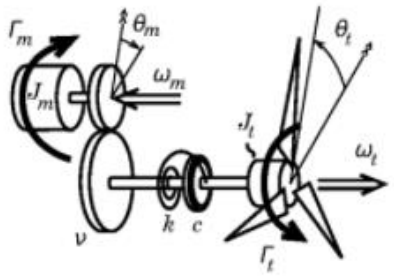

Figure 4.5: Two-mass model visual representation

In the model shown in *Figure 4.5* [11], the part of the blades and the low-speed shaft is shown on the right side, while the part of the high-speed shaft is shown on the left side. Each of the masses has its own inertia J_t and J_m and its own angular velocity ω_t and ω_m. In addition, the stiffness k and the damping c are represented at the coupling and the gearbox ratio is represented by the parameter v.

One-mass model

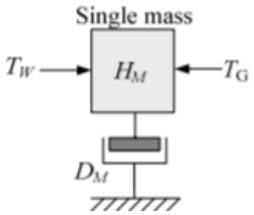

Figure 4.6: One-mass model visual representation

In this case, *Figure 4.6* [11], all mechanical components of the wind turbine are concentrated in a single mass. Here there is only a single inertia and a single damping coefficient. Having only one degree of freedom, only one angular velocity is applied, comprising the blades, the gearbox, the hub, and the generator rotor. In this model we lose the connections between joints, and therefore there are no elastic constants and mutual damping, in fact this model is considered as an ideal rigid solid.

It is a model that derives directly from the two-mass model which will be studied in the next section. Calculations will be made as a two-mass model drivetrain, and it will then be assumed as rigid to see the one-mass model. After that, the two-mass model will be implemented on the model and the differences between a one-mass and a two-mass DT will be observed. The rest of the work will use this latter one.

4.1.2. Equations and concepts

Having explained the two-mass drivetrain model, it is possible to introduce the equations and concepts that govern it. It is seen that this model, as shown in *Figure 4.5*, has an inertia for each mass and an angular velocity for each. The coupling between the two masses makes for the appearance of a stiffness coefficient (*k*) and a damping coefficient (*c*). Everything leads to obtain a torque on each side of the system, always considering that they are of opposite sign, and that the transformation ratio of the gearbox is of value *v*. Applying Newton's law to such a system, the following dynamic equations are obtained [5]:

$$\begin{bmatrix} \dot{\omega}_m \\ \dot{\omega}_t \\ \omega_m \\ \omega_t \end{bmatrix} = \begin{bmatrix} \frac{-c}{v^2 J_m} & \frac{c}{v\,J_m} & \frac{-k}{v^2 J_m} & \frac{k}{v\,J_m} \\ \frac{c}{v\,J_t} & -\frac{c}{J_t} & \frac{k}{v\,J_t} & -\frac{k}{J_t} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} \omega_m \\ \omega_t \\ \theta_m \\ \theta_t \end{bmatrix} + \begin{bmatrix} \frac{1}{J_m} & 0 \\ 0 & \frac{1}{J_t} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} * \begin{bmatrix} \Gamma_m \\ \Gamma_t \end{bmatrix} \tag{12}$$

$$[\omega_t] = [0 \quad 1 \quad 0 \quad 0] * \begin{bmatrix} \omega_m \\ \omega_t \\ \theta_m \\ \theta_t \end{bmatrix} + [0 \quad 0] * \begin{bmatrix} \Gamma_m \\ \Gamma_t \end{bmatrix}$$

Where Θ_t and Θ_m are the angles of the turbine and generator respectively, ω_t and ω_m are the turbine and generator speeds, Γ_t and Γ_m are the torque of the turbine and generator.

The matrix equation (12) leads to obtain 4 different equations, but what is intended to study in this case are the dynamic characteristics of the system. For this, it is possible to apply the Laplace transform:

$$\omega_t(s) = \frac{cvs + kv}{J_m J_t v^2 s^3 + (J_m c v^2 + J_t c)s^2 + (J_m k v^2 + J_t k)s}\Gamma_m(s) + \frac{J_m v^2 s^2 + cs + k}{J_m J_t v^2 s^3 + (J_m c v^2 + J_t c)s^2 + (J_m k v^2 + J_t k)s}\Gamma_t(s) \tag{13}$$

It was introduced in the previous section that the two-mass model can be made rigid to make it behave like the one-mass model. To say that it is rigid, it is simply necessary to take the value of *c* and *k* as infinity. If these values are introduced into equation (13), a much-simplified equation appears:

$$\omega_t(s) = \frac{v}{(J_m v^2 + J_t)s}\Gamma_m(s) + \frac{1}{(J_m v^2 + J_t)s}\Gamma_t(s) = \frac{v}{J_g s}\Gamma_m(s) + \frac{1}{J_g s}\Gamma_t(s) \quad (14)$$

The J_g appearing on equation (14) refers to the aggregated inertia of the totality of the mechanical system. Therefore, J_g is represented through this formula:

$$J_g = (J_m v^2 + J_t) \quad (15)$$

The values for the variables can be seen in *Table 2*:

Table 2: Drivetrain parameters

Parameter	Value	Units	Description
v	90	1	Gearbox transform ratio
J_t	3,6e6	kg m^2	Two-mass aggregated turbine inertia
J_m	49,38	kg m^2	Two-mass aggregated generator inertia
J_g	4e6	kg m^2	One-mass aggregated turbine inertia
c	10^6	Nm rad^{-1}	Two-mass damping ratio
k	6e7	Nm s rad^{-1}	Two-mass stifness coefficient
ω_m^N	2054,7	min^{-1}	Nominal angular velocity of the generator
Γ_m^N	0,320e6	Nm	Nominal torque of the generator

At the beginning of this work, it was explained that the factor which most affects the power extraction by the wind turbine is the average wind that hits its blades. That is why there are theoretical maximum values of power that can be extracted by a particular turbine. It is somewhat complex to measure the wind speed because in practice there are many variations in the wind and therefore the speed is never continuously the same. There are control methods to make the turbine always extract the maximum power without the need for wind measurement (MPPT is a common and

accurate one). A very basic but well-known technique is the constant tip speed ratio [13]. It is typically used for power quality simulations on wind turbines.

The control action for this basic method is:

$$\Gamma_m{}^* = \frac{1}{v} K_{Cp} \omega_t{}^2 \tag{16}$$

$\Gamma_m{}^*$ is the generator torque reference value and where K_{Cp} is:

$$K_{Cp} = \frac{1}{2} \rho A R^3 \frac{c_1 (c_2 + c_6 c_7)^3 e^{-\frac{c_2 + c_6 c_7}{c_2}}}{c_2{}^2 c_7{}^4} \tag{17}$$

It is important to note that the speed control law and the aerodynamic curve of the wind turbine are nonlinear. It is possible to make a small linearization of the equations while operating around the nominal point. Hence, we can consider the control as perfect:

$$\Gamma_m{}^* = \Gamma_m \tag{18}$$

4.2. Simulink model for a one-mass drivetrain

Now that the fundamental parts of the DT and all the formulas governing its system are known, we will proceed to show the model designed in *MATLAB Simulink*. Going back to section 3.2.2, *Figure 3.12* shows how the value of ω_r is a constant and of nominal value (*Table 1*). With the drivetrain implementation, the idea is to try to connect the aerodynamic model to the mechanical system and create a loop of the ω_r to recalculate as time progresses. Clearly, at t = 0 seconds, the value of ω_r is not defined and therefore we will have to give an initial value for the system to start working at that point. *Figure 4.7* shows the overall system interconnecting the mechanical model to the aerodynamic model and the rotor angular velocity looped (notice the blade pitch angle is still set to 0 degrees):

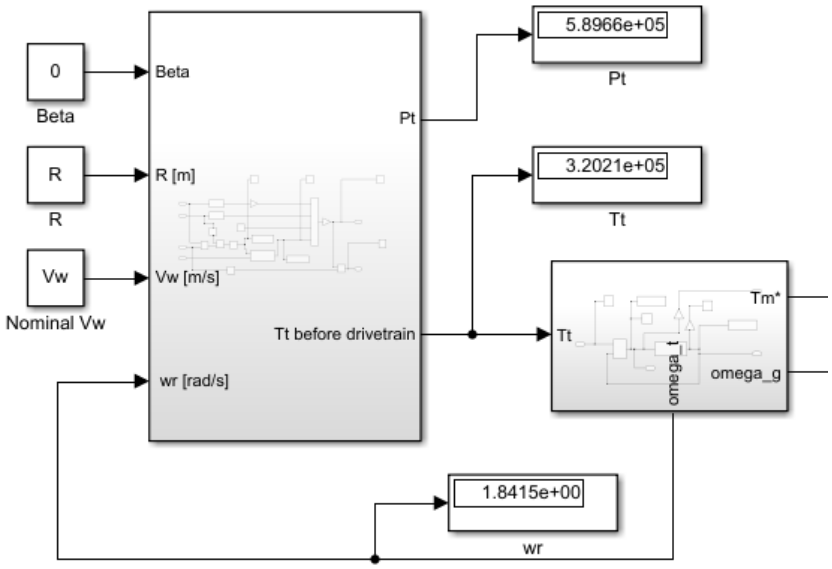

Figure 4.7: Aerodynamic and mechanical model connected on Simulink

The rotor omega is fed back from the model part of the drivetrain. Its value is in *rad/s* and is different from the nominal value. This is because as we have pointed out before, the system only takes the nominal value as the initial value, then the system recalculates itself until, in this case, with a wind speed of *9 m/s*, it stabilizes at *1,84 rad/s*. *Figure 4.8* shows the interior of the drivetrain subsystem:

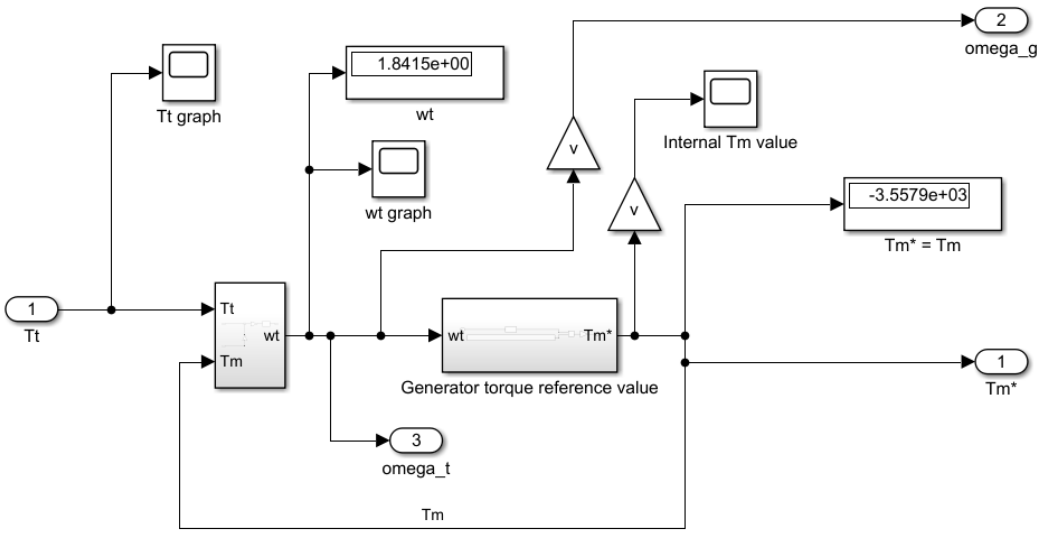

Figure 4.8: The drivetrain model in a general view on Simulink

Two new subsystems are observed, each of which forms the basic equation of the drivetrain model. It is important to mention that the second output in *Figure 4.8* will be an important value for calculating mechanical power later. The subsystem on the left that receives T_t and T_m as inputs and has ω_t (which would refer to the internal of the mechanical system and is what is fed back into the aerodynamic system) as output, is equation *(14)*:

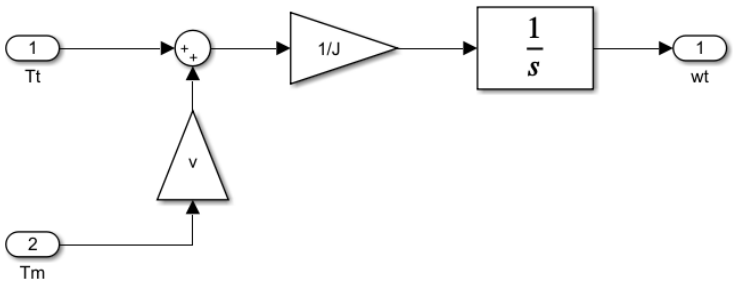

Figure 4.9: Equation (14) modelled on Simulink

In *Figure 4.9*, an integrator is present. This must have an initial condition so that the system does not collapse when simulated, and it is here where the nominal value of the rotor angular velocity is defined as the initial condition. The value of T_m that this subsystem receives as input, is the one calculated as reference mechanical torque, which by equation *(18)* are stipulated as equal values.

Returning to *Figure 4.8*, we will analyze the subsystem on the right, which is the definition of equation *(16)*:

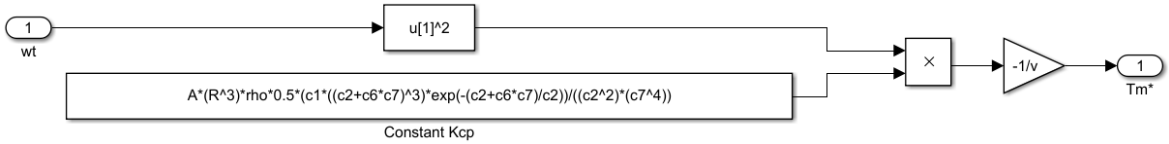

Figure 4.10: Equation (16) modelled on Simulink

Equation *(17)*, which is the constant K_{Cp}, is also represented here. In this way it is possible to obtain the reference value of the mechanical torque that will be used later to calculate the mechanical power obtained by the turbine.

4.3. Simulink model for a two-mass drivetrain

The two-mass drivetrain is more complex than the one-mass DT. We must return to the matrix system *(12)* which represents the four equations needed to model this two-mass design. To implement it in *MATLAB* we are going to use the state space method. The state space representation is the way to describe a physical system through the variables, their inputs and outputs which are related by differential equations, in this case FODEs (of first order). *Figure 4.11* shows the general form of a state-space:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

Figure 4.11: State-space general form

The parameter *x* represents the states, *u* represents the inputs and *y* represents the outputs of the system. As already discussed in the one-mass model, we want as inputs the two torques, both the generator and the turbine, and as outputs the omega of the turbine.

A, *B*, *C* and *D* are all matrices and of dimensions *nxn*, *nxm*, *pxn* and *pxm* respectively:

$$\overset{\dot{x}}{\underset{=}{\begin{bmatrix}\dot{\omega}_m\\ \dot{\omega}_t\\ \omega_m\\ \omega_t\end{bmatrix}}} = \overset{A}{\underset{=}{\begin{bmatrix}\frac{-c}{v^2 J_m} & \frac{c}{v\,J_m} & \frac{-k}{v^2 J_m} & \frac{k}{v\,J_m}\\ \frac{c}{v\,J_t} & -\frac{c}{J_t} & \frac{k}{v\,J_t} & -\frac{k}{J_t}\\ 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix}}} * \overset{x}{\underset{=}{\begin{bmatrix}\omega_m\\ \omega_t\\ \theta_m\\ \theta_t\end{bmatrix}}} + \overset{B}{\underset{=}{\begin{bmatrix}\frac{1}{J_m} & 0\\ 0 & \frac{1}{J_t}\\ 0 & 0\\ 0 & 0\end{bmatrix}}} * \overset{u}{\underset{=}{\begin{bmatrix}\Gamma_m\\ \Gamma_t\end{bmatrix}}}$$

$$\overset{y}{\underset{=}{[\omega_t]}} = \overset{C}{\underset{=}{[0 \quad 1 \quad 0 \quad 0]}} * \overset{x}{\underset{=}{\begin{bmatrix}\omega_m\\ \omega_t\\ \theta_m\\ \theta_t\end{bmatrix}}} + \overset{D}{\underset{=}{[0 \quad 0]}} * \overset{u}{\underset{=}{\begin{bmatrix}\Gamma_m\\ \Gamma_t\end{bmatrix}}}$$

Figure 4.12: State-space matrices of the two-mass model drivetrain

It is observed that matrix *C* only contains a 1 in the second column to obtain the turbine omega as the output, and matrix *D* is null since we do not need any of the torques as output. If we wanted some other input, we would put a 1 in the column that corresponds

to the state needed in *x* (column number will be the row number in *x*), but we must add it in a new row in matrix *C* and add another row of zeros in matrix *D*. In this way we would obtain a matrix *y* of dimensions *2x1* (which would mean two outputs).

To know if the system will be stable, we can use the eigenvalues method on the *A* matrix (calculated with *MATLAB Figure 4.13*). Since two eigenvalues have negative real part and the other two are 0, matrix *A* is stable and hence the system will be stable.

```
>> eig(A_2mass)

ans =

  -1.3890 +12.8353i
  -1.3890 -12.8353i
   0.0000 + 0.0000i
   0.0000 - 0.0000i
```

Figure 4.13: Eigenvalues of matrix A

To pass all this information to *Simulink*, there is a block called *State-Space (with initial conditions)*. To observe the change in the model design, *Figure 4.14* shows the block that changes from the one-mass (*Figure 4.9*) to the two-mass DT:

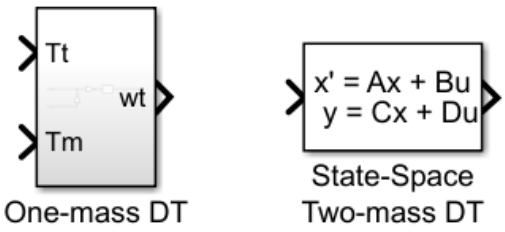

Figure 4.14: Block changes from one-mass to a two-mass model DT

In the one-mass model, an initial condition was defined in the integrator so that the system would not collapse when simulated. It is also necessary to give initial conditions for the two-mass model so that the program has a starting point, however, here initial conditions must be introduced for both the outputs (ω_t) and the inputs (T_m and T_t). It is possible to define only the initial condition of the output, but to improve the response to different wind inputs, we will help the system by introducing initial conditions to the inputs as well. All ω_t, T_m and T_t will have as initial condition their nominal value (*Figure 4.15*):

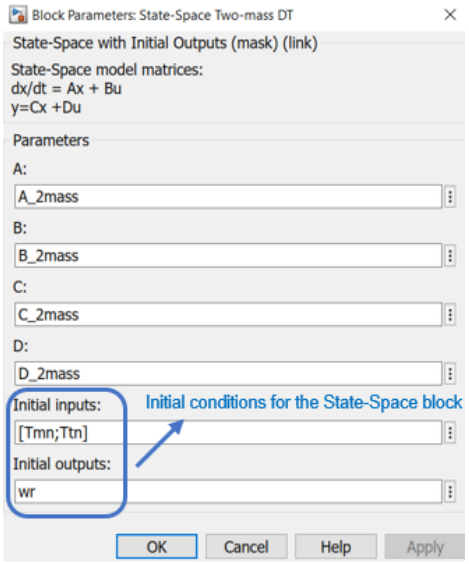

Figure 4.15: Block parameters for the State-Space block

In *Figure 4.14* it is visible that the State-Space block in *Simulink* only supports one input and one output visually, therefore, to provide both inputs it is necessary to place a *MUX* with two inputs, where the first one corresponds to T_m and the second one to T_t. In this case, since we only have one output, the one offered by the *Simulink* block is sufficient; however, if we had more, the solution would be to place a *DEMUX* with *n* outputs to deliver the *n* responses of our State-Space. *Figure 4.16* shows the complete model of the two-mass DT, in which T_t enters and has as outputs the values of T_m, ω_g and ω_t.

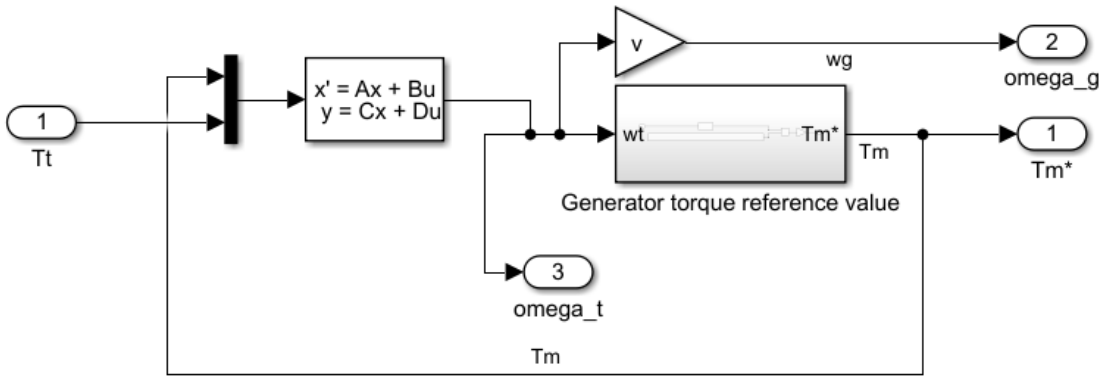

Figure 4.16: Complete Simulink structure of the two-mass DT model

4.4. Proof of the drivetrain

During the analysis of the drivetrain model 150 s will be set as the time domain, wind speed will be 9 m/s and the pitch angle will stay as 0 degrees. First the omega of the generator is required to be *v* times that of the rotor. This is because *v* is the gearbox transformation ratio, so it is necessary that $\omega_g = v * \omega_t$. *Figure 4.17* shows that this condition is met in the stationary zone:

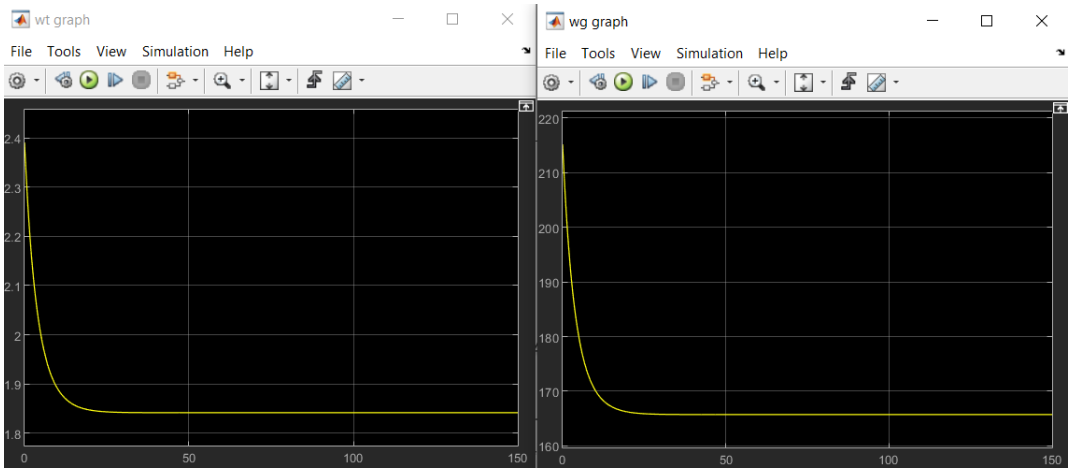

Figure 4.17: ω_t and ω_g comparison for the one-mass drivetrain model

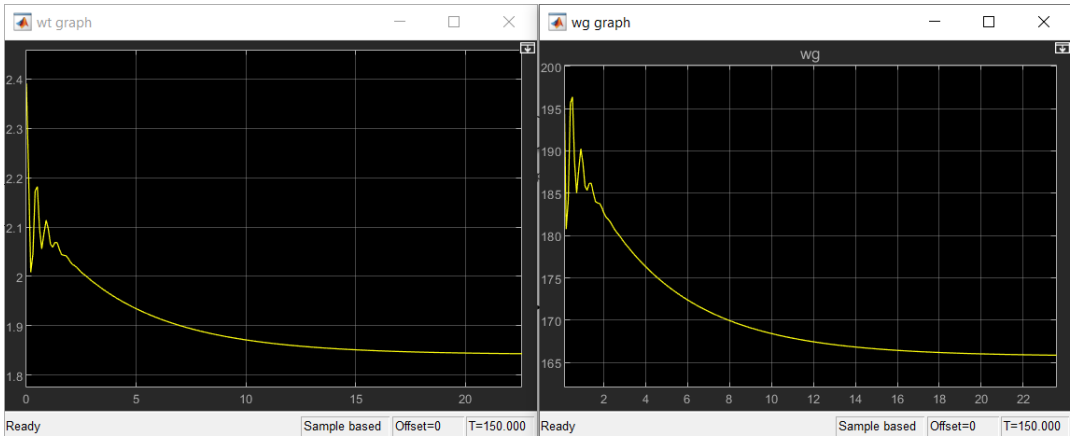

Figure 4.18: ω_t and ω_g comparison for the two-mass drivetrain model

Between *Figures 4.17* and *4.18*, we observe that in stationary state, both models reach the same values, however, due to the complexity added by the two-mass model, it is observed that at the beginning of the simulation the values oscillate (this is due to the poles introduced by the Laplace transformation of the State-Space).

The values of T_t and T_m should be of the same value but of opposite sign if they are in the same mechanical reference. *Figure 4.19* shows the fulfillment of this fact where the absolute value of both is the same and of magnitude 320 kNm:

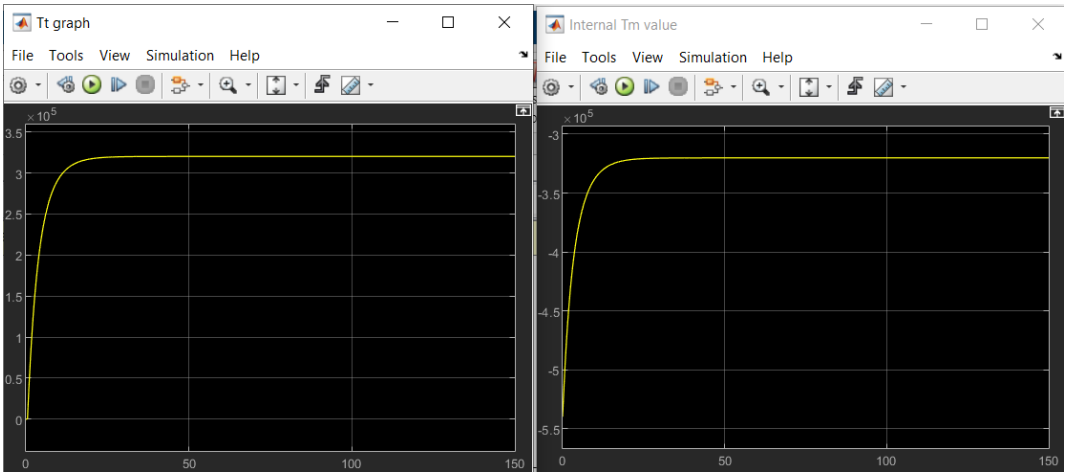

Figure 4.19: T_t and T_m comparison for the one-mass drivetrain model

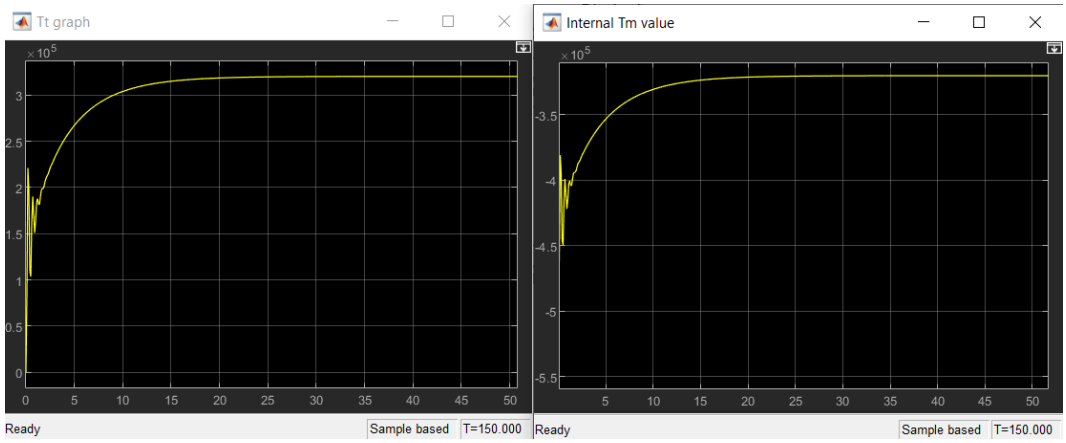

Figure 4.20: T_t and T_m comparison for the two-mass drivetrain model

Again, if we compare both models, the difference is only noticeable at the beginning of the simulation. The steady state value is equivalent in both models, and the oscillations at the beginning are again due to the poles of the system.

Finally, to corroborate that the system is correct, it is necessary to verify that the values of P_t and P_m are identical. For this, it is necessary to calculate P_m through the following formula:

$$P_m = \Gamma_m * \omega_m \quad (19)$$

This formula will be then transferred to the Simulink model as shown in *Figure 4.21*:

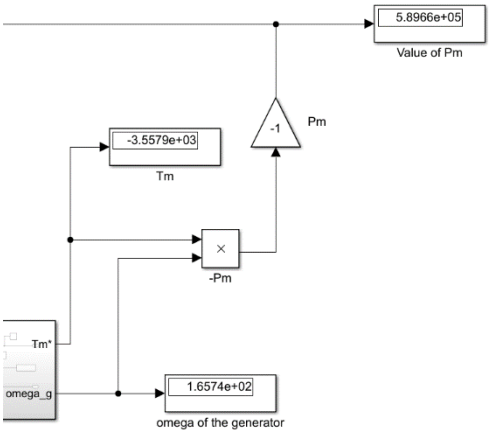

Figure 4.21: Equation (19) implemented on Simulink

Proceeding to analyze both P_t and P_m verifying their equality in stationary state:

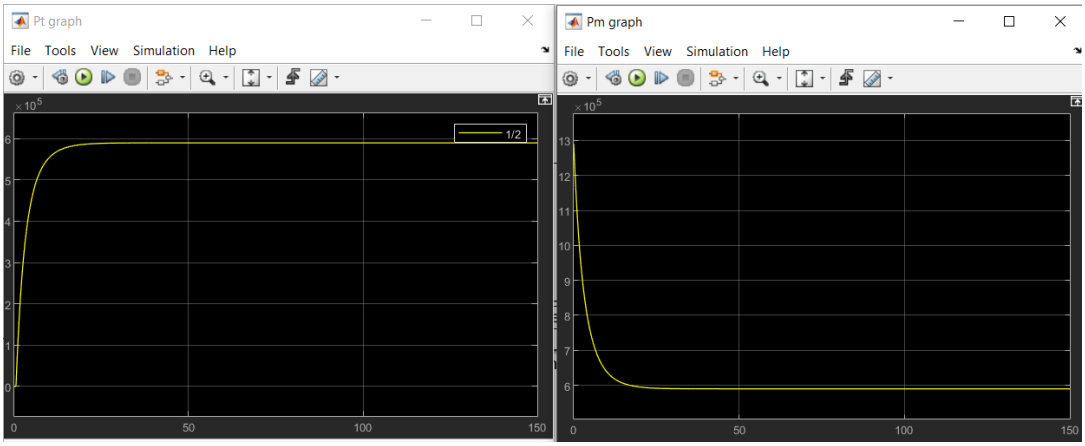

Figure 4.22: P_t and P_m comparison the one-mass drivetrain model

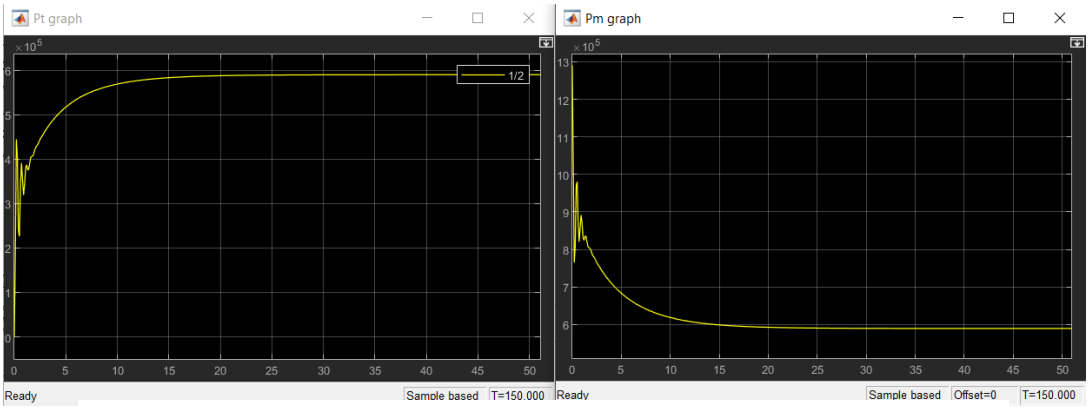

Figure 4.23: P_t and P_m comparison for the two-mass drivetrain model

The same argument explored for the torques applies; the values at the stationary state are the same for both models, and there's oscillation at the start for the two-mass DT.

5. Blade pitch controller

In this section we will design the model of a blade pitch controller for the WT. So far, we have managed to create a model in which, given a wind speed and a pitch angle, we obtain the value of the power extracted by the generator. In a real turbine, clearly the wind value is variable because it depends on the weather conditions and the site where the turbine is located. However, the pitch angle of the blades is automatically regulated to keep the nominal power value at its maximum. Therefore, the wind should be the only parameter to be provided in our model, and the system will be the one that autonomously regulates itself in the case the power exceeds its nominal value.

5.1. Fundamentals

Turbines are not always in operation. There are still many wind farms that do not have places to store the energy obtained by the turbines. This means that the energy extracted must be equal to the energy demanded at that precise moment. To give a simple example, if a house demands 1kW of electrical power at a given instant, it is unnecessary for the turbine to collect 5kW since those 4kW would be lost and therefore it would be completely inefficient. That turbine should be adapted to only extract the kW required by the house and no more. To do this, there are controllers for the angular velocity of the turbine, which automatically change it to obtain a specific power. On the other hand, if the meteorological conditions are making the WT to generate more power than the nominal, then the blade pitch controller is the one that comes into play and adjusts to only extract the rated power of the turbine.

The pitch controller receives as input the power being extracted by the system (P_m), and based on that value and the reference power (P_{ref}), the blades change their angle to match the P_m to the P_{ref}. Therefore, the output is the pitch angle which will be fed back to the aerodynamic model already created.

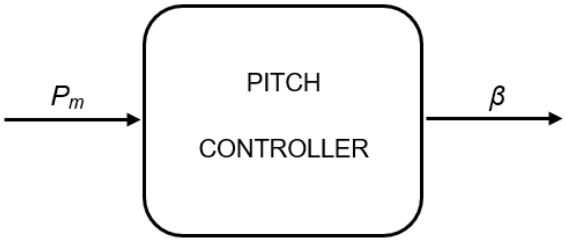

Figure 5.1: Block diagram for the Pitch Controller

Now that the utility of the controller has been explained, we will try to briefly explain the fundamentals that govern it and that must be considered when designing it.

Aerodynamic Force

If we focus on the fundamentals of aerodynamics, it is known that during the incision of fluid on a surface, the nearest particles to the surface accelerate which creates a pressure variation in their surroundings. Low pressure is produced in the particle acceleration zones, while high pressure is produced in the deceleration zones. In *Figure 5.2* [17] there is an example of pressures around the blades of a turbine:

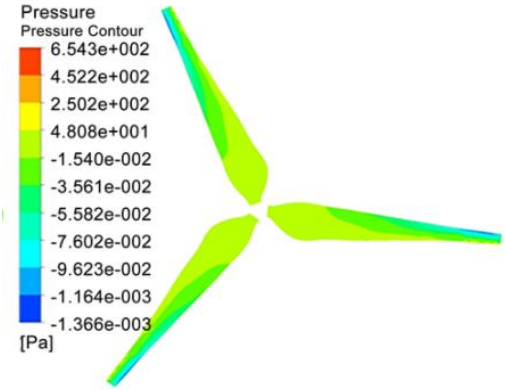

Figure 5.2: Example of pressure map on turbine blades

These pressure differences that are created around the surface of the blade are responsible for the force that is generated, and which generates the rotation itself. This force is governed by the following equation where A_r is the area of the surface where the fluid impinges:

$$F = \rho * {v_w}^3 * A_r \tag{20}$$

Pitch angle β

First there is need to understand what the angle of incidence is, which is equivalent to the pitch angle but expressed in generic nomenclature. The angle of incidence is defining the position at which the fluid impinges on the surface. Thus, the greater the pitch angle, the less energy the turbine can absorb because the blade surface that would come into contact with the fluid would be smaller. On the other hand, if the blade pitch angle is smaller, then the surface area exposed to the fluid will be larger and so the energy it can absorb will be maximum. This is shown in *Figure 5.3* [18].

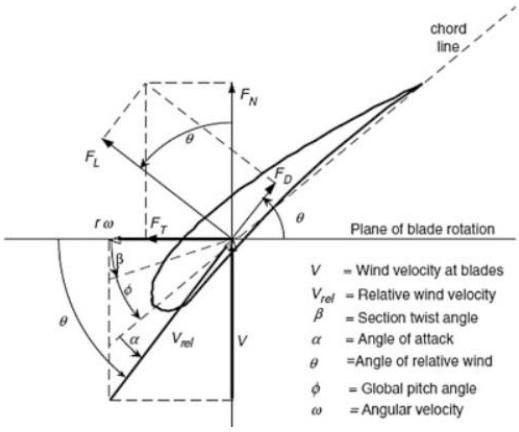

Figure 5.3: Detailed diagram of a turbine blade to understand pitch angle

Depending on the wind speed there is an optimum pitch angle where the power generated by the turbine is at its maximum [18]. Since this work does not have a large scope, we will not calculate or study the optimum pitch angle point for each wind speed value. It should be noted it will be taken as minimum pitch angle 0º and maximum pitch angle 45º. The former refers to the surface of the blades being perpendicular to the wind while the latter one refers to the surface of the blades being diagonally placed with respect to the wind.

5.2. Restrictions

As much as the blade pitch controller allows us to regulate the power to be absorbed by the turbine, there are certain mechanical limits. It is not possible to ask the turbine

to generate ten times more power than the rated power, as this would be impossible and would even damage the components.

The minimum pitch angle, which has already been discussed in the previous section is 0º while the maximum will be 45º due to the system's own mechanical constraints. The lower limit will be providing a maximum turbine torque T_t while the upper limit will be giving a far smaller T_t since the blades will be closer to being in a "flag" position for this last one. In addition, the change of blade pitch angle cannot be instantaneous because it would not be real and hence smooth regulation is required due to limited power issues of the drives, 5º/s will be the maximum angle variation admitted.

For systems, there must be a generation power limit that is defined by the maximum conditions of both the equipment and the power grid itself. In fact, it is not of interest that the turbine works in meteorological conditions where the wind speed is very high because we could for example damage the turbine. This maximum wind speed is called $v_{w,cut}$, and in this case it is 32,5 m/s.

5.3. Simulink model

The overview of the *Simulink* design of the pitch controller preceding the representation in *Figure 5.1* is as follows, where we see that it has the P_m as input and the β as output.

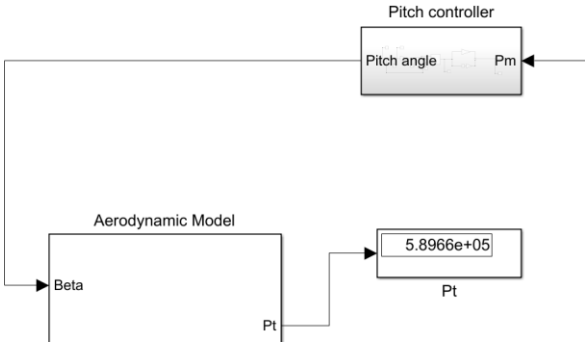

Figure 5.4: Overall view of the pitch controller in Simulink

For the pitch controller to function correctly, it is necessary to create a system that only acts when the extracted power is greater than nominal. A saturator, a PI controller, a

step function, an anti-windup system and generic operators are used. *Figure 5.5* shows the inner system of this controller.

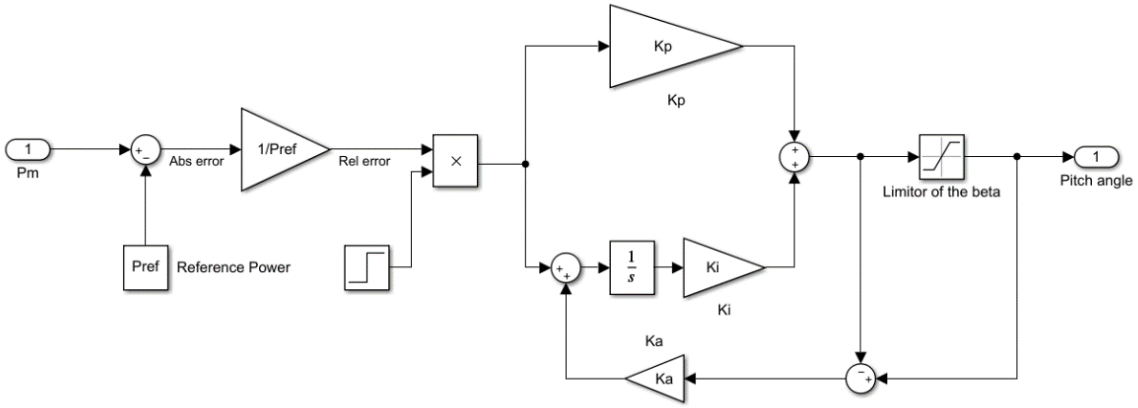

Figure 5.5: Design of the pitch angle controller in Simulink

The controller works based on error signal calculation. At the beginning, the regulator receives the P_m and a reference power P_{ref}, which are subtracted from each other by means of an adder block. The P_{ref} is the rated power of the WT, and by means of the subtraction the absolute error (21) is obtained, which has as unit the *Watt*. To this absolute error the P_{ref} is divided, thus obtaining the relative error (22), which is now non-dimensional. The relative error is the one that will pass through to the PI controller.

$$e_{abs} = P_m - P_{ref} \tag{21}$$

$$e_{rel} = \frac{P_m - P_{ref}}{P_{ref}} \tag{22}$$

It is important to note that the regulation part will not start to operate at the beginning of the simulation (t = 0s). In other words, a *step* function block has been placed with a final value of 1, multiplying the relative error by a multiplier block. Until the step has a value of 1, the relative error will always be multiplied by zero (for a time equal to the step time). In this way, it is possible to leave a time margin before the controller starts to act and thus allow the system to stabilize before altering the power value. This step is necessary because otherwise it is very likely that the system could collapse at the start of the simulation and give invalid results.

To design the blade pitch controller, it will be necessary to use a PI type controller. These controllers contain only one proportional parameter (K_p) and one integral parameter (K_i). These parameters must be adjusted according to the system we have and how we want it to evolve. The equation governing a PI controller is as follows:

$$PI\ controller\ (s) = K_p + K_i * \frac{1}{s} \tag{21}$$

- K_p → It is a parameter which multiplies the error signal given. The higher its value, the faster the response of the system and the lower the error of the system in the stationary state. However, making the K_p too big can make the system to destabilize so it is necessary to balance it enough because a stable system is always a must.
- K_i → It acts as a multiplication too, however it has an integrator as well which, as it names denotes, it integrates de error. The higher the K_i, the lower the error in stationary state. It also increases a little the response velocity, but it makes the system unstable.

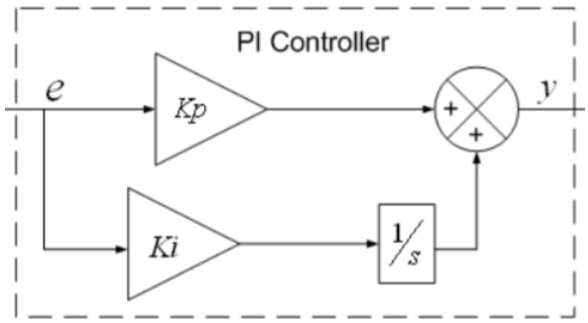

Figure 5.6: PI controller diagram

To determine the values of K_p and K_i there are two ways: the first is to make a calculation in multivariable systems and the other is through trial and error by simulating. In this paper we will use the second method because of the ease of use with the help of *Simulink* compared to the difficulty of the first method. First, both values of K_p and K_i must remain zero. We should start increasing the value of the proportional part until the system starts to oscillate and therefore fails to reach an exact value. We should then leave the value of K_p at the previous iteration from which it started to

oscillate; the proportional part is now adjusted. To adjust the value of K_i we should use the same methodology as with the proportional value and should stop iterating when the dynamism of the system is realistic based on the system constraints. Therefore, there should not be a variation of the angle greater than $\beta = 5º/s$.

$$K_p = 4$$

$$K_i = 1$$

To meet the other constraint discussed in *section 5.2*, the pitch angle should not exceed 45º, and therefore a saturator must be placed at the end of the controller, which is shown in *Figure 5.5*.

Finally, if we look again at *Figure 5.5*, around the saturator and the integral action of the PI controller, there are gains and adders. These are part of a system called *anti-windup*. This system must be taken into consideration in a closed-loop system to prevent the integrator from overloading. That is, when it starts to simulate the model and time goes by, the absolute error would accumulate little by little in the periods that the system was not within the working limits (out of the range of the pitch saturator). This would create problems in the controller and end up giving erroneous results. If the *anti-windup* is not introduced, the moment the system returns to the operating zone, the controller would not be able to act immediately since it would have to eliminate with each iteration the error that has accumulated. This avoids the need to wait for all the accumulated error (overloaded integral action) to be discharged. (The constant K_a used for the *anti-windup* is equal to 1).

5.4. Results of the pitch controller

To analyze if the controller is working properly, we will enter several wind speeds and the P_m and pitch angle graphs will be observed. To begin with, a wind speed of 3 m/s will be set:

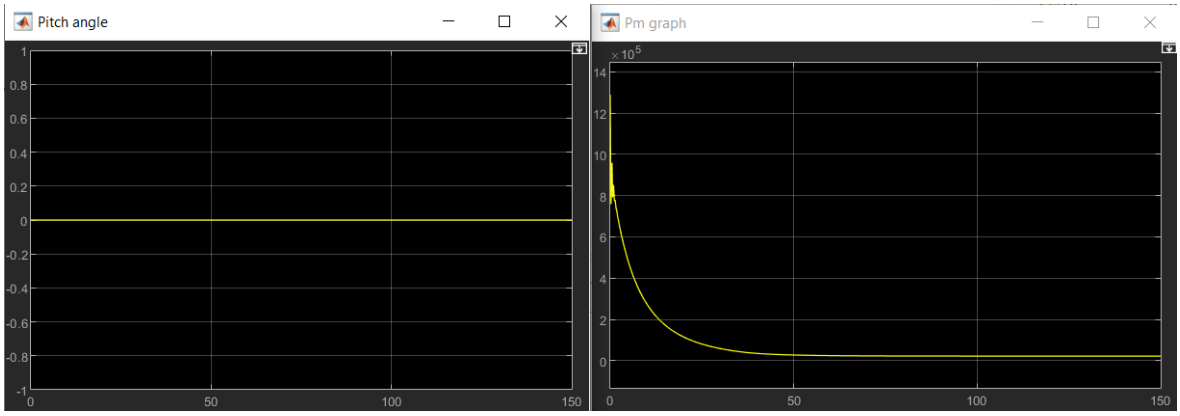

Figure 5.7: Graphs of β and P_m with a wind velocity of 3 m/s

In *Figure 5.7*, as the wind speed is lower than nominal, the pitch controller should not act and therefore its graph should be a horizontal straight line at 0º, and so it is observed. As for the generated power, it should clearly provide a value below the nominal, which is also fulfilled.

We will now test the controller with a wind higher than nominal to see how the system behaves:

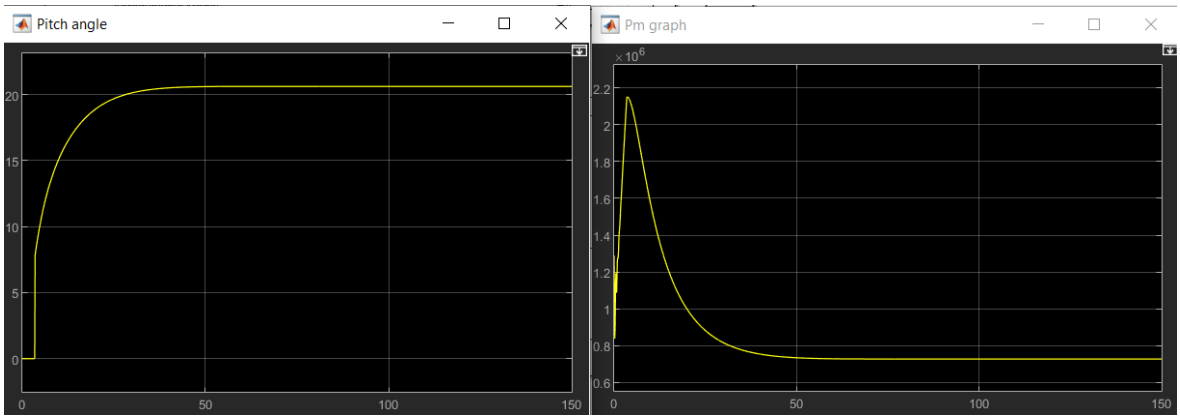

Figure 5.8: Graphs of β and P_m with a wind velocity of 16 m/s

In the case of *Figure 5.8* it is possible to emphasize the fact that the value of P_m starts to lower in the direction of the rated power once the controller can start working (starting when the step function is set to 1). The value of this generated power when the system finally stabilizes is 727,3 kW which is the rated power, then, it is shown that the pitch controller is working correctly since 16 m/s is a wind above the nominal.

Finally, we will check the correct operation by analyzing with an extreme wind and higher than the nominal to see how the turbine would behave. In *Figure 5.9* we can see how the pitch angle value is around 38º and that the maximum of 45º is not exceeded at any time. As for the P_m, we continue to see how the model is adjusted thanks to the controller to extract only the rated power from the WT. Therefore, it is possible to argue that the system is working properly.

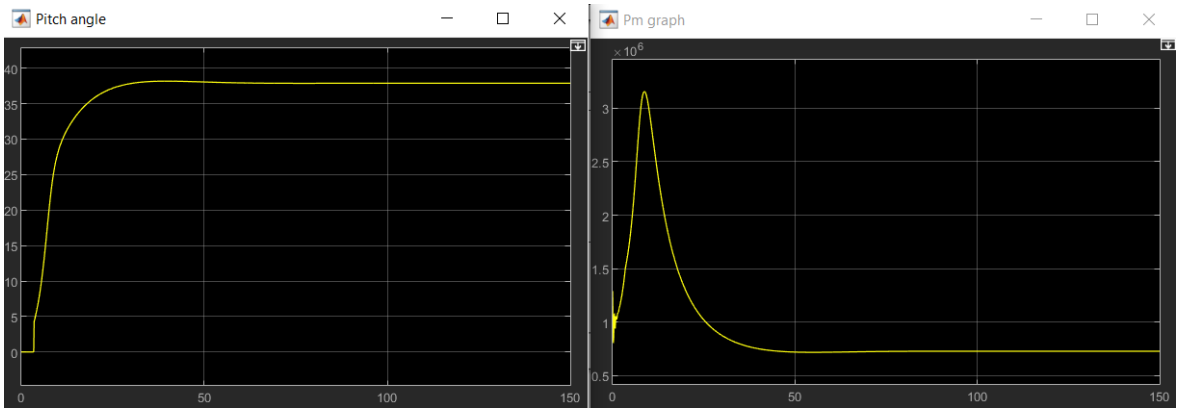

Figure 5.9: Graphs of β and P_m with a wind velocity of 30 m/s

6. Results

In this section the behavior of the model designed in this work will be analyzed. For this purpose, a convenient wind model will be designed to see the evolution of the system at different operating points, as well as at different transitions. In addition, the graphs of C_p against λ for different wind speeds will be studied, comparing them with *Figure 3.11*.

To model the wind and implement it in *Simulink*, the program block called *1-D Lookup Table* will be used (*Figure 6.1*). This block allows to obtain values (also to extrapolate them) from a given database in which only 2 variables exist. In this case, a database has been created where for a range of times, certain wind values are corresponded. The intention is to create a wind model where during a time interval there is a constant wind value, and then make transitions to other wind levels. For a better understanding, see *Figure 6.2*.

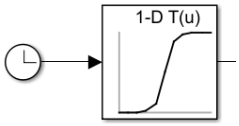

Figure 6.1: 1-D Lookup Table block from Simulink with a Clock as an input

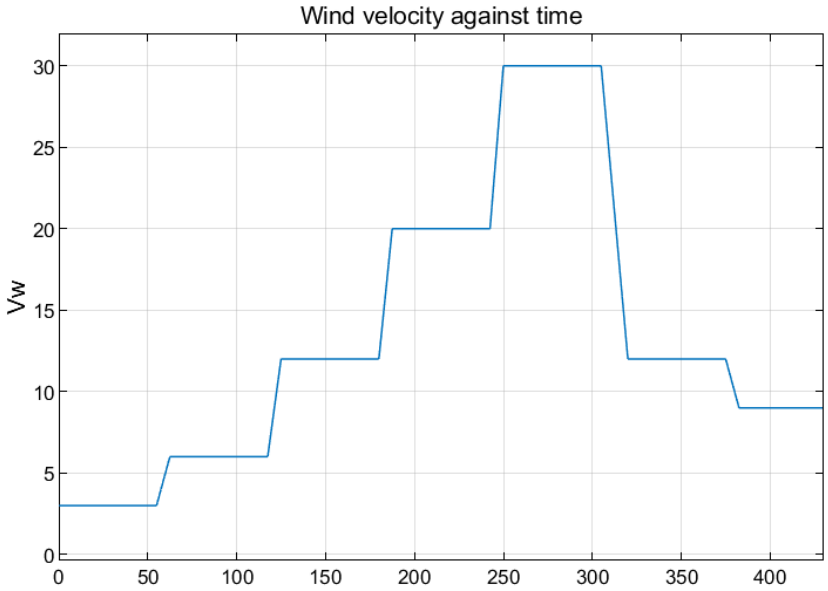

Figure 6.2: Wind model displayed over time

In this designed model there are 7 continuous velocity sections. Their order is as follows: *3*, *6*, *12*, *20*, *30*, *12* and *9 m/s*. It should be noted that the model can be separated into two zones, the first one where the wind increases, and the second one where it decreases. In this way we can see how the pitch controller acts in both situations. The 7 sections have a duration of 55 seconds, and their transitions resemble the behavior of a ramp signal with a duration of 7,5 seconds. In total, if the times are added up, a simulation duration of *430 seconds* is obtained. The database with the times and their respective wind speed has been created in *Excel* and contains 4300 rows of data (one row every 0,1 second). They have been transferred to *MATLAB* in the form of matrices, which are then entered in the *1-D Lookup Table* block.

For the *Lookup Table* to provide the values as the simulation runs, a *Clock* is implemented (on the left of *Figure 6.1*). This clock provides the current value of the simulation time, and then the *Lookup Table* gives as output the wind value corresponding to that time. In this way, the wind curve is generated at the same rate as the turbine model is being simulated.

Simulating the wind turbine model with this velocity profile, we obtain the following plots of pitch and generated power seen in *Figure 6.3* and *Figure 6.4* respectively.

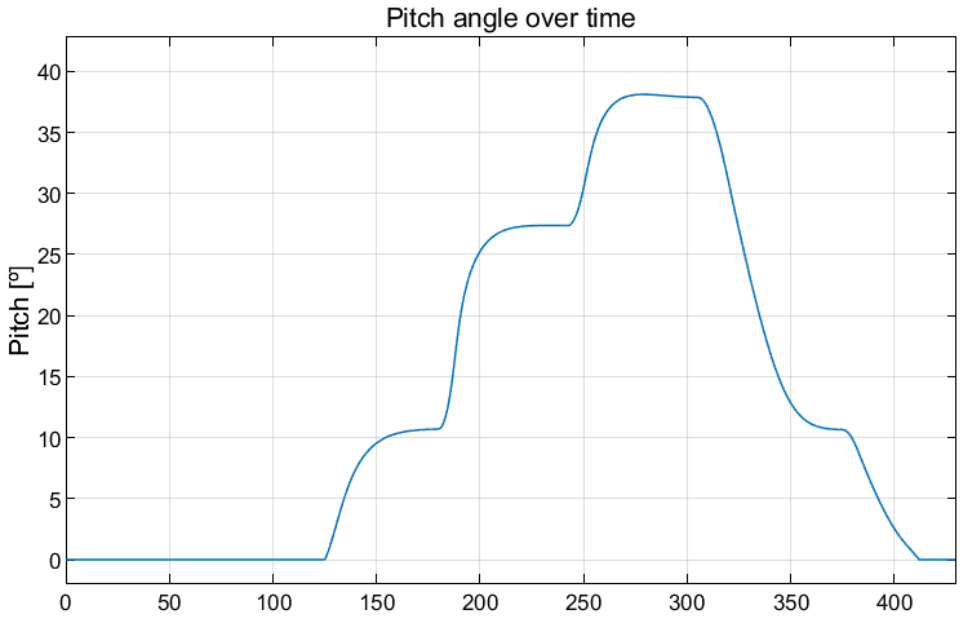

Figure 6.3: Evolution of the pitch angle simulating with the velocity profile

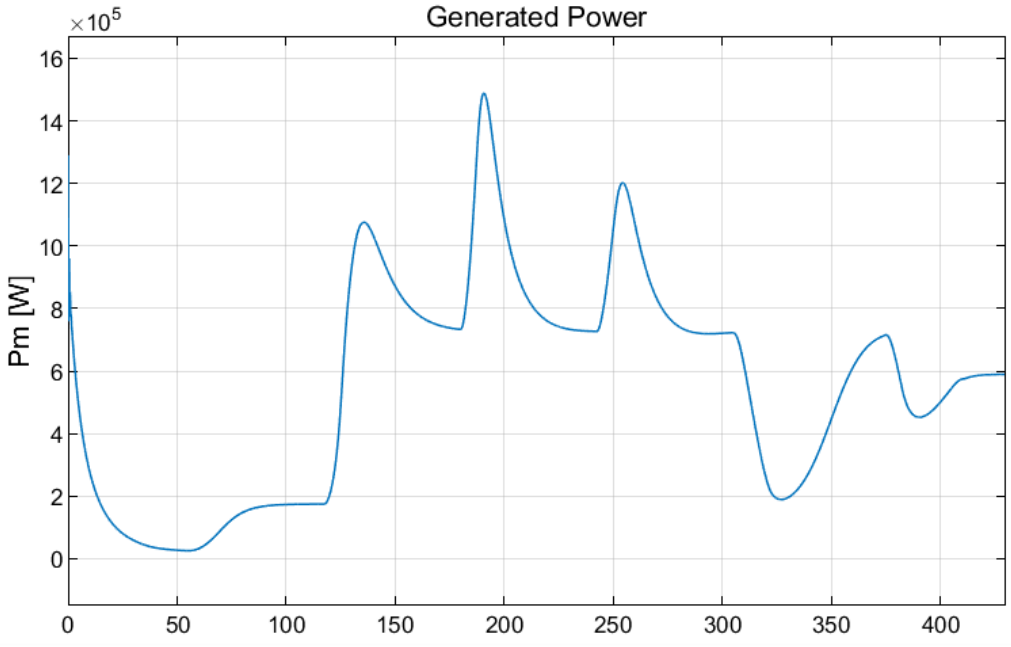

Figure 6.4: Generated power when simulating with the velocity profile designed

Referring to *Figure 6.3*, for speeds below nominal, the pitch angle is *0º*, which is correct because it is not generating power above nominal, and this is corroborated by *Figure 6.4*. Once the velocity profile is above nominal, we observe that the pitch controller comes into operation and therefore starts to change its angle. The peaks seen in *Figure 6.4* are precisely this movement. The system detects that it has started to generate more power than the reference power, and therefore starts to adjust. At the peak, it starts to reverse the power trend until it stabilizes at the nominal power value (*727x10^3 W*). The same happens in the following speed increments.

On the other hand, when the simulation is at *305 seconds*, the velocity decreases from *30 m/s* to *12 m/s* and it no longer forms a maximum but a minimum. This is because the absolute error calculated in the controller is no longer positive; it becomes negative. This fact makes it behave in the same way, but in the opposite direction. However, we see that the final value when stabilized returns to the nominal power and therefore it can be said that it works correctly. Furthermore, knowing that the velocity profile has two periods at *12 m/s* (one when increasing velocities and the other when decreasing velocities) we can see how the value of the angle in *Figure 6.3* is the same in both.

It is also noticeable how the pitch controller returns to outputting a 0^o angle when it reaches the last level of the velocity profile which is under the nominal wind velocity. Hence, it is possible to say that the model of the wind turbine is working correctly.

To deeper analyze, it is possible to obtain the graph with the C_p against λ:

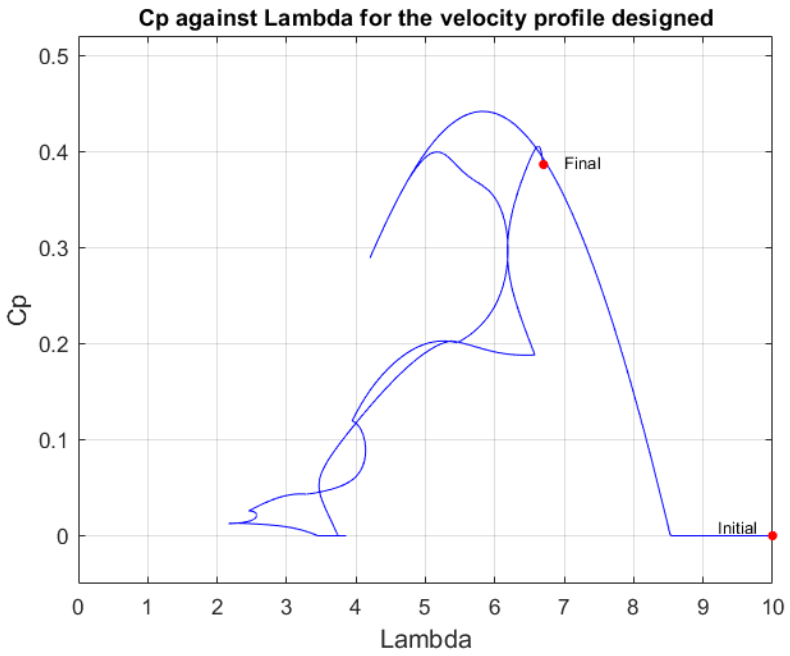

Figure 6.5: Cp against lambda for the velocity profile designed

The initial and final points have been plotted to understand the evolution of this profile. It is observed that for low winds, as it is obvious, the power coefficient is zero, and as the velocity slightly increases it forms a smooth inverted parabola that looks like that portrayed in *Figure 3.7*. As the velocity continues to increment, the shape of the curve turns chaotic. However, it is noticeable that in every change of constant wind, edges appear on the graph and when the model is working on a specific constant wind, its evolution is smooth. This is because the turbine is trying to settle and when the wind changes abruptly it creates the edges seen. Furthermore, there is a tendency to approach a zero C_p as the wind velocity increases, in fact, there is a time where it stays at 0. After that, it returns to increasing its value since the velocity is approaching the nominal value and therefore its optimal working point. *Figure 6.6* shows the value of C_p alone over time.

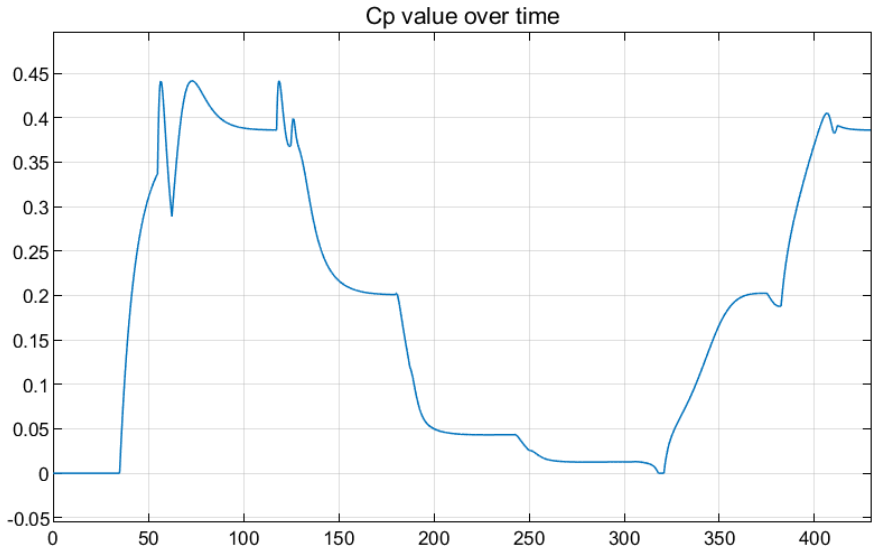

Figure 6.6: Value of Cp over time with the velocity profile designed

Now it is possible to compare *Figures 6.5* and *6.6*. We know the existence of both periods at *12 m/s* and the power coefficient in them is about *0,2*. If we compare it with *Figure 6.5,* we see that the graph intersects with a slope of 0 at that same C_p with a λ of 5,3. We know that all the crossings in the graph are created because the designed wind profile passes through the same points twice; once during the section where it increases velocities and once during the section where it decreases velocities. Therefore, each crossing corresponds to points that have already been passed through before. It can be seen how the final point in *Figure 6.5*, which corresponds to *9 m/s*, is reflected in the last 15 seconds of *Figure 6.6*. It stabilizes at a C_p of 0,386 which is a very marked zone at 110 seconds of simulation. Again, this is because the same working point is passed again. Finally, we can conclude that the optimum power coefficient is met at the nominal wind velocity, and if surpassed, its value continues to decrease until the turbine is saturated. Hence, the importance of placing a specific wind turbine in places where it will work around its optimum point.

To show in more detail the evolution of C_p as a function of the tip speed ratio, two simulations will be performed with two different wind speeds, one being higher than the nominal value and the other being lower. In *Figure 6.7* we can see the results for the former.

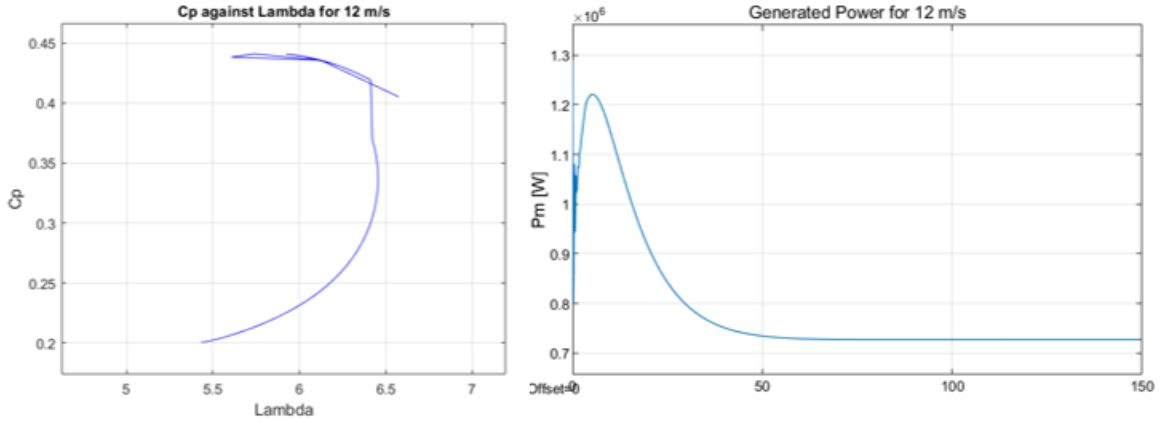

Figure 6.7: Cp against Lambda and generated power for a 12 m/s wind

The way in which the power coefficient evolves is noteworthy. At the beginning it is around values close to the optimum, however, when the pitch controller begins to act, the value of C_p decreases to 0,2, which is the value that was mentioned previously when the model was analyzed with the velocity profile. As for the generated power, it is observed how it stabilizes at the nominal value after some time.

It will now be analyzed in the case of a *9 m/s* wind, a value widely used during this work.

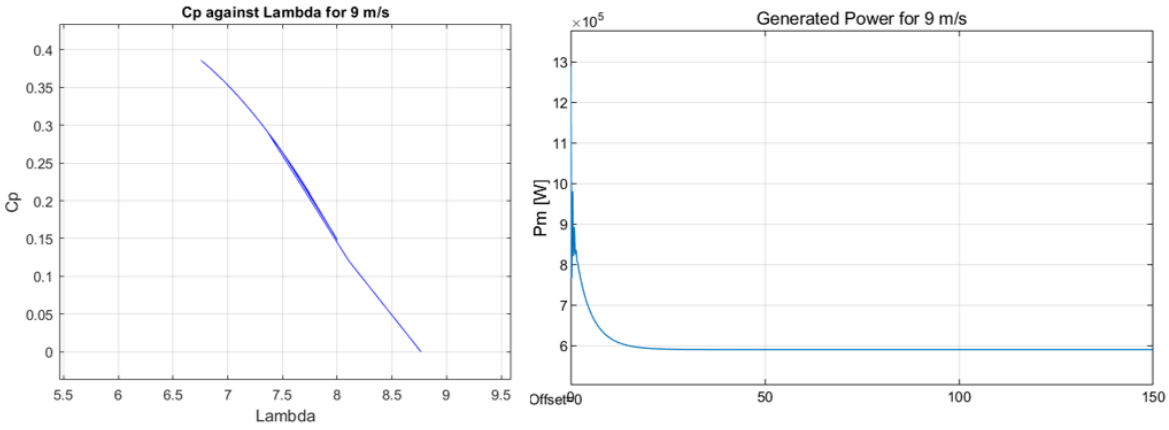

Figure 6.8: Cp against Lambda and generated power for a 9 m/s wind

In this case, exactly the opposite of the previous case happened. Here the value of C_p instead of decreasing, increases until it is close to the optimum (because *9 m/s* is close

to the nominal wind value). We also see that the trajectory is certainly linear, and that is because the model does not work to regulate the generated power and therefore no singular movements are observed. It is further corroborated by the graph of the generated power, which has no peak since, again, the pitch controller does not come into action and therefore the turbine consolidates at $589{,}66 \times 10^3\ W$.

7. Environmental Impact

Renewable energies have a very clear advantage over energies obtained through fossil fuels, which is that they do not contribute to climate change. This has been much talked about in the last decade as it is something that could probably end life on Earth if we do not begin to give it the importance it requires.

Beyond the advantages of renewable energies, in the case of wind farms or isolated turbines, they have certain impacts on the environment that must be considered when installing them. Impacts such as noise pollution, land degradation and direct impact on wildlife are aspects that will be discussed below.

7.1. Noise pollution

Noise pollution is not zero as many believe; noise produced by both the mechanical components of the turbine and noise created due to turbine-related aerodynamics can be considerable in certain cases.

In addition, wind farms are prone to generate more noise because the sound is more noticeable in groups than if it is an isolated turbine. Depending on the area where they are placed, it may also involve more noise. In windy areas, the noise factor due to the aerodynamics of the blades increases exponentially, although these areas are generally less populated and do not affect directly.

7.2. Land occupation and landscape impact

Nowadays turbines have a substantial size, and not only for the size of the blades but also for the thickness of the tower. The great weight and dimension of these turbines requires a good fixation on the ground, which supports both its weight and the momentum that will create the wind impacting on the blades during operation. And not only the location of the wind turbine is considered, but also substations, access roads

and other services for the maintenance and safety of the wind farms must be built in the surrounding areas. All this leads to certain changes in the terrain.

The visual impact is then something that worries many. Turbines are not placed in industrial or neglected areas, in fact, they are placed in rural, green, mountainous areas and it entails a social rejection for those who want to enjoy pure nature.

7.3. Impact on wildlife

According to the report written by Bird Life [19], the fauna most affected by the implementation of wind turbines are birds and bats. They detail that, broadly speaking, what most affects these groups are collisions, disturbance to crossings and habitat destruction.

It happens that the birds are not able to see the blades as they rotate, or even the power lines so they end up colliding. This causes their death or serious injury. Depending on where they are positioned when flying, they can even be absorbed by the pressures that are generated near the blades as they rotate, or if they are placed behind them, there is a release of the wind that generates turbulence making the birds or bats lose balance when flying and they can end up on the ground.

The noise, vibrations and electromagnetic waves generated by wind turbines are also annoying for this group of fauna, which causes them to migrate to other quieter areas so as not to be disturbed by these factors. When changing their habitat, they may encounter problems such as reproductive success or lack of food to keep them alive.

Finally, it is obvious that to build the entire wind farm area, some trees or land that are part of the habitat of many animals will have to be destroyed. They may even be habitual routes for the animals that are now cut off and therefore change their habits which could lead them to a slow death.

8. Economic analysis

To carry out this modeling work there are some associated costs. The costs of the licenses of the programs needed for modeling, the tools to use those programs and of course the cost as an engineer for the design.

- The programs used to make the model of this work have been *MATLAB* and *Simulink*, which is a tool associated with *MATLAB* but has an external cost. The annual cost of these licenses is 1200€ combined.
- To use the programs, it is necessary to use a computer powerful enough to be able to use them without fail. In this case, the price of the computer used which has the essential graphic card to be able to use the programs, has a cost of 1000€.
- Finally, the cost as an engineer to create the model. The hours needed to design the model correctly without errors is about 300 (about 3h per day for 100 days as maximum delivery term). If we take into account a price of 12€ per hour, that equals a total of 3600€ as labor cost.

If we add up all the costs, the total price for the complete design of the model and the written memory with the analyses and explanations of the processes used to design is 5800€.

Conclusion

In this work the main objective has been achieved; the modeling of a wind turbine capable of transforming the wind energy captured by the blades and converting it into power that could be used, for example, to feed an electrical grid. The most important parts of the turbine itself have been designed, such as the aerodynamic modelling and the gearbox. Each of them has been explained in detail through their operation and the equations that govern them. The implementation of a pitch controller has helped to create a complete dynamic system in which just by defining a wind speed, which could be random, the turbine will adapt to provide a certain power; it is self-regulating.

The pitch controller designed comes into operation in case of high and variable winds. Normally, these systems include safety protocols and different efficiencies depending on the wind that is affecting the turbine. This fact is important to consider for the final design of the project to make the right decision for financing or to set the power range to be generated by the turbine.

There has been an analysis of the model through a wind model designed to test the behavior of the turbine under different wind conditions. The behavior of the controller when there were wind increases and when there were sudden decreases has also been observed. Not only that, but it has also been possible to create a smooth model that responds correctly and realistically dynamically speaking, even when there are sudden wind changes. Restrictions found in real turbine systems have been fulfilled and a very common gearbox model has been used (two-mass model), so it is possible to say that a model close to a real case has been designed, which was one of the objectives of this work.

It is remarkable how difficult it is to design this type of models and the work involved in making them correctly. The many existing possibilities of designing wind turbines allow for the obtention of optimal solutions depending on the geographical area in which the turbine is going to work. However, it is this wide range of possibilities that often makes the work of designers difficult, since the dynamic evolution, for example, can vary

greatly depending on the methods used. In addition, the restrictions imposed by the different parts that make up the turbine due to their nominal speeds, require control systems such as the pitch controller designed in this work. All these factors are part of the design process of a model and clearly show its high degree of difficulty on many occasions.

As a point of improvement, in the future it would be interesting to be able to focus more efforts on the electrical part of the turbine, which also plays an indispensable role in its real-life operation.

References

[1] E. Hau, Wind Turbines: Fundamentals, Technologies, Application, Eco- nomics. Springer, 2005. 7, 8

[2] WWEA. (2021, March 24). Worldwide Wind Capacity Reaches 744 Gigawatts. World Wind Energy Association. Retrieved October 12, 2021, from https://wwindea.org/worldwide-wind-capacity-reaches-744-gigawatts/

[3] Red Eléctrica de España. (2020, March 12). Renewable energy already exceeds the installed power capacity of other sources of energy on the Spanish peninsula. Retrieved October 15, 2021, from https://www.ree.es/en/press-office/news/press-release/2020/03/renewable-energy-already-exceeds-installed-power-capacity-of-other-sources-of-energy-on-the-Spanish-peninsula

[4] Quaschning, V. (2005). Understanding Renewable Energy Systems. EarthScan.

[5] Junyent, A., & Gomis, O. (2011). Control of power electronic converters for the operation of wind generation systems under grid disturbances. UPC, 1–19.

[6] González, J. G. (2019, March 5). Representation and estimation of the power coefficient in wind energy conversion systems. SciELO. Retrieved October 2, 2021, from http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0121-11292019000100077#e5

[7] MMPA. (2014). Calculating the Tip Speed Ratio of Your Wind Turbine.

[8] Calo, A., & Burgos, M. (2015). Simulación del Funcionamiento de una Turbina Eólica con Generador de Inducción Doblemente Alimentado. Universidad de Sevilla. Published.

[9] Leba, M., Pop, M., & Ioana, C. (2008). Modeling, simulation and control of wind turbine. ResearchGate. Published.

[10] Zhao, M. (2015, August 15). Nonlinear torsional vibrations of a wind turbine

gearbox. ScienceDirect. Retrieved October 25, 2021, from https://www.sciencedirect.com/science/article/pii/S0307904X15001882

[11] Gonzalez, F., & Regulski, P. (2011). Effect of the shaft stiffness on the inertial response of the fixed speed wind turbines and its contribution to the system inertia. ResearchGate. Published.

[12] F. M. González-Longatt, «Modelo de Sistemas de Turbinas de Viento Parte I: Rotor y Tren Mecánico,» de II Congreso Iberoamericano de Estudiantes de Ingeniería Eléctrica II CIBELEC 2006, Puerto La Cruz, Venezuela, 2006.

[13] D. Goodfellow and G. Smith, “Control strategy for variable speed wind energy recovery,” in Proc. of 8th BWEA Conference, Cambridge, 1986, pp. 219–228. 16, 161

[14] Garcia, J. (n.d.). La modelación, los modelos y su importancia para las ciencias de la educación (página 2) - Monografias.com. Monografias. Retrieved December 28, 2006, from https://www.monografias.com/trabajos36/los-modelos/los-modelos2.shtml

[15] Adán, A. (2016). Modelo de sistema de control de paso de pala aplicado a turbina eólica. UPM. Published.

[16] Grillo, M., Silvestro, F., & Marinelli, M. (2011). Wind Turbines. ResearchGate. Published.

[17] Hasan, M., El-Shahat, A., & Rahman, M. (2017). Performance Investigation of Three Combined Airfoils Bladed Small Scale Horizontal Axis wind Turbine by BEM and CFD Analysis. Journal of Power and Energy Engineering, 05(05), 14–27. https://doi.org/10.4236/jpee.2017.55002

[18] A.R., S., Pandey, M. C., Sunil, N., N.S., S., Mugundhan, V., & Velamati, R. K. (2016). Numerical study of effect of pitch angle on performance characteristics of a HAWT. Engineering Science and Technology, an International Journal, 19(1), 632–641. https://doi.org/10.1016/j.jestch.2015.09.010

[19] Atienza, J. C., Martín, I., Infante, O., Valls, J., & Domínguez, J. (2012, January). Directrices para la evaluación del impacto de los parques eólicos en aves y murciélagos (No. 3). SEO/BirdLife.

[20] Heuveneers, M. (2018). Control design and simulation of an HVDC-connected Offshore Wind Power Plant. UPC. Published.

[21] Tiwari, R., & Ramesh, N. (2017). Comparative Analysis of Pitch Angle Controller Strategies for PMSG Based Wind Energy Conversion System. International Journal of Intelligent Systems and Applications, 9(5), 62–73. https://doi.org/10.5815/ijisa.2017.05.08

[22] Lara, M., Garrido, J., Ruz, M. L., & Vázquez, F. (2021). Adaptive Pitch Controller of a Large-Scale Wind Turbine Using Multi-Objective Optimization. Applied Sciences, 11(6). https://doi.org/10.3390/app11062844

Appendix

Appendix A

```
R = 33; rho = 1.225; v = 90; A = pi*R^2; Beta = 0;

c1 = 1; c2 = 110; c3 = 0.4; c4 = 0.002; c5 = 2.2; c6 = 9.6; c7 = 18.4; c8 = 0.02;
c9 = 0.03;

vw =[11 9 7 5 3];

wr = 22.83*pi/30;

lam = wr*(R) ./ vw;

k1=(lam+c8*Beta).^(-1)-c9/(1+Beta^3);

Cpp=c1*(c2*k1-c3*Beta-c4*Beta^c5-c6).*exp(-c7*k1);

P1=0.5*rho*Cpp*A.*vw.^3;

lam1=0:.1:11;

for j=0:3:21
    k12=(lam1+c8*j).^(-1)-c9/(1+j^3)
    plot(lam1,max(0,c1*(c2*k12-c3*j-c4*j^c5-c6).*exp(-c7*k12)));hold on;grid on;
end
text(3,0.37,'Pitch = 0°','FontSize',10);
xlabel('\lambda','FontSize',12);
ylabel('C_p','FontSize',12);
title('Cp - Lambda for different pitch angles') % title
```

Appendix B

```
R = 33; rho = 1.225; v = 90; A = pi*R^2; Beta = 0;

c1 = 1; c2 = 110; c3 = 0.4; c4 = 0.002; c5 = 2.2; c6 = 9.6; c7 = 18.4; c8 = 0.02;
c9 = 0.03;

w2=0:1:450; % Vector of machine speeds [rad/s]
for ii=1:1:9; % Creation of the 'for' loop. It will execute it 6 times, as
defined (1:1:6)
vw=1+ii*2; % Wind speed (for simplicity, we calculate it based on the iteration
number)
clear tsr; % Clear the tip speed ratio for the calculation of the current
iteration
tsr= w2*R/(v*vw); % Calculation of the TSR for the current wind speeds[vw] and
the linear blade speed [w2*R_turbina/(n_multiplicador)]
k1=(tsr+c8*Beta).^(-1)-c9/(1+Beta^3); % aux variable for the cp calculation
cp=max(0,c1*(c2*k1-c3*Beta-c4*Beta^c5-c6).*exp(-c7*k1)); % Calculation of the Cp
constant for the current iteration
```

```
P_turbine(ii,:) = (1/v)*0.5*rho*A*cp*vw^3.*(1/v).^-1; % Turbine torque (fast
shaft)
txt{ii}=['Wind=' num2str(vw) ' m/s']; % Creation of a 'txt' vector for the legend
end;
%Grafic [Velocitat eix - Parell turbina]
figure(); % New fig
plot(w2*30/pi,P_turbine,'LineWidth',2);grid on; % plots the turbine to the speed
of the machine
xlabel('\omega fast shaft [rpm]','FontSize',14); % x axis label
ylabel('P_ turbine fast shaft [Nm]','FontSize',14); % y axis label
legend(txt); % legend
title('Speed - Turbine power') % title
```

Appendix C

Initial variables are the same that those in Appendix A.

```
vw2=0:.1:15;

lam2= min(20,max(wr*(R) ./ vw2,0));

Cpp2= (cpopt/289) * (289 - (lam2-17).^2);

P2=0.5*rho*Cpp2*A.*vw2.^3;

P2a=min(0.5*rho*Cpp2*A.*vw2.^3,727.3e3);

h=subplot(1,1,1);

plot(vw2,P2a,'k');hold on;grid on;
plot(vw2,P2a,'k');
plot(vw,P1,'k.');
plot(vw2,P2,':k');
for ii=5:1:5
 txt{ii}=['v_w = ' num2str(vw(ii)) ' m/s'];
 text(11.1,800e3,txt{ii},'FontSize',8.2);
end;
text(vw(6),650e3,['v_w = ' num2str(vw(6)) ' m/s'],'FontSize',8.2);
text(vw(4),460e3,txt{4},'FontSize',8.2);
plot(11,727.3e3,'ro','MarkerSize',5);
plot(13,727.3e3,'ro','MarkerSize',5);
xlabel('v_w','FontSize',12);
ylabel('P','FontSize',12);
title('Turbine Power - Velocity of the wind') % title
set(h,'FontSize',12);
```