



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

TRABAJO FINAL DEL

REALIZADO POR

TUTORIZADO POR

CURSO ACADÉMICO: 2020/2021

Abstract

Nowadays, in order to measure the dynamic performance of different propellers, wind tunnels are the common option used by researchers. However, wind tunnels require high initial investments, and their operating costs are also elevated, these conditions make them unavailable for most researchers and engineers. Along this thesis, a more accessible alternative to wind tunnels is presented and implemented in order to model the dynamic performance of different propellers. This alternative consists of a car top testing rig which eliminates the need for a wind tunnel. All the equipment, software, and procedures that have been implemented are carefully explained. Four different propellers have been tested, obtaining their thrust and power coefficients as well as their efficiencies. This acquired data will be used in the future for calculating the drag of a prototype UAV, also it will contribute to the election of the most suitable propeller for the UAV mission.

Acknowledgements

Throughout the writing of this final Bachelor thesis and during the four years of my bachelor I have received a great deal of support and assistance.

I would first like to thank my supervisor, Professor David Lundström, whose help and guidance throughout the whole project have been key to the success of this thesis. Also, I would like to thank LIU for giving me the opportunity and resources to develop my thesis in such an interesting and challenging thesis. Then I would like to thank my colleagues and friends Ricardo Garcia, Francisco Angel Jaime, and Ferran de Andres for all the hours and late nights that we have spent together working hard in order to finish the bachelor, without them I would not be where I am now.

I would also like to thank my parents for all the effort that they have put on me and all the opportunities that they have given me in life so I could develop as a person and as an engineer. Especially I would like to thank my dad Jose Miguel Bleda, without his support and encouragement I'm sure that I would have not achieved have of my goals in life, including this bachelor's degree, as he taught me the value of effort.

Nomenclature

Abbreviations and Acronyms

Abbreviation	Meaning
LiU	Linköping University
RPM	Revolutions per minute
NI	National Instruments
GPS	Global positioning System
ESC	Electronic Speed Control
APC	Advanced Precision Composites
LiPO	Lithium Polymer
UAV	Unmanned Aerial Vehicle

Latin Symbols

Symbol	Description	Units
A	reference area	$[m^2]$
C_D	Drag coefficient	$[-]$
C_P	Propeller power coefficient	$[-]$
C_T	Propeller thrust coefficient	$[-]$
D	Drag	$[N]$
d	Propeller diameter	$[m]$
n	Propeller revolutions per second	$[rev/s]$
P	Power	$[W]$
T	Thrust	$[N]$
v	Free stream velocity	$[m/s]$
ρ	Air density	$[Kg/m^3]$
η	Efficiency	$[-]$
Q	Torque	$[Nm]$

Contents

1 Introduction	1
2 Theory	3
2.1 Propulsion System Characterization	3
2.2 UAV drag calculation	3
2.3 Propeller Efficiency	4
3 Method	5
3.1 Required equipment	5
3.1.1 Electronic Equipment	5
3.1.2 National Instruments	6
3.1.3 Propellers	6
3.1.4 Electric motor and batteries	6
3.2 Data Logging Software (LabView)	7
3.2.1 Analog signal reading and processing (NI 9205)	8
3.3 Thrust and torque reading and processing (NI 9237)	10
3.3.1 RPM reading and processing (NI 9401)	11
3.4 Sensor Calibration	12
3.4.1 Static Pressure Calibration	12
3.4.2 Dynamic Pressure Calibration	12
3.4.3 Alpha and Beta Calibration	13
3.4.4 Current Calibration	13
3.4.5 Thrust and Torque Calibration	14
3.5 Electric Motor Control	14
3.5.1 VESC	14
3.5.2 Polulu Micro Maestro	14
3.6 Testing Rig	15
3.7 Test procedure	18
3.8 Troubleshooting	19
3.8.1 No RPM reading making Labview crash	19
3.8.2 Buffering issue causing breaks on the data login	20
3.8.3 Thrust and Torque measurement error	21
3.8.4 Electric Motor and ESC Failure	23
4 Results	25
4.1 Propeller APC 18x10	25
4.2 Propeller APC 18x12	27
4.3 Propeller APC 20x10 Pusher	29
4.4 Propeller APC 20x10 PN	31
5 Conclusions	32
6 Perspectives	34
6.1 GPS implementation	34
6.2 Possible solution for Buffering	35

Appendices	39
A First appendix	39
A.1 Electric Motor Specifications	39

1 Introduction

The purpose of a propeller is to convert engine power (in this case from an electric motor), delivered to the propeller by a rotating shaft, into thrust force, and to do so as efficiently as possible throughout a suitable range of vehicle velocities. Propellers generate thrust by accelerating large masses of air, which is possible thanks to the propeller blades that are shaped like airfoils.

The performance of every propeller can be measured in terms of the thrust and power that it generates, as well as its efficiency. However, in order to calculate these parameters, the propellers need to be tested against the entire velocity and RPM range over which is intended to operate. Until now, wind tunnels have been used for this purpose, although they present a significant constraint, the elevated investment and operating costs, which makes them inaccessible to most researchers and engineers.

In this thesis, an alternative to wind tunnels consisting of a car top propeller testing rig has been implemented. This equipment allows simulating the incoming airflow that the propellers would face when flying mounted on an aircraft, in this way the obtained results for thrust and power generated to match the ones that would be generated in flight.

The objective of this project is to test four different propellers to model their dynamic performance. The computation of these parameters has two main goals. The first goal is to select the most suitable propeller for the different missions that will be carried out by an experimental UAV developed by Linköping University. The second objective is to use the thrust data generated by one propeller to calculate the drag force that is generated by the UAV itself.

Through this thesis, a detailed explanation of all the equipment and software used for the operation of the testing rig has been presented. Moreover, a step-by-step guide of the testing procedures is detailed, so anyone who would want to repeat the experiment with different propellers can follow this report successfully. In order to save time and effort for future engineers carrying out the test, a specific section with all the errors and troubles that have been faced along the project with their solutions has been included, as well as a final section with suggestions for improving the experiment in the future.

2 Theory

2.1 Propulsion System Characterization

Propeller performance can be characterized by means of the thrust and power coefficients, C_T and C_p , as a function of advance ratio, J .

$$C_T = \frac{T}{\rho n^2 d^4} \quad (1)$$

$$C_P = \frac{P}{\rho n^3 d^5} \quad (2)$$

$$J = \frac{v}{nd} \quad (3)$$

In order to compute the C_T and C_p coefficients, the propeller's thrust and input power need to be mapped against the entire velocity and RPM range over which is intended to operate [1]. Usually, these data measurements are made inside a wind tunnel. These coefficients are useful to predict the behavior of propellers with the same shape but different sizes.

2.2 UAV drag calculation

The main objective of this experiment is to measure the dynamic performance of different propellers so the drag of the UAV can be calculated.

When an aircraft is on a steady flight, flying at a fixed velocity and there is no acceleration nor altitude changes, it means that the forces acting on the aircraft are in equilibrium.

$$Lift = Weight \quad (4)$$

$$Thrust = Drag \quad (5)$$

According to the equation [5] on a steady flight at a fixed velocity, the thrust generated by the motor and propeller is equal to the drag of the UAV. So if the thrust that is generated by a propeller at a given RPM and an incoming airflow velocity is known, the drag is also known.

In order to calculate the drag coefficient (C_D) of the UAV, the drag equation [6] has to be used.

$$D = \frac{1}{2} \rho v^2 C_D A \quad (6)$$

Knowing the density of the air, the flight velocity, the reference area of the aircraft, and the thrust generated by its propeller at a given moment which is equal to its drag, the drag coefficient for the aircraft can be solved.

$$C_D = \frac{2D}{\rho v^2 A} \quad (7)$$

To sum up, the main purpose of obtaining the dynamic performance of the propellers is to be able to compute the drag coefficient of the UAV.

2.3 Propeller Efficiency

Propeller efficiency is defined as the power produced by the propeller divided by the power applied by the electric motor [2].

The power produced by the propeller is determined by the thrust it generates at a certain velocity:

$$P_{\text{out}} = Tu_0 \quad (8)$$

And the power generated by the electric motor can be calculated by multiplying the torque generated with the angular velocity of the propeller:

$$P_{\text{in}} = 2\pi nQ \quad (9)$$

Finally, with the values of the input and output power the efficiency of the propeller can be calculated as a function of advance ratio (J):

$$\eta_{\text{prop}} = \frac{P_{\text{out}}}{P_{\text{in}}} = \frac{Tu_0}{2\pi nQ} \quad (10)$$

3 Method

In this section all the equipment and software required to conduct the experiment will be thoroughly explained, as well as the methodology followed for the test itself and the data analysis.

3.1 Required equipment

In order to conduct the experiment to measure the dynamic performance of a propeller, the following equipment has been used:

3.1.1 Electronic Equipment

The sensor used for the measurements are the following:

- **Static pressure sensor:** The static pressure will be measured with the HCA-BARO sensor connected to the static port of the Pitot Tube.
- **Dynamic pressure sensor:** The dynamic pressure will be measured with the HDI-E sensor connected to the total pressure port of the Pitot Tube.
- **Alpha and Beta sensors:** In order to measure wind direction in the vertical and horizontal axis, two wind probes have been installed on the pitot tube, both giving an angle as output.
- **Temperature sensor:** To compute air temperature the LM35 sensor will be used
- **Voltage sensor:** To read the voltage a voltage divider will be used with a 1/10 relation.
- **Current sensor:** The current will be measured using the ACS758LCB sensor.
- **RPM sensor:** In order to measure the motor RPM two possibilities have been tested, a magnetic RPM sensor and a brushless RPM sensor, finally the brushless RPM sensor will be used as the magnetic sensor does not make proper contact with the motor giving incorrect readings.
- **Pitot Board:** All the previously mentioned sensors are connected to the pitot board, which collects all the data and sends it to the NI module 9025. The pitot board needs to be powered by an input 5V which is provided by a battery connected to the board.
- **Thrust and Torque sensor:** To measure the thrust and torque produced by the propeller the motor and propeller are mounted on the thrust and torque rig which uses strain gauges for measuring.

3.1.2 National Instruments

In order to read and process all these signals, National Instrument components together with the software LabView has been used. To read the digital signals the module NI-9205 will measure the voltage output generated by the sensors of α , β , static and dynamic pressure, temperature, voltage, and current. The thrust and torque signals are processed by a NI module NI-9237. To measure the RPM the National instruments module NI-9401 has been used to measure digital signals with a counter in the program. These modules have been mounted in a NI cDAQ-9174 chassis so they can be connected via USB to a computer for signal and data processing.

3.1.3 Propellers

The three propellers that are going to be tested are manufactured by the company Advanced Precision Composites (APC):

- **APC 18x10:** with a pitch of 10 inches and a diameter of 18 inches.
- **APC 18x12:** with a pitch of 12 inches and a diameter of 18 inches.
- **APC 20x10P:** with a pitch of 10 inches and a diameter of 20 inches, made for pusher configuration.

3.1.4 Electric motor and batteries

The electric motor used for the test is the Turnigy Aerodrive SK3 - 6364-213KV Brushless Outrunner Motor, with a maximum power output of 2550 [W], the rest of the specifications can be seen in Appendix [A.1](#). The battery used to power the motor is a LiPo battery of 10 cells.

The pitot board is powered by a 2 cell 7.4 [V] LiPo battery and the NI DAQ-9174 chassis is powered by a 4 cell LiPo battery of 14.8 [V] and 4450 [mAh]

3.2 Data Logging Software (LabView)

The software Labview from National Instruments has been used for signal processing and data acquisition tasks. The different NI modules have been configured with the help of the National Express DAQ Assistant. The implemented code can be seen in Figure 1.

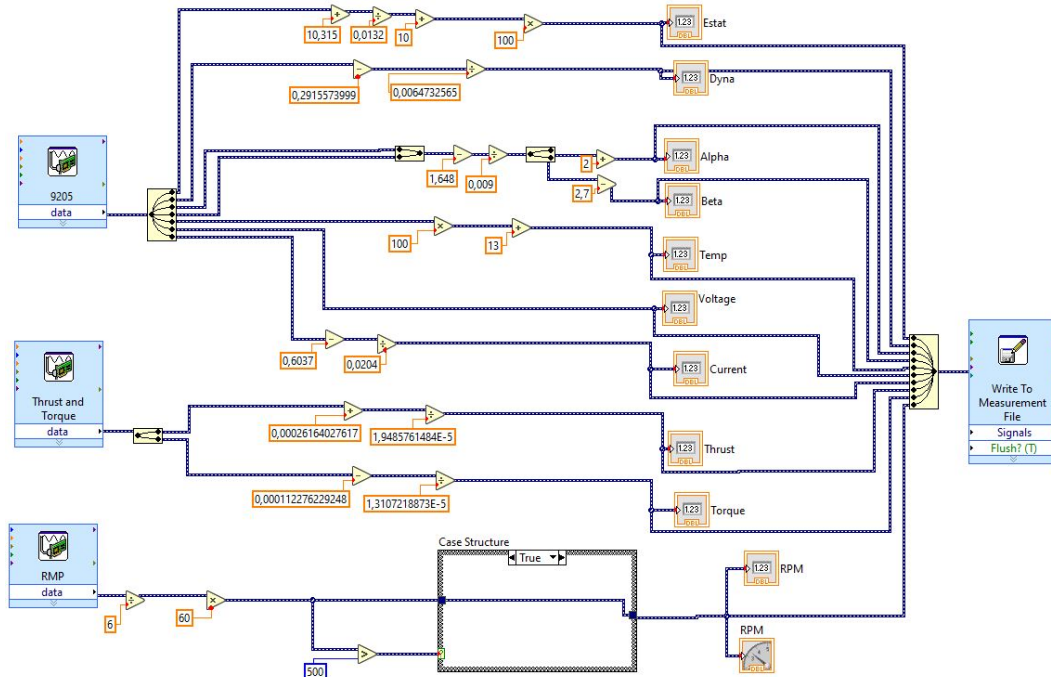


Figure 1: Physical input channel for each sensor

The first three items that can be found on the left are the DAQ assistants for the NI 9205, NI 9237 and NI 9401, whose configuration will be explained on the following sections. Then for the NI 9205 a split signal item has been used in order to have each signal on a single wire so it can be manipulated through numerical tools. Then each signal is connected to a numeric indicator in order to display the value on the front panel for visualization during the test, this can be seen in Figure 2.

Finally, in order to be able to analyze all the data after the test is completed, a write to measurement file module has been implemented, this tool automatically creates an Excel file where each column corresponds to one of the outputs and the exact time where each measurement was made.

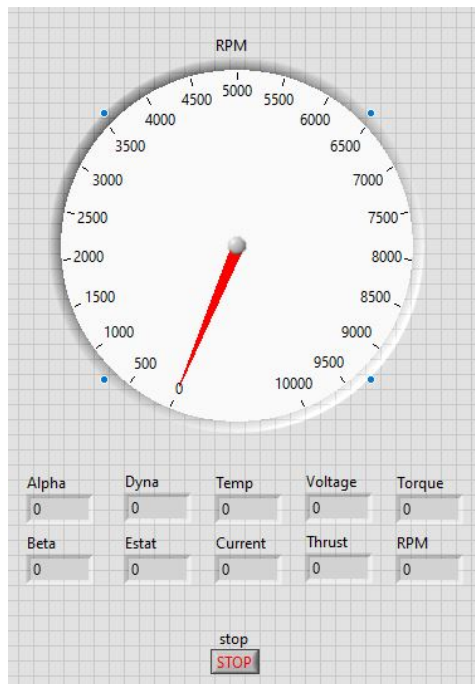


Figure 2: Front Panel

3.2.1 Analog signal reading and processing (NI 9205)

As mentioned before, the analog signal will be processed through the module NI 9205 which reads the voltage output of each sensor. Through the configuration panel of the DAQ assistant, every input signal can be named, and depending on which pin the sensor is connected it is selected on the configuration panel as can be seen in Figure 3.

Channel Settings		
	Order	Physical Channel
Estat	0	cDAQ1Mod1/ai0
Dynamic	1	cDAQ1Mod1/ai1
Alpha	2	cDAQ1Mod1/ai2
Beta	3	cDAQ1Mod1/ai3
Temp	4	cDAQ1Mod1/ai4
Voltage	5	cDAQ1Mod1/ai5
Current	6	cDAQ1Mod1/ai6

Figure 3: Physical input channel for each sensor

For the rest of the configuration, it is really important to choose RSE in the Terminal Configuration. Regarding the timing settings, the acquisition mode N samples have been used with 100 samples to read at a rate of 1k [Hz] as it is shown in Figure 4.

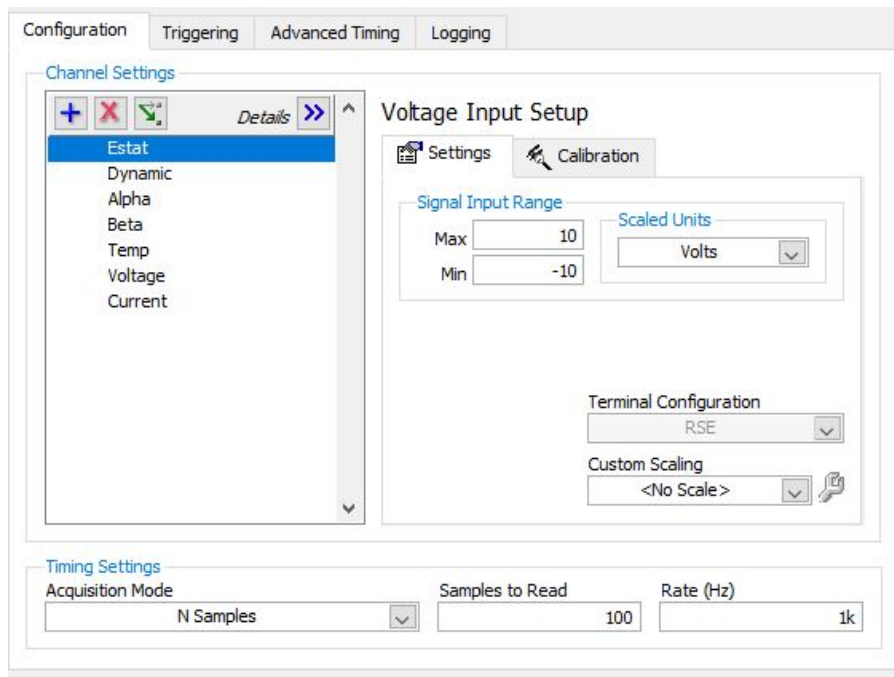


Figure 4: Configuration of NI 9205

Finally, in order to convert the voltage measurements into useful data readings the following Labview code has been implemented:

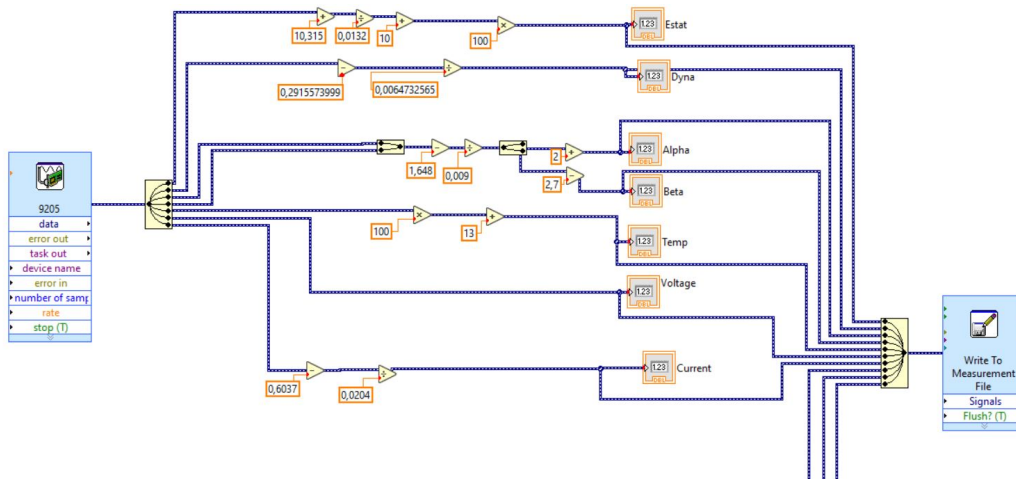


Figure 5: Labview code for voltage signal processing

As it can be seen in Figure 5 all the signals (except the voltage signal) go through a processing stage where through mathematical operators the output voltage is converted to the desired physical magnitude that needs to be measured. The calibration process of every signal is explained in detail in the following section 3.4

3.3 Thrust and torque reading and processing (NI 9237)

In order to read the thrust and torque sensor measurements, the sensor has been connected through an RJ50 cable to its own National Instruments module NI 9237 , both signals use the signal type of Bridge V/V, the timing acquisition settings have to be the same for the three modules. The configuration of the NI 9237 can be seen in Figure 6.

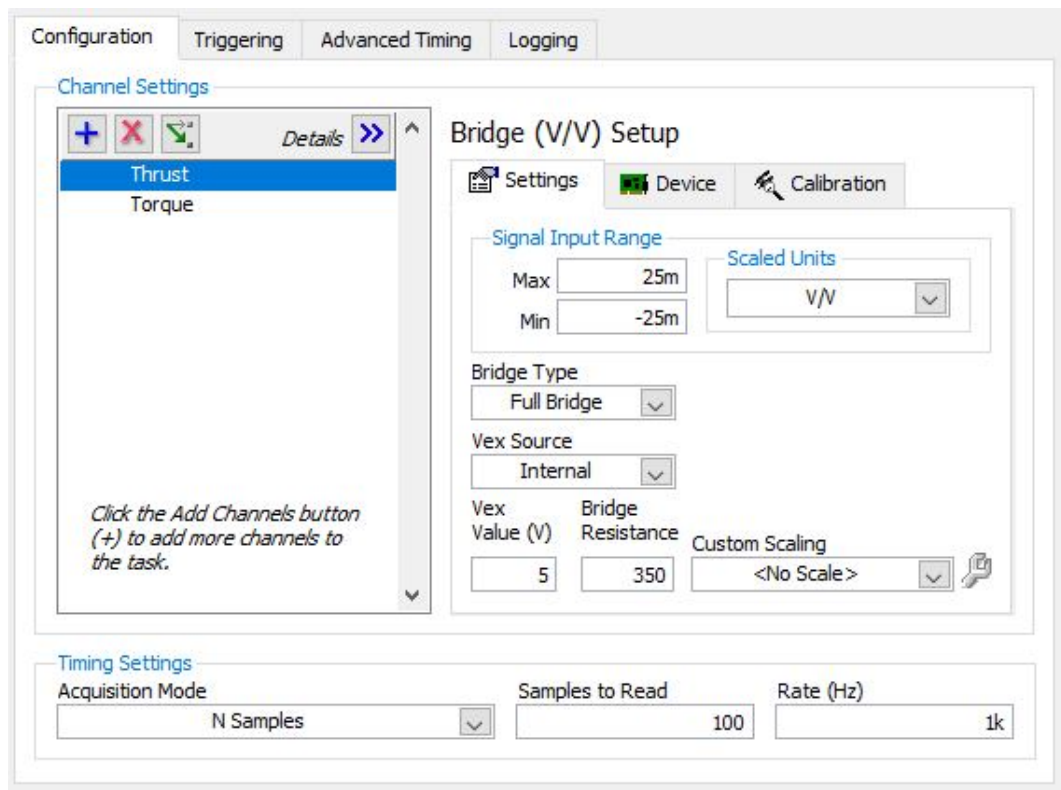


Figure 6: NI 9237 configuration

Then, in order to convert voltage measurements into thrust and torque values after calibration, the code implemented in Figure 7 has been used.

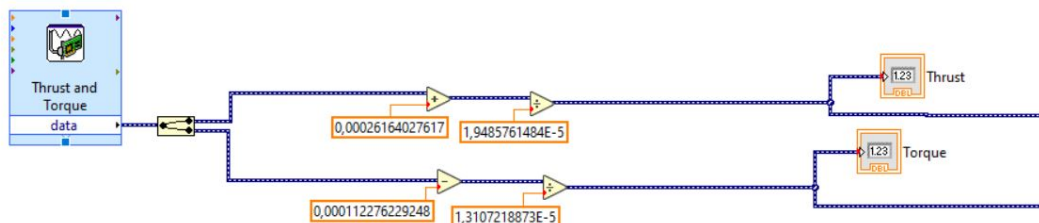


Figure 7: NI 9237 wiring

3.3.1 RPM reading and processing (NI 9401)

For the RPM reading the module, NI 9401 has been implemented and configured as it is shown in Figure 8

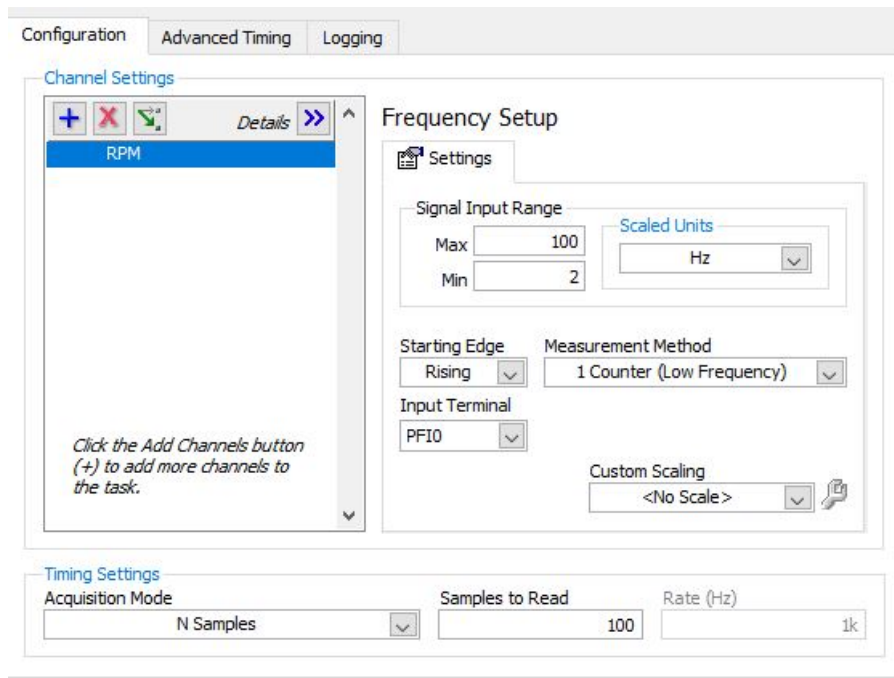


Figure 8: Configuration of NI 9401 for RPM reading

As it can be seen, the counter measurement method has been chosen, then the frequency has been set as low frequency, and the input terminal has been defined. Regarding the timing settings, the acquisition mode N samples have been used with 100 samples to read at a rate of 1k [Hz], as all the NI modules need to have the same timing settings.

Then, for processing the signal and converting the ERPM into real RPM, the measured signal has been divided by 6 as this is the number of pole pairs of the electric motor and then multiplied by 60. Also, a case structure (if statement) has been implemented so when the RPM is below 500 the reading will be 1 RPM, this has been introduced to avoid Labview crashing when the motor is off and there is no reading of RPM. The LabVIEW code can be seen in Figure 9.

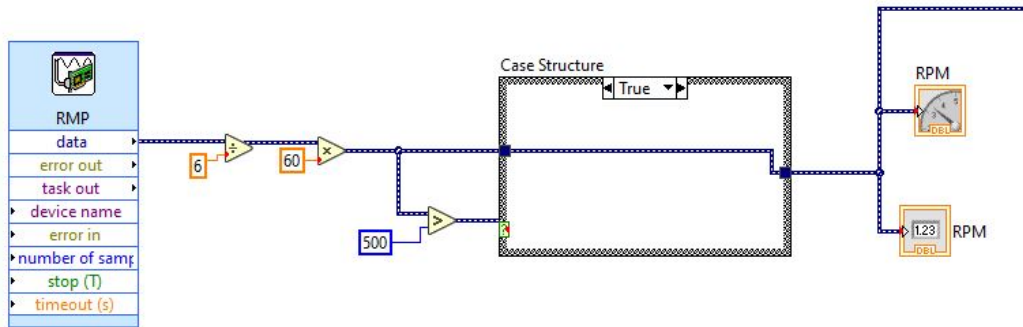


Figure 9: Labview code for RPM signal processing

3.4 Sensor Calibration

In this section, the calibration, as well as the conversion from output voltage into a physical quantity (Pressure, thrust, etc.), will be presented for each of the sensors used for the experiment. The results of these calibration procedures can be observed in Figure 5.

It is worth mentioning that all the sensors have been calibrated using a linear trendline, these trendlines are composed of a slope that is multiplying the coefficient (in our case the coefficient is the physical property) and a constant which could be adding or subtracting.

$$Y(\text{voltage}) = M(\text{slope}) * X(\text{property}) + C(\text{constant}) \quad (11)$$

For this experiment the most important part is to get an accurate slope value, the constant is not so important because its value can drift depending on the conditions, so it can be readjusted directly during the test.

3.4.1 Static Pressure Calibration

First, the static pressure sensor needs to be calibrated, the sensor is located on the pitot board and reads the pressure from the static ports of the pitot tube. In order to convert its output voltage into Pascals, the data from the sensor HCA-BARO datasheet 3 has been used. At the minimum operating pressure range of 800 [mbar], the output voltage is 0.25 [V] and at its maximum pressure of 1100 [mbar], the output voltage is 4.25 [V]. So, with the help of Excel, a linear trendline is created between the two points obtaining the calibration equation 12.

$$y = 0,0133x - 10,417 \quad (12)$$

Once the voltage is transformed in pressure the value is multiplied by 100 to get the pressure measurement in Pascals.

3.4.2 Dynamic Pressure Calibration

This sensor is also located on the pitot board and reads the pressure from the total pressure port of the pitot tube. In order to calibrate this sensor, an experimental

procedure has to be followed. The whole testing rig has to be mounted on top of the car and the calibration should be conducted with the lowest wind speed possible. The procedure is to speed up the car up to 50 [kph] and start measuring the voltage output of the sensor for 10 seconds, then the car speeds up in steps of 10 [kph] up to 120 [kph] and the same procedure will be repeated for periods of 10 seconds. It is very important to perform the same test in both directions so the error due to wind direction and speed can be neglected. All the voltage data recorded for the 10 seconds has to be averaged to get a single voltage output value, then this value must be averaged with the one obtained at the same speed but opposite direction. Following this procedure, the data obtained can be seen in Figure [10](#)

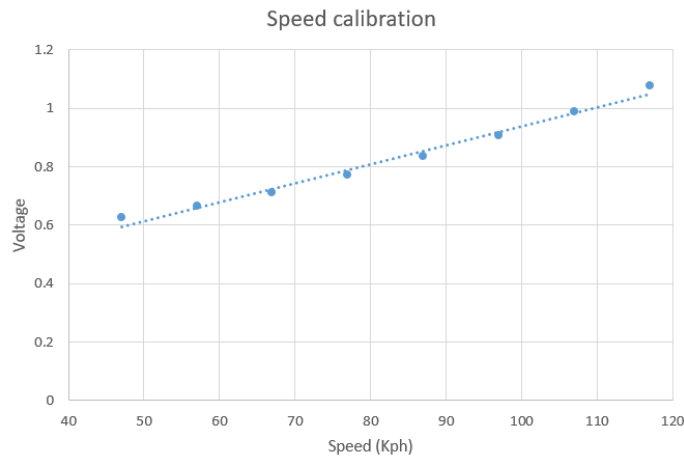


Figure 10: Trendline used for speed calibration

Finally obtaining the calibration equation [13](#)

$$y = 0.0065x + 0.2916 \tag{13}$$

3.4.3 Alpha and Beta Calibration

In order to calibrate both sensors, the output voltage of four angles was measured, the angles were 0°, 90°, 270°, and 360°. The linear trendline provided the calibration equation [14](#) obtained for both sensors.

$$y = 0,009x + 1,648 \tag{14}$$

3.4.4 Current Calibration

The current sensor has been calibrated using an external current reader, following the same procedure using excel, a linear trendline has been fitted to obtain the calibration equation [15](#)

$$y = 0,0204x + 0,6037 \tag{15}$$

3.4.5 Thrust and Torque Calibration

In order to calibrate the thrust sensor a special mount has been adapted to the sensor, thanks to this mount and the use of two 5 [Kg] weights the calibration has been made. The voltage output has been measured using no weight, then with one 5 [Kg] weight and then with two 5 [Kg] weights, with this data the following calibration equation relating voltage and thrust has been obtained:

$$y = 1.94e^{-5}x - 0.00026 \quad (16)$$

Using the same equipment, but in this case, setting up the weights to the side of the sensor and multiplying the load by the radius of the sensor so the torque could be measured, the following calibration equation has been obtained:

$$y = 1.31e^{-5}x + 0.00011 \quad (17)$$

3.5 Electric Motor Control

Two different alternatives have been tested in order to control the electric motor during the experiment, the first one has been to use the open-source electric speed control module VESC with its software VESC Tool, and the second one is using the servo controller Polulu micro maestro.

3.5.1 VESC

VESC module controls the electric motor via Pulse Position Modulation (PPM) signal and is connected to the control software VESC Tool via a USB cable.

The VESC Tool software provides a wide variety of options and different configurations, it allows for motor position control which is useful for avoiding damage to the propeller when the UAV lands, as the propeller can be set in line with the horizontal axis avoiding touching the ground upon landing.

The most useful feature for this test is the built-in RPM sweep, it allows to set an initial, final, and step RPM values and also a time step. Setting up these parameters a great variety of RPM values can be covered on just a single test run.

However, the main drawback and the reason why this method has not been used for the experiment is that the software does not allow to set a ramping time between RPM steps. The issue of having sudden changes on the RPMs is that when a propeller is mounted on the thrust and torque sensor, the strain gauges can be damaged by these abrupt changes.

3.5.2 Polulu Micro Maestro

In order to avoid these sudden changes of RPM, the Polulu micro maestro controller has been implemented, this software allows to create a script where different points in the RPM range can be saved and the time for each point can be set. The most

important feature is that the speed and acceleration of the motor can be tuned. This allows for setting low values of speed and acceleration in order to avoid any sudden changes in the RPM that can damage the strain gauges of the thrust sensor. The setting for the motor control can be seen in Figure 11

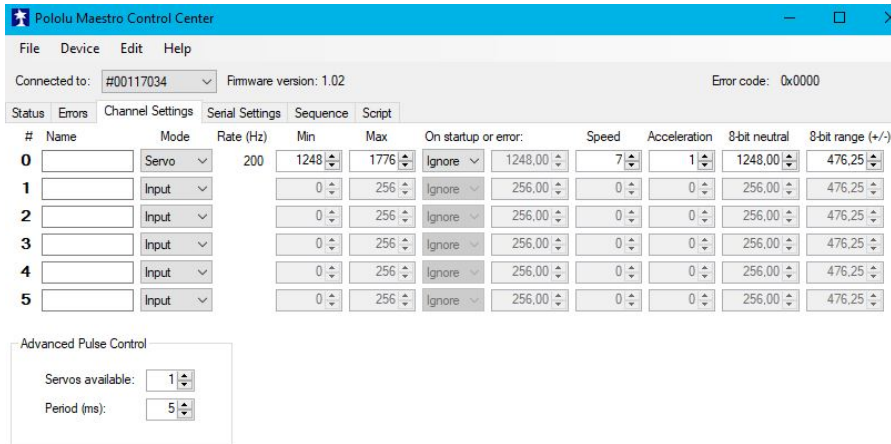


Figure 11: Channel settings for the motor control

The sequence used for the tests can be seen in Figure 12

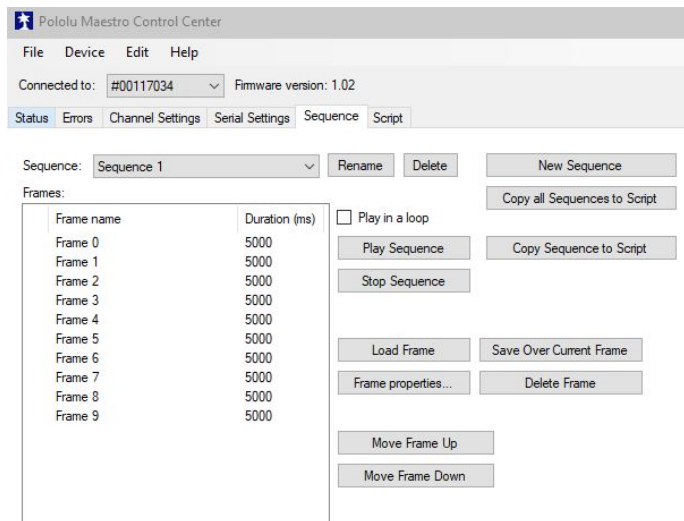


Figure 12: Sequence used for testing

3.6 Testing Rig

In order to measure the dynamic performance of a propeller, it needs to be tested against the entire velocity and rpm range over which it is intended to operate. Due to the unavailability of a wind tunnel big enough to accommodate the diameter of

the propellers a testing rig has been assembled. The purpose of this testing rig is to mount it on top of a car so the electric motor, propeller, and sensors can be exposed to an incoming airflow. The car speed is set using the cruise control functionality which keeps the car at a fixed velocity which is measured respective to the ground. However, the relative wind speed measured through the pitot tube determines the airspeed that the propeller is facing. The propeller is mounted 2 meters above the car's roof in order to get the smoothest airflow possible, avoiding the turbulence created by the car itself.

The rig itself consists of a steel structure that can be attached to the roof rack of a car. On the top part sits the electric motor and propeller attached to the thrust and torque sensor as it can be seen in Figure 13, at the same level but located to its right sits the pitot tube and pitot board and also the temperature and angle sensors. Also, it can be seen in Figure 13 that there is a wood platform below the motor mount where the batteries that power the electric motor and the pitot board are located.



Figure 13: Motor and propeller mounted on the testing rig

Finally, on the base of the structure, there is another platform where the NI modules and the battery that powers them are located, also a cover for the NI chassis has been implemented to avoid any damage during testing, Figure 14

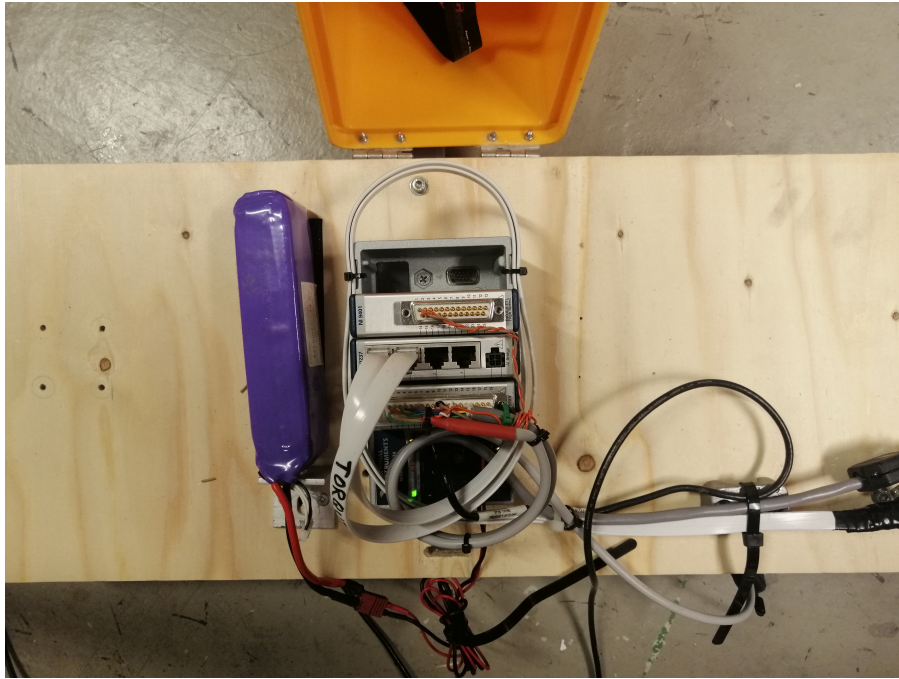


Figure 14: Base platform for the NI module and their battery

The complete testing rig mounted on the car roof can be seen in Figure [15](#).



Figure 15: Testing rig mounted on car roof

3.7 Test procedure

The test has been conducted on the 16th of June 2021, at night between 22:00 and 24:00. The test is conducted at night for two main reasons, during those hours there is less traffic on the testing road, this is important because cars in front create turbulence that can affect the obtained results and also they can be an obstacle to keep the constant speed needed for the test runs. The second reason is that during night time the wind speed is the lowest of the whole day, so it will have less impact on the results. The road chosen is a flat two-way road without buildings or structures on the sides that could affect the incoming airflow.

The testing procedure has been the following. First, with the car stopped the propeller has been mounted on the testing rig, then the car has been accelerated up to the desired testing speed, for this experiment, as the propellers need to be tested for the velocities that they would face when mounted on the UAV, three speeds have been tested 77 [km/h], 97 [km/h] and 117[km/h]. When the car reaches the desired velocity the cruise control is activated, and when the car speed is stable the Labview data log is started as well as the RPM sweep sequence. This procedure is repeated for the three velocities for each propeller, so a total of 9 runs.

Two people are needed for the test, the driver and the co-pilot. The co-pilot is in charge of starting the log-in and RPM sequence so the driver can focus only on the road. During the data acquisition phase, the co-pilot needs to monitor the values that are being obtained constantly.

3.8 Troubleshooting

In this section, all the issues and errors that have been found during the project will be explained, as well as how they have been solved.

3.8.1 No RPM reading making Labview crash

The first issue found with the software was regarding the RPM readings. When the motor was switched off (0 RPM) and the Labview code was run to get data, it crashed immediately, stopping the data acquisition. In order to fix the issue and be able to record data when the motor wasn't spinning a case structure has been implemented which is equivalent to an if-statement on graphical programming. The case structure has been set in a form that when the RPM measure drops below a certain value (for this test the value was set to 500 RPM) it outputs always a value of 1 RPM, in this way Labview always has an RPM input and it does not crash. The case structure can be seen in Figure [16](#).

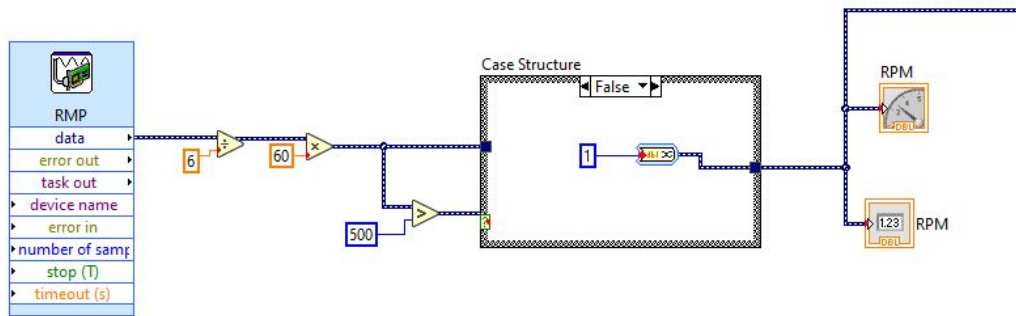


Figure 16: Case statement displaying the False case

3.8.2 Buffering issue causing breaks on the data login

One of the main issues that have been identified during the testing of the rig was related to the data login rate.

During these tests, the data log rate was set at a frequency of 1K Hz and a number of 1000 samples to be acquired on each loop iteration. The expected result was to get a continuous data login at a 1K Hz rate, however, the results obtained presented considerable gaps in between measurements as can be seen in Figure 17.

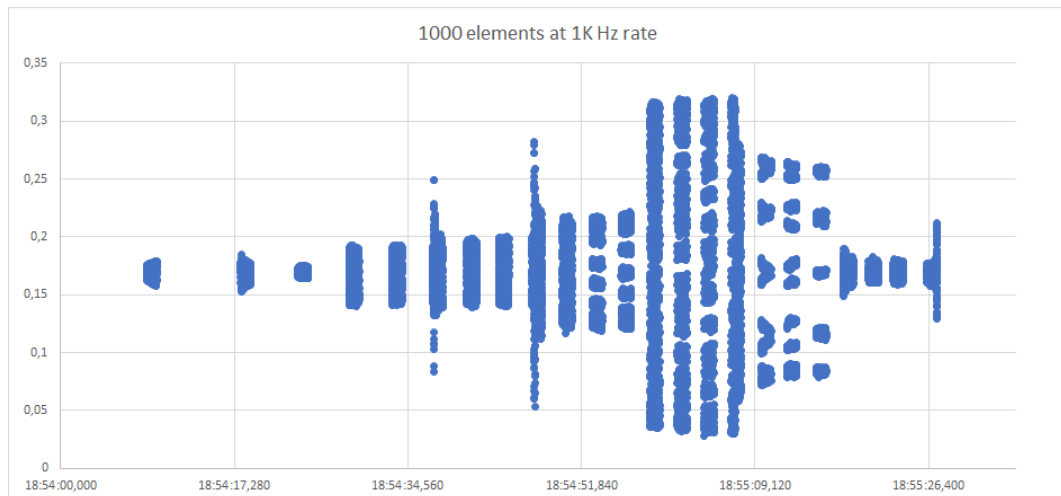


Figure 17: Thrust values when logging 1000 elements at a 1K Hz rate

Analyzing the data presented in Figure 17, it was concluded that the LabVIEW was recording 1000 measurements at a 1K Hz rate, and then it stopped logging data until a certain time when it started logging 1000 elements again and then stopping, this process was being repeated through the whole test. The issue has been identified as a buffering problem, LabVIEW is not able to write on the excel file all the data acquired fast enough to keep up with the logging rate, so the while loop in which the whole code is set, does not restart until the 1000 measurements are written on the spreadsheet creating the data gaps that can be seen.

The solution to this problem has been reducing the number of elements from 1000 to 100, thanks to this reduction the writing into the spreadsheet time are considerably reduced reducing the gaps as well, this can be seen in Figure 18.

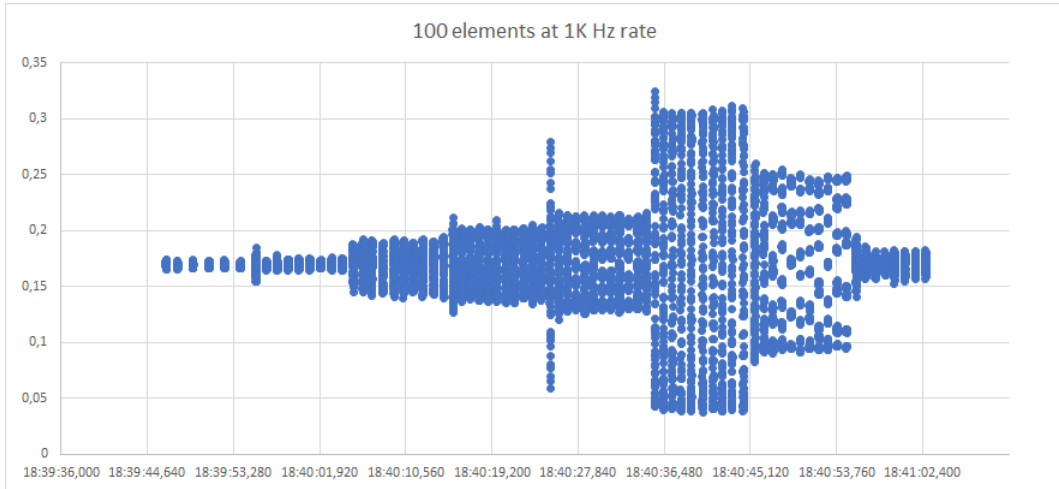


Figure 18: Thrust values when logging 100 elements at a 1K Hz rate

Other possible solutions to this problem are proposed in section 6.2.

3.8.3 Thrust and Torque measurement error

The first option to measure the thrust and torque was to connect the sensor through an RJ50 cable to its own National Instruments module NI 9237. However, after the first tests noise in the measurements was found, as can be seen in Figure 19.

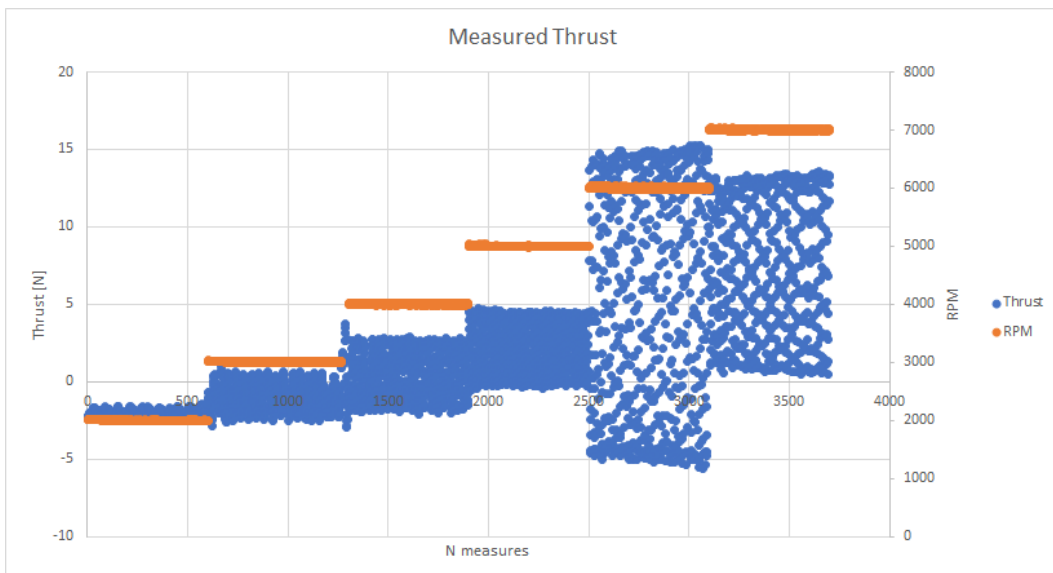


Figure 19: Measured Thrust through RPM sweep

This noise in the signal could be caused by electrical interference created by the VESC module or mechanical vibrations created by the motor and propeller spinning. In order to identify the origin of the issue, the same test was carried out, but in this case, the electric motor with the propeller was dismantled from the thrust sensor and was run on the side, obtaining the results of Figure 37.

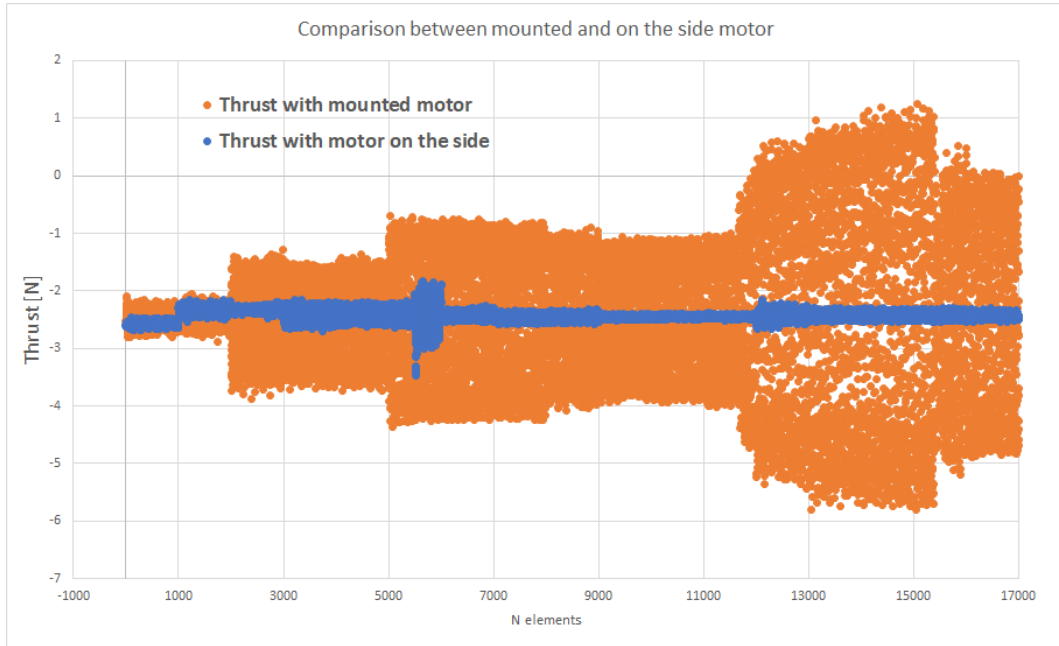


Figure 20: Proof that the oscillations come from mechanical vibrations

Looking at the comparison made in Figure 37 it can be seen that when the motor is dismantled of the thrust sensor the oscillations are considerably lower. This confirms that the noise is coming from mechanical oscillations produced by the motor and propeller and not from electrical interference from the VESC module.

In order to investigate deeper into the issue, another configuration has been tested. Connecting the thrust and torque sensor to a signal amplifier and then connecting this amplifier to the NI 9205 module, so the input now is an analog signal and the NI 9237 is not used any longer. As it can be seen in Figure 21 there aren't any substantial changes, so in order to simplify the setup of the testing rig, the NI 9237 without amplifier has been used.

To obtain accurate data of the thrust and torque values the solution implemented has been setting the log rate at a 1000 Hz, in this way when the propeller is spinning at its maximum RPM of 9000 which translates to 150 *rev/s* creating the mechanical oscillations, the logging rate is 7 times higher. This allows for mapping correctly the sinusoidal oscillation in order to be able to average the values obtained and get a correct Thrust value for each RPM rate.

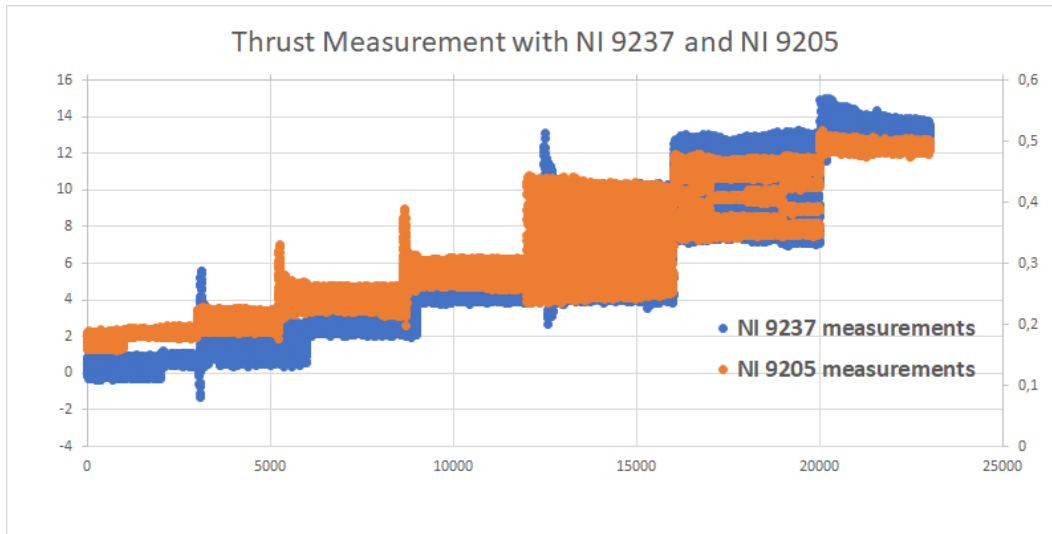


Figure 21: Thrust measured with the NI 9205 and NI 9237

3.8.4 Electric Motor and ESC Failure

During the testing runs, when the last propeller APC 20x10 PN was being tested, the electric motor and the ESC failed making it impossible to continue with the experiment. It is suspected that the cause of this failure is related to the windmilling phase of the propeller before starting the test.

When the car is being accelerated up to the required speed and the electric motor is not driving the propeller, the incoming wind causes the propeller to spin. When this happens the motor acts as a generator instead, depending on the ESC used this energy can be used to recharge the batteries or if the ESC does not have this function the energy generated is rejected as heat.

In our case, when the car reached the 97 [km/h] desired the running sequence was started, and as soon as it started the motor and ESC burned. It is believed that this happened because of the abrupt transition between generating energy and applying it to the propeller.

Two possible solutions have been discussed for this problem, the first one is to accelerate the car with the propeller fixed in position to avoid windmilling, in order to do so the control software must be able to fix the position of the propeller and then transition to the running sequence. Another option is to have the motor engaged at all times when the car is speeding up, avoiding the abrupt change between generator and running motor.

4 Results

In this section, the acquired data for each propeller will be presented

4.1 Propeller APC 18x10

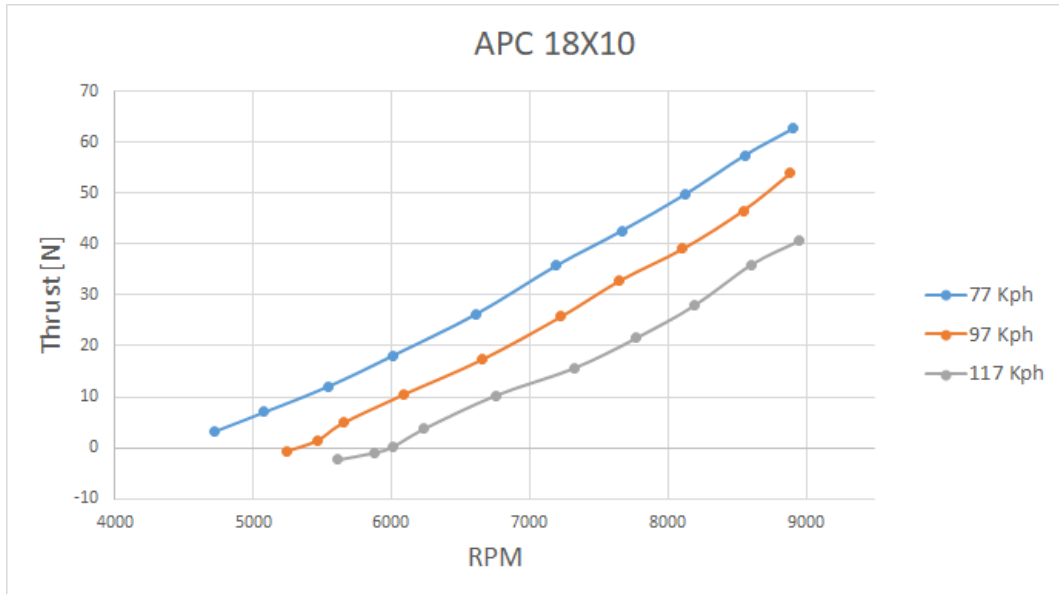


Figure 22: Thrust generated through the RPM range APC 18x10

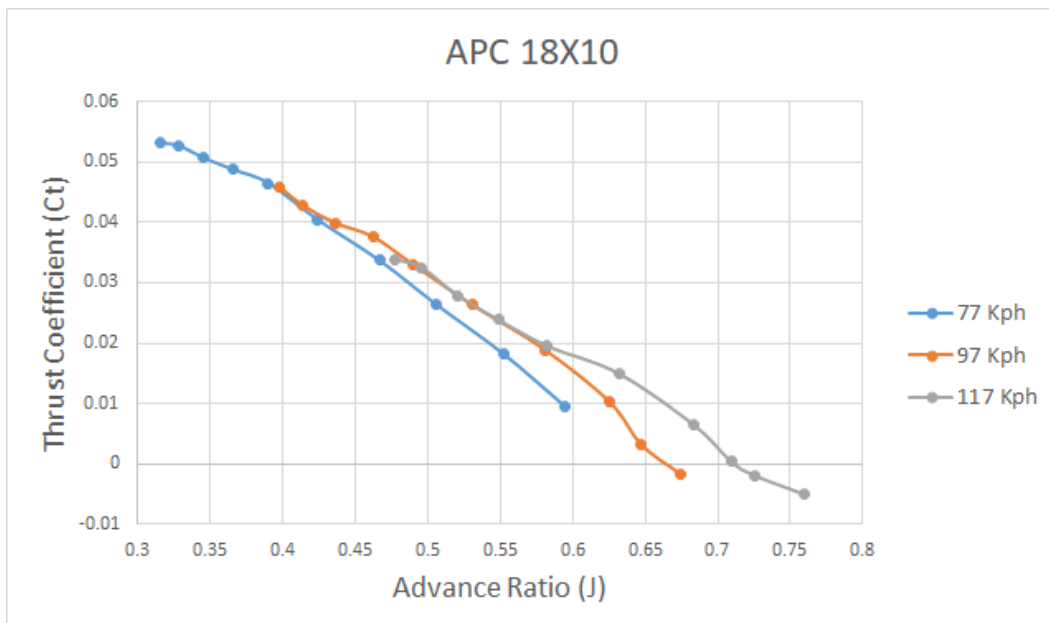


Figure 23: Thrust coefficient vs. Advance ratio APC 18x10

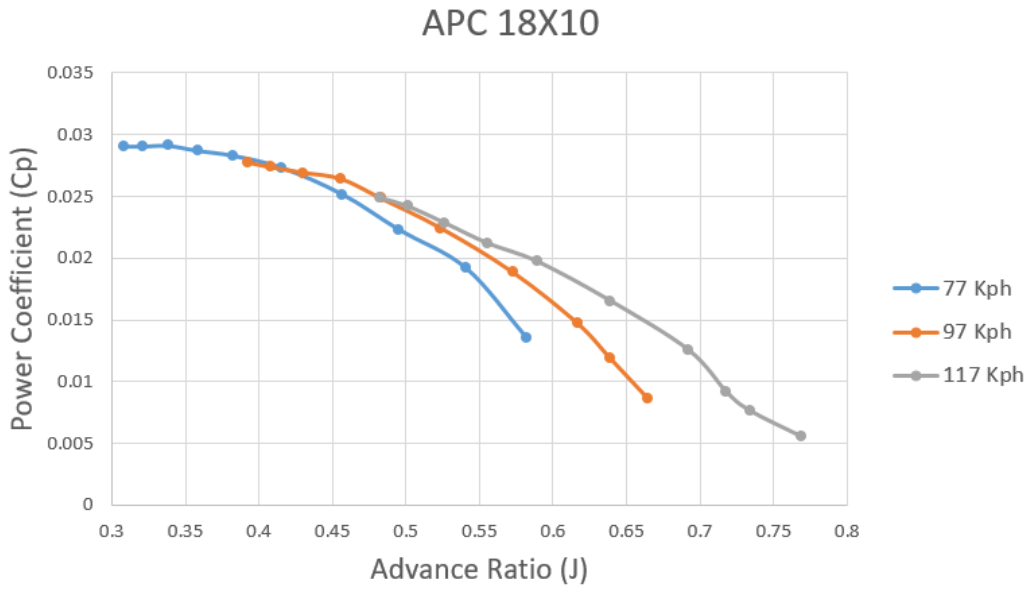


Figure 24: Power coefficient vs. Advance ratio APC 18x10

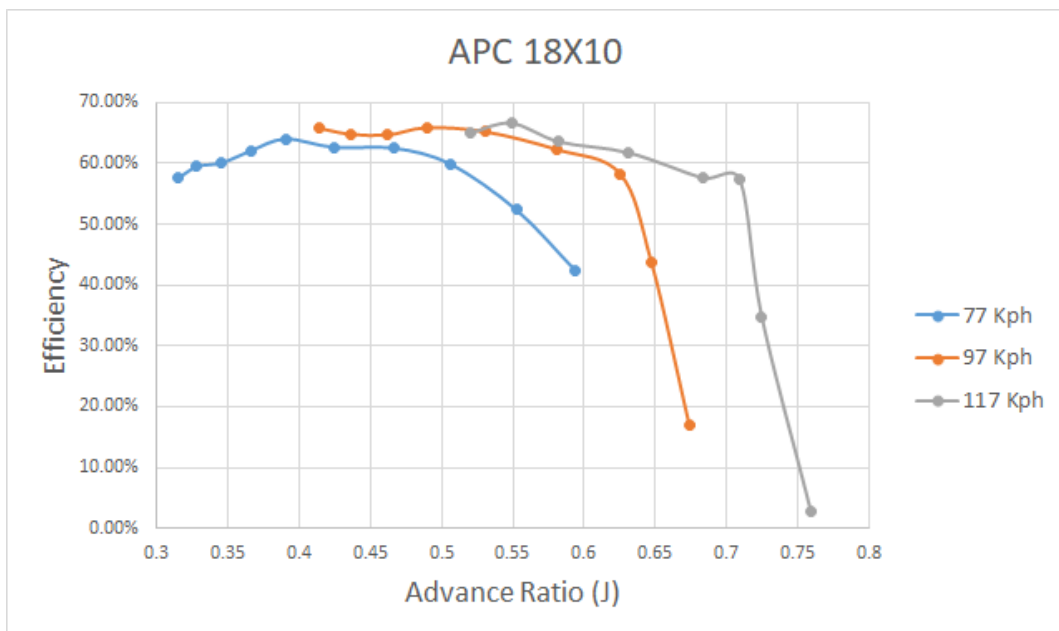


Figure 25: Efficiency vs. Advance ratio APC 18x10

4.2 Propeller APC 18x12

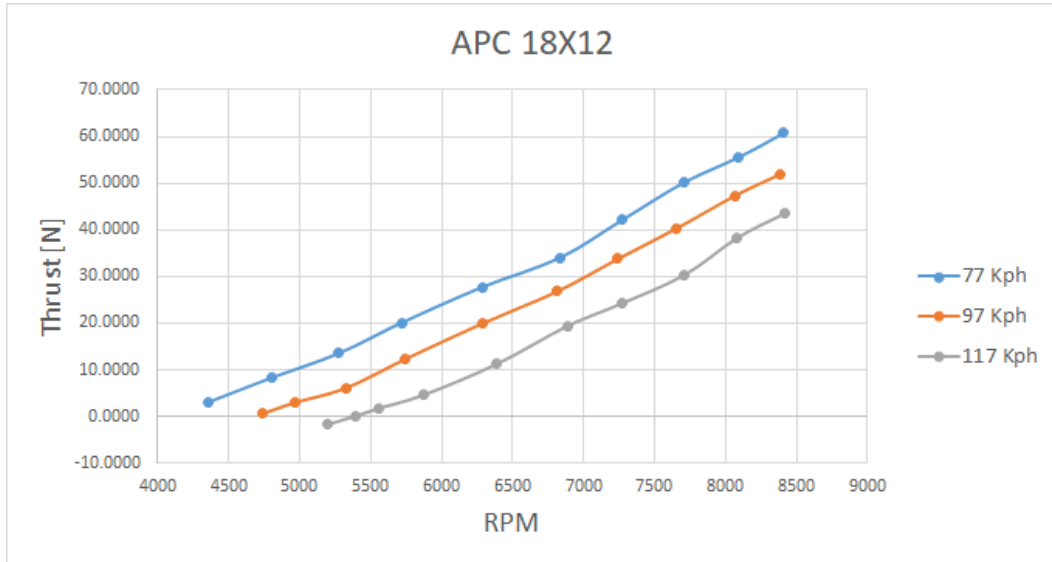


Figure 26: Thrust generated through the RPM range APC 18x12

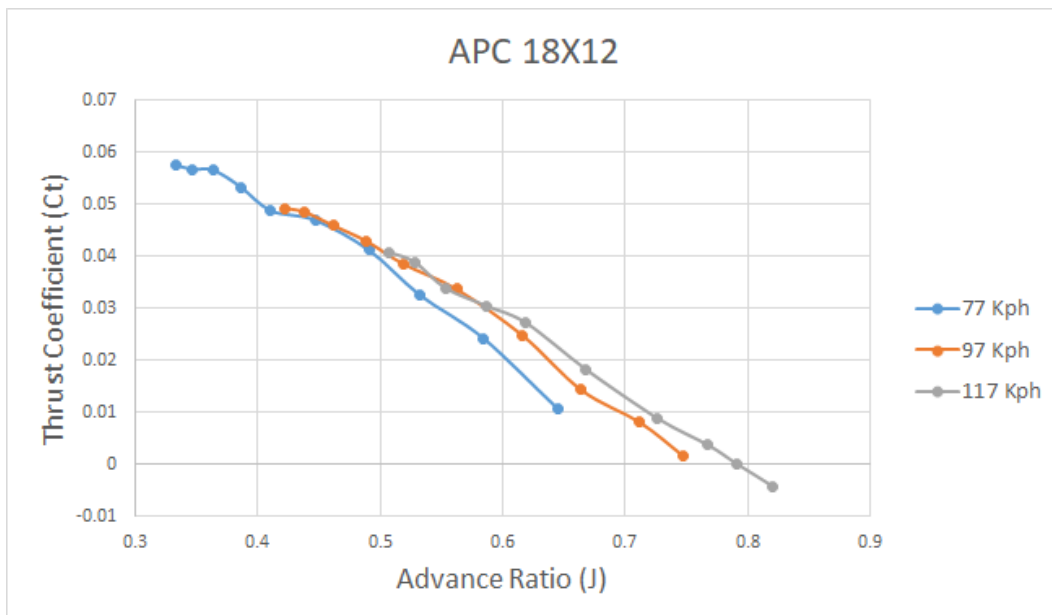


Figure 27: Thrust coefficient vs. Advance ratio APC 18x12

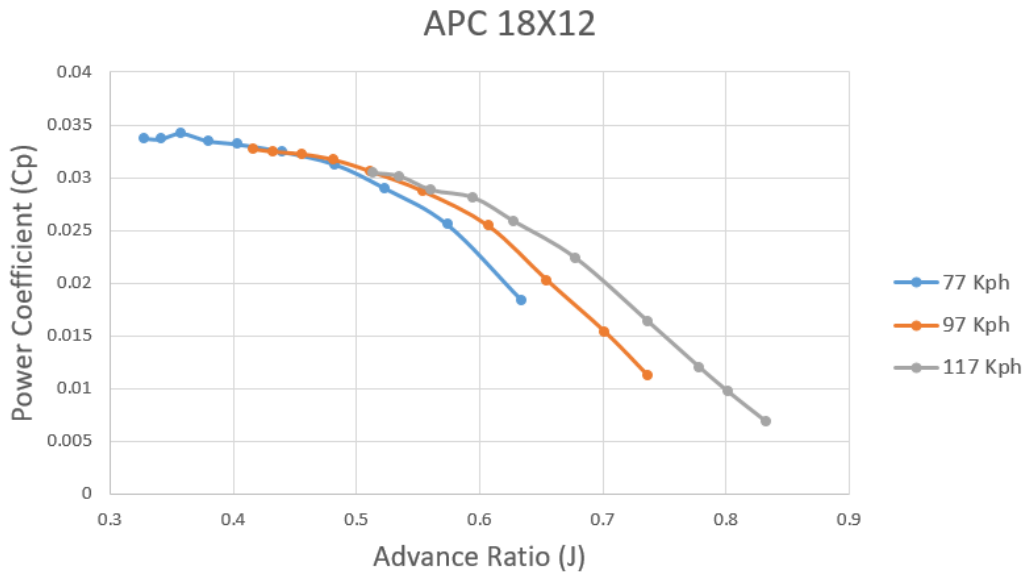


Figure 28: Power coefficient vs. Advance ratio APC 18x12

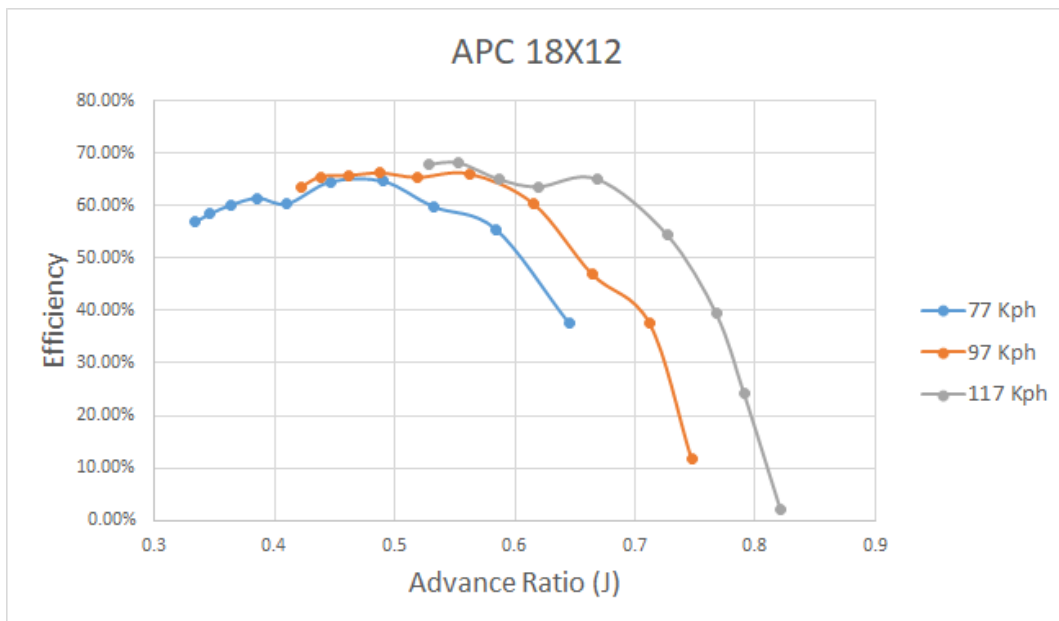


Figure 29: Efficiency vs. Advance ratio APC 18x12

4.3 Propeller APC 20x10 Pusher

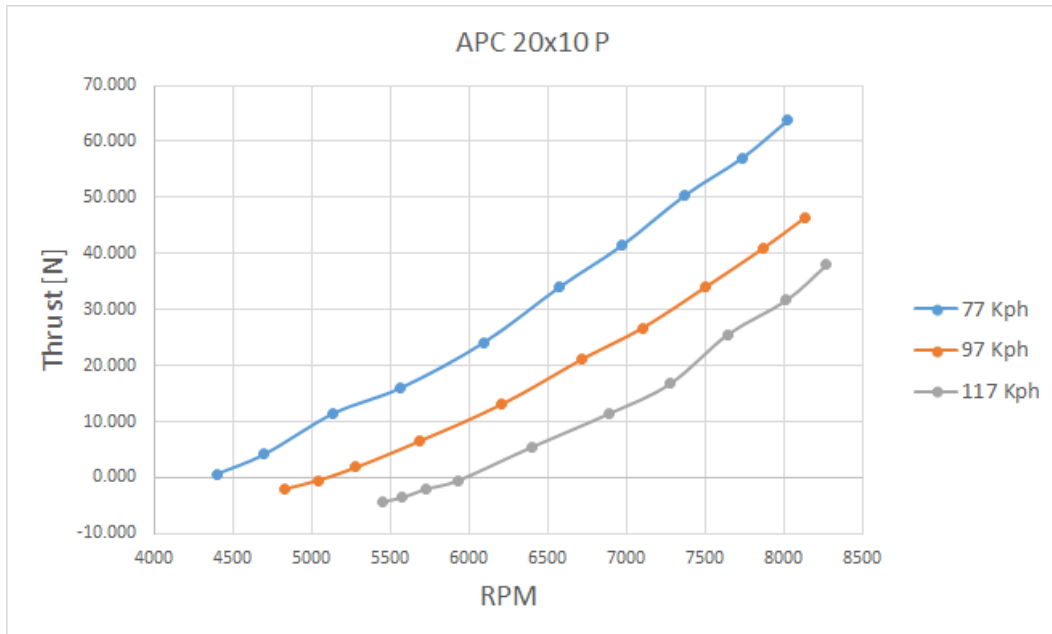


Figure 30: Thrust generated through the RPM range APC 20x10 P

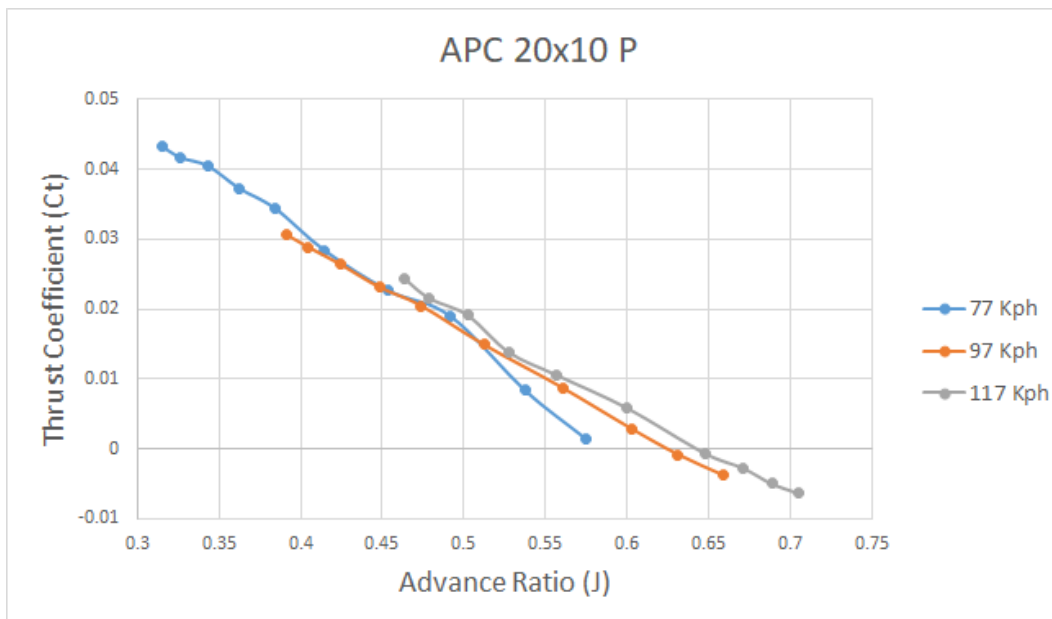


Figure 31: Thrust coefficient vs. Advance ratio APC 20x10P

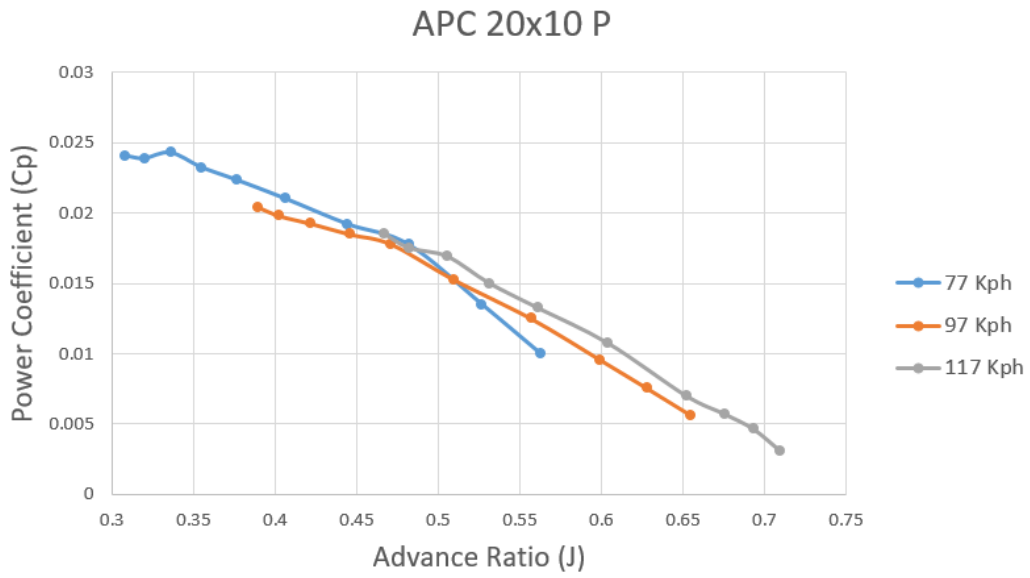


Figure 32: Power coefficient vs. Advance ratio APC 20x10P

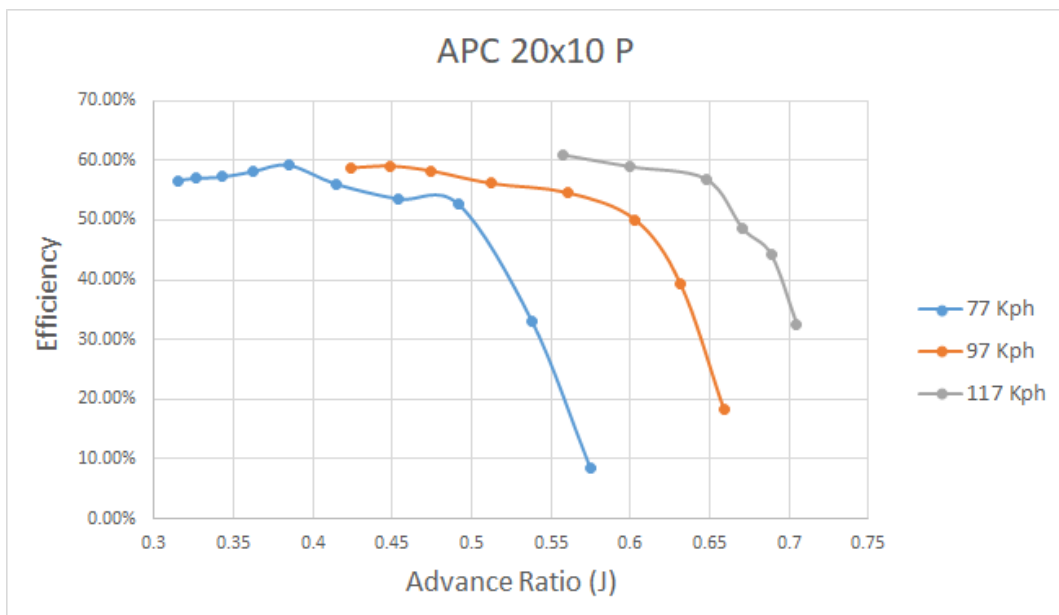


Figure 33: Efficiency vs. Advance ratio APC 20x10P

4.4 Propeller APC 20x10 PN

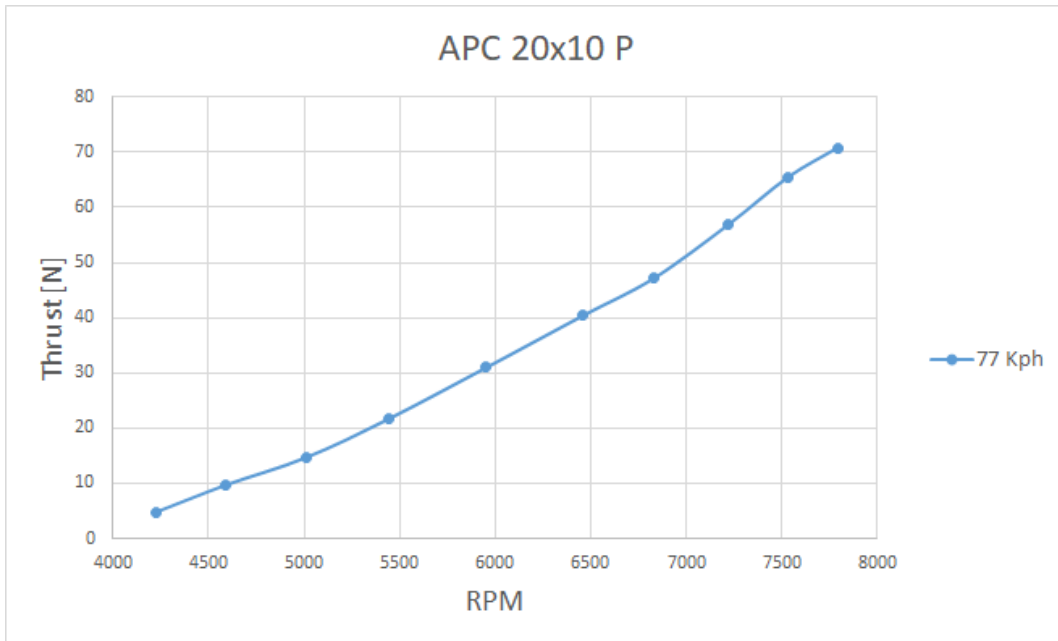


Figure 34: Thrust generated through the RPM range APC 20x10 PN

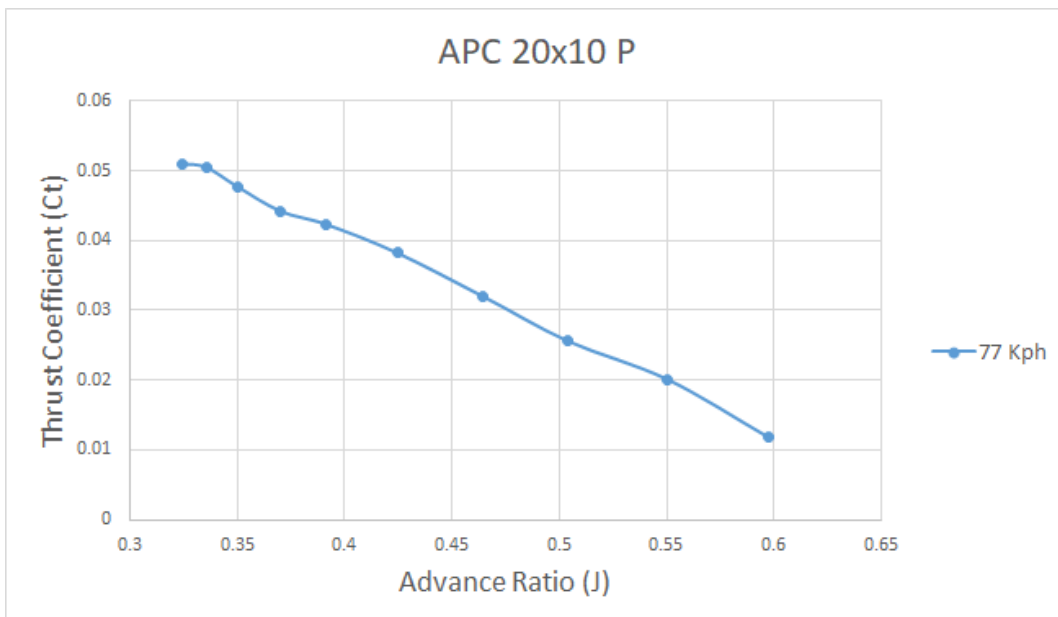


Figure 35: Thrust coefficient vs. Advance ratio APC 20x10 PN

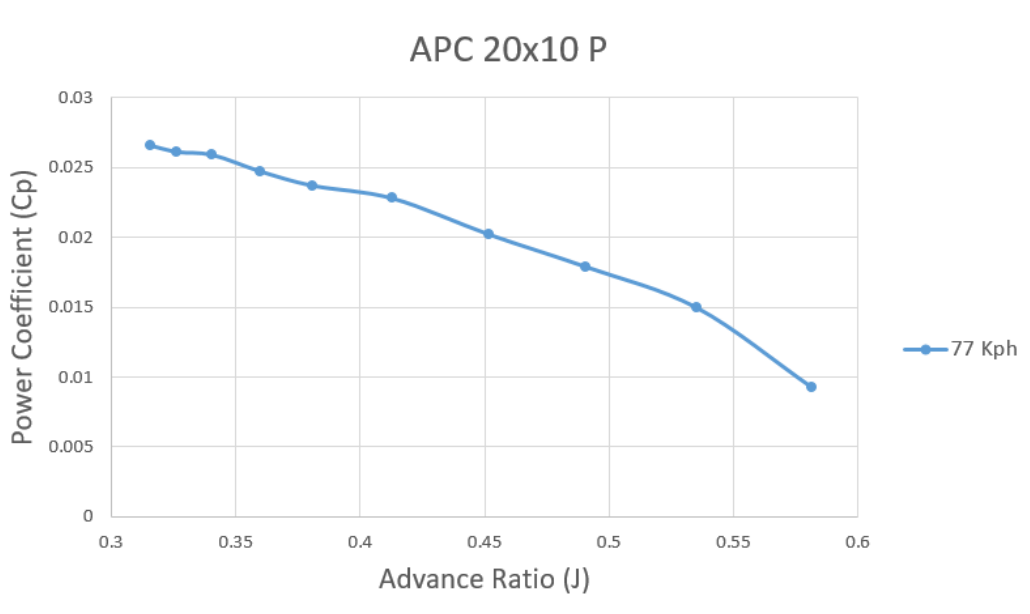


Figure 36: Power coefficient vs. Advance ratio APC 20x10 PN

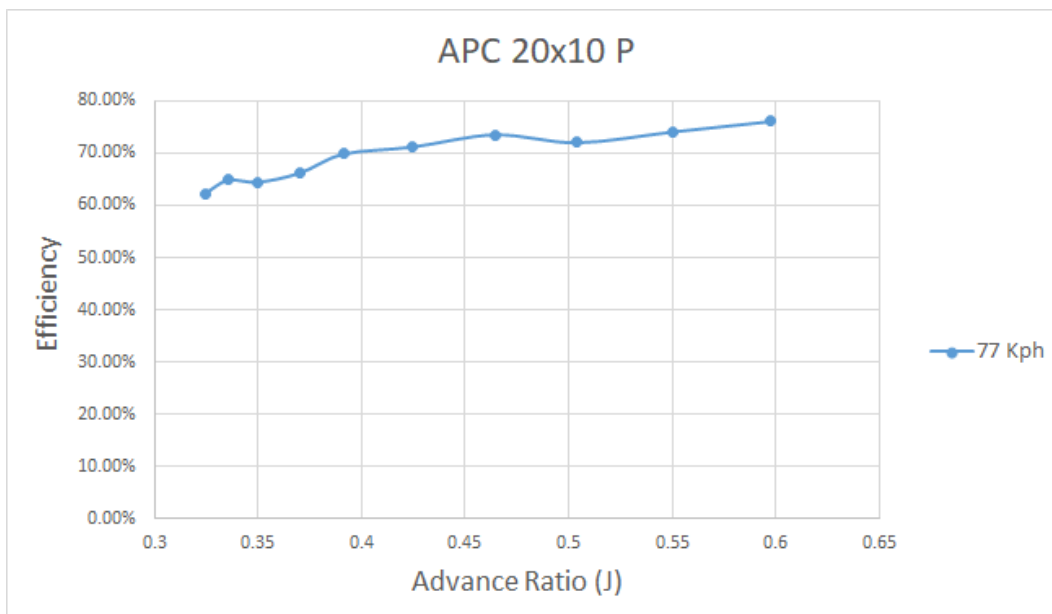


Figure 37: Efficiency vs. Advance ratio APC 20x10 PN

5 Conclusions

Along this thesis, an alternative methodology has been implemented in order to obtain the dynamic performance of different propellers, the use of a car-top testing rig has proven to be efficient and effective compared to a conventional wind tunnel resulting in a cheaper and more accessible alternative. All the necessary equipment,

as well as the software and the procedures implemented, have been thoughtfully explained. Also, all the errors and drawbacks that have been encountered are presented in this report, with the implemented solutions.

The objective of the experiment has been fulfilled, three different propellers have been tested, obtaining all the needed data to model their dynamic performance, their thrust and torque values for the different incoming flow velocities, and RPM have been needed for computing the efficiency of each propeller. Also, as explained in section [2.2](#), the thrust obtained by one propeller can be used in the future to calculate the Drag force of the UAV that will mount the propeller.

To sum up, this thesis presents a low-cost alternative to wind tunnels that allow obtaining the dynamic performance of different propellers, which can be used to calculate the drag of a specific aircraft and the efficiency of the power plant.

6 Perspectives

In this final section, different suggestions regarding ways of improving the experiment will be given, these can be implemented in the future as an upgrade.

6.1 GPS implementation

The addition of GPS data logging to the testing experiment will be a considerable upgrade. This GPS data will allow obtaining the ground speed of the testing rig accurately, this will allow to compare it with the relative speed obtained from the pitot tube measurements, by doing so the real wind speed could be calculated.

The key aspect is to be able to log the GPS data into the Labview software so it is displayed along with all the measurements on the excel file, which facilitates considerably the data post-processing stage.

There are three possible options compatible with the testing rig. The first and most simple option is to use the VESC android app, this app allows for the VESC module to be connected through Bluetooth with a smartphone inside the car, once it is connected, the app has the option of logging all the motor data (RPM, current, etc.) together with the GPS information of the smartphone on an excel file for each test run. There are two drawbacks, first in order to connect the VESC to the phone a Bluetooth module has to be bought, connected, and configured. However, the main drawback is that with this method the GPS data won't be logged through Labview in the same file as the sensor measurements, so the logging frequency will be different and thus incompatible. For this reason, this method has been discarded.

The two following methods solve the previous issue, as the GPS data is logged through Labview. The first option is to purchase the National Instruments module NI myRIO and connect it to the GPS receiver Digilent Pmod GPS, with this method the GPS data can be easily logged and processed through Labview, however, the cost of implementing this solution is considerably high, the NI myRIO module cost is kr 4.995,00 as it is stated on the official NI web page [\[4\]](#), and the cost of the GPS receiver is 450 kr [\[5\]](#).

The last and most suitable option is to implement a Laptop GPS module, which would be connected through a USB port directly to the Laptop. In order to read the GPS data on the Labview software, the API NI-VISA needs to be downloaded and installed. NI-VISA is an API that provides a programming interface to control USB instruments in NI application development environments like LabVIEW, one strong candidate for the GPS receiver could be the GlobalSat BU-353-S4 USB GPS Receiver, with a cost of 300 kr.

6.2 Possible solution for Buffering

Regarding the buffering issue that has been solved in section [3.8.2](#), there are different ways to solve the problem that would be really beneficial to implement in the future. The issue is caused by the slow writing rate of the .xlsx (Excel) file, so one of the strongest options is to change the type of file on which the data will be stored, LabVIEW offers three options:

- Text (LVM)
- Binary (TDMS)
- Binary with XML header (TDM)

According to National Instruments, the fastest option would be to write the data on a binary file (TDMS). However, this type of file difficulties the data post-processing stage, so a conversion software between binary and .xlsx format would have to be implemented in order to be able to process the obtained data. The other two options are also faster than a .xlsx file but as well as the Binary file they difficult the post-processing stage.

Also, there is a second viable option, writing a Labview code without using the DAQ Assistants [\[6\]](#), in this way there is access to more options and parameters that can be modified in order to solve the buffering issue. However, the code will be significantly more complicated.

References

- [1] Lundström D, Krus P. Testing of Atmospheric Turbulence Effects on the Performance of Micro Air Vehicles. International Journal of Micro Air Vehicles; 2012.
- [2] Performance of Propellers. Massachusetts Institute of Technology; 2017.
- [3] HCA-BARO Series Miniature amplified barometric pressure sensors Datasheet. First Sensor, Sensor Technics;. https://www.first-sensor.com/cms/upload/datasheets/DS_Standard-HCA-BARO_E_11641.pdf.
- [4] myRIO Student Embedded Device;. <https://www.ni.com/sv-se/shop/hardware/products/myrio-student-embedded-device.html>.
- [5] Digilent Pmod GPS: GPS Receiver;. <https://store.digilentinc.com/pmod-gps-gps-receiver/>.
- [6] Rivas DD. A low cost approach to dynamic propeller performance measurement. Linköping University; 2017.

A First appendix

A.1 Electric Motor Specifications

- Turns: **16 T**
- Voltage: **10S LiPo**
- RPM/V **213 kV**
- Internal resistance: **0.023 Ohm**
- Max Loading: **65 A**
- Max Power: **2550 W**
- Shaft Diameter: **8.0 mm**
- Bolt holes: **32 mm**
- Weight: **706 g**
- Motor Plug: **4 mm Bullet Connector**