

Group 4 - Optimisation of Drone Deliveries

*Subsystem 1: Adedunke Oyenuga; Subsystem 2: Chris Vail;
Subsystem 3: Theo Kane; Subsystem 4: Alexandre Berkovic*

Abstract

Today, Amazon are pioneers in drone delivery, a sector in which Imperial College's design engineers could easily get involved. This report carries out a system design problem aimed at reducing the overall power consumption of drone trip between the Dyson building and the White City Hackspace for the delivery of 3D printed parts.

In doing so, four sub-systems were devised: frame, motor, propellers and flight trajectory alongside a battery-aware model. For the frame and the motor, mass was optimised as it is directly linked to the drone's power consumption. The frame was optimised to weigh only 318g by accounting for the geometries of the subsystem as well as its material. The motors were optimised to a mass of 46.5g per motor while ensuring that the required torque and rpm can be achieved. Then, the propulsion system was optimised by minimising resistance torque whilst still producing sufficient thrust. Finally, flight trajectory in a given wind environment was optimised, allowing to determine the state of charge of the battery at the end of the flight. This equates to a battery energy depletion of 64% against 80% for a straight path. Overall, on a system level, the aforementioned sub-systems are optimised to substantially decrease power consumption of the quadcopter. However, these optimisations might be non commutable which means that they cannot produce a tangible global system optimisation result.

1 Introduction

Engineers strive to build systems which make their life easier. Leading up to stressful deadlines, students may stay for hours in the workshop. During hands-on projects, additive manufacturing is a great way to prototype a given model. However, when submissions draw near, students are confronted with the issue of there not being enough 3D printers to use. Going to the Hackspace can solve this issue but requires a 30-minute commute from the South Kensington campus. Hence, the goal of this project is to devise the optimal drone to deliver the 3D printed parts. This would mean travelling about 4km in the shortest time frame possible whilst depleting the smallest amount of energy.

A key drawback hindering the success of most drone delivery systems is battery life. Consequently, the idea behind this optimisation problem is to reduce power consumption as much as possible so that the drone can safely make the desired trip. There are two types of battery models that exist for this intended purpose: short-use and fast-charging or long-use and slow-charging. For the sake of having the 3D printed parts rapidly, we will be looking into the first scenario. Out of the existing studies on drone optimisation, most are related to altitude optimisation for image reconstruction [1], multi-trajectory optimisation [2] and multi-objective optimisation for autonomous drone's solar energy

charging stations [3]. Currently, none of them look to optimise a drone for a single journey in a specific setting.

2 System Analysis and Decomposition

$$\begin{aligned} \min_{\vec{x}} \quad & f(\vec{x}) = P \\ \text{where} \quad & \vec{x} = (P_e, N_p, l_p, p_p, m_d, v, u, t_t) \\ & P = \eta P_e \\ & t_t \sim P \\ \text{s.t.} \quad & h_1 : m_d - (m_b + m_f + m_p + m_m) = 0 \\ & h_2 : N_p - 4 = 0 \\ & h_3 : x - 4000 = 0 \\ & h_4 : y - 2000 = 0 \\ & h_5 : m_p l - 200 = 0 \\ & h_6 : g - 9.81 = 0 \\ & g_1 : l_p - l_a \leq 0 \\ & g_2 : z - 100 \leq 0 \\ & g_3 : \sqrt{v^2 + u^2} - 30 \leq 0 \\ & g_4 : u - 20 \leq 0 \\ & g_5 : v - 20 \leq 0 \\ & t_t - 1200 \leq 0 \end{aligned}$$

The negative null form looks at minimising power consumption which is a function of variables in \vec{x} and linked to travel time. Constraint $h_1 - h_5$ relate the masses together, h_2 sets the number of propellers of the drones, h_3, h_4 and g_2 give boundaries for the drone's flight, g_1 relates length of the propeller to length of the arm, $g_3 - g_5$ are all related to winder velocity and g_6 bounds total flight time of the drone.

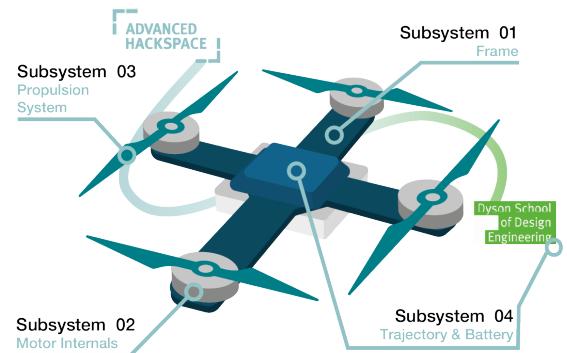


Figure 1: Subsystem Breakdown

System Decomposition

Design variables relevant to drone optimisation were pinpointed and devised into subsystems (Figure 2). Those subsections of the model are what informed us to choose the different subsystems we created.

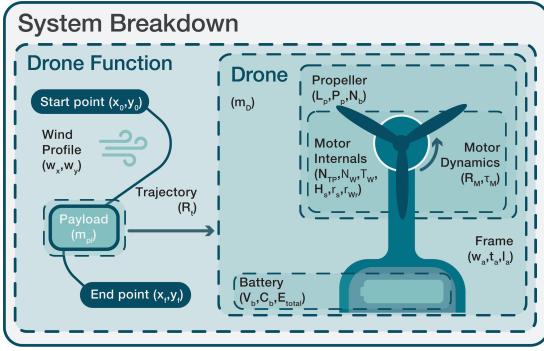


Figure 2: System boundary diagram

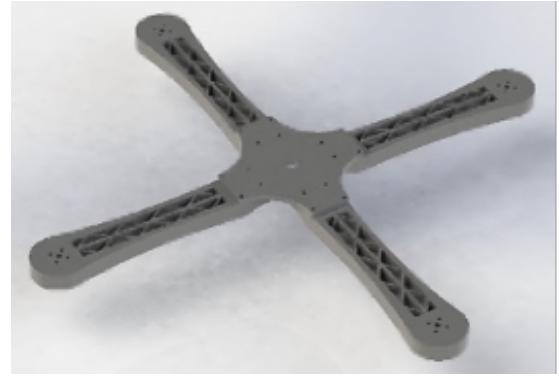


Figure 3: Frame Modelled in SOLIDWORKS

Subsystem Interdependencies

Due to the nature of drones, all the subsystems are linked, primarily through the mass of the drone. The propellers are required to produce sufficient thrust in order to lift and accelerate the mass of the drone and are also strongly linked to the motors used to drive them. Attempting to match the peak efficient torque and angular velocity of the propeller and motor is imperative to the design a high-performance system. One of the key design variables for the efficiency of the propeller is the blade length, the maximum of which is informed by the geometry of the frame. The combination of the propeller and motor efficiencies define the ability of the drone to transform electrical energy into mechanical work. As such, increasing efficiencies result in an increased range. Another way to increase the range is to select the optimal trajectory which is limited by maximum thrust the propellers can provide as well as the overall weight of the drone and the route the quadcopter takes.

Subsystem Trade-offs

One of the key trade-offs is between the drone mass and the length of propeller. Longer propellers provide more thrust but require a larger, and therefore more massive frame, in addition to requiring a higher motor torque. High torques lead to increased motor mass and higher current draw, reducing the drone range. The optimal trajectory is one that minimises the energy consumption - this is not always the same as the shortest distance travelled as wind can impede the drone's motion. By knowing the efficiency of the drone and the power consumed across optimal paths we can select batteries which meet the required capacity and power discharge rates. There is a trade-off here as smaller batteries can be used for less time but weigh less and charge faster. Optimising this trade off would require detailed knowledge of drones use case and operation frequency which is outside the scope of this report.

3 Subsystem 1: Frame

3.1 Problem Formulation

The subsystem's design objective was to minimise the mass of the drone frame. This is because this minimisation will significantly reduce the drone's power consumption and therefore can increase the drone's number of trips. It also improves the drone's overall performance by improving its manoeuvrability and increasing its payload capacity. Due to the symmetry of the drone frame, it was decided that only a quarter of the quadcopter would be examined. For simplicity, the geometric design of the base plate was fixed. This meant that only a single arm would be optimised.

Variables

Design variables were selected by determining what would heavily influence the frame's mass. These were largely geometric- arm width, arm thickness and arm length were chosen. Material selection (density) was also selected as a discrete variable.

Constraints

Various constraints were put into place to ensure the problem was well-bounded and that the frame met industry standards. These constraints have been split into 2 types: functional and physical.

g_x	Functional Constraint
g_1	Propellers should not clash with each other
g_4	Deflection of the arm should not exceed 1% of the frame's thickness to prevent structural failure

Table 1: Functional Constraints

g_x	Physical Constraint	Justification
g_2, g_3	Arm Width	LB: Prevents conflict with arm's inner geometry. UB: Industry standards [4]
g_5, g_6	Arm Thickness	LB: Keeps thickness of drone uniform. UB: Industry standards [4]
g_7, g_8	Arm Length	LB: Prevents conflict with arm's inner geometry. UB: Industry standards [4]

Table 2: Physical Constraints

Design Parameters

Due to the nature of the optimisation and fixing the design of the frame's base plate, certain dimensions had to be accounted for.

h_x	Parameter
h_1	$l_b = 117.79$
h_2	$l_p = 180$
h_3	$r_a = 22.5$
h_4	$t_b = 5$

Table 3: Parameter Values

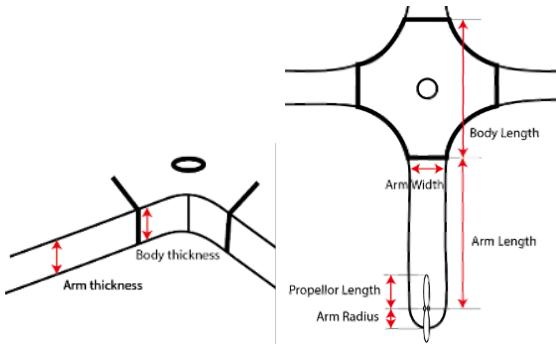


Figure 4: Model detail

Material	β_1	β_2	β_3
Aluminium	0.091	1.542	0.168
6061 Alloy			
ABS PC	0.186	0.972	0.233
Nylon 101	0.122	0.880	0.344
Carbon Fibre (Hexcel AS4C (3000 filaments))	-0.736	1.123	-0.137

Table 4: Normalised β values

	w_a	t_a	l_a	ρ
f	-	-	-	-
g_1				-
g_2	-			
g_3	+			
g_4		-		
g_5		+		
g_6		-		
g_7			+	
g_8			-	

Table 5: Monotonicity Analysis Results

Linear Regression

Before undertaking linear regression on the data, all data was normalised. This made the data more comparable. The regression was developed in python and the data was split 75:25 into a training and test set. Each metamodel confirmed the design variables' linear relationship with the arm's mass. The normalised beta values obtained for the equation are displayed in Table 4.

Overfitting of the models was checked by obtaining Root Mean Squared Error (RMSE) values. As displayed in Table 5, all the RMSE values were low, indicating that the model was of high accuracy.

3.3 Problem Space Exploration

Monotonicity Analysis

Monotonicity Analysis was undertaken in order to determine if the problem was well-bounded and to identify active and inactive constraints. Table 5 shows that the problem has been well-bounded as each design variable has both an upper and lower bound. However, due to the complexity of the objective function, it was difficult to decipher which constraints were critical. This meant that the model could not be simplified.

3.4 Optimisation

The optimal mass of each material was found in MATLAB using Fmincon SQP and Genetic Algorithm (GA) methods. Fmincon was chosen due to its versatility in supporting both linear and nonlinear constraints. GA was selected for its robustness and capacity to produce high quality solutions for optimisation problems. The similarity between results demonstrated the model's high reproducibility.

fmincon SQP Algorithm

Fmincon proved simple to implement, however it is a local minimiser and therefore it was difficult to find a good starting point. This meant that it took a long time to produce an optimal solution.

Model Summary

$$\min_{\vec{x}} f(\vec{x}, \vec{p}) = 4\rho t_a (w_a l_a + 0.5\pi w_a^2) + \rho V_b$$

$$\text{where } \vec{x} = (w_a, l_a, t_a)^T$$

$$\vec{p} = (l_b, l_p, t_b, r_a)$$

$$\text{s.t. } h_1 : l_b = 117.79$$

$$h_2 : l_p = 180$$

$$h_3 : r_a = 22.5$$

$$h_4 : t_b = 5$$

$$h_5 : \rho_1 = 2700$$

$$h_6 : \rho_2 = 1020$$

$$h_7 : \rho_3 = 1150$$

$$h_8 : \rho_4 = 1780$$

$$g_1 : -2l_a - l_b + 2l_b \leq 0$$

$$g_2 : 22 - w_a \leq 0$$

$$g_3 : w_a - 45 \leq 0$$

$$g_4 : l_{def} - 0.01t_a - 0.01t_b \leq 0$$

$$g_5 : t_a - 15 \leq 0$$

$$g_6 : 2.7 - t_a \leq 0$$

$$g_7 : l_a - 300 \leq 0$$

$$g_8 : 180 - l_a \leq 0$$

3.2 Model Development

Approach

The frame's structure should not fail, even under the 200g payload and weight of the other components (assumed to be 300g). Structural failure can happen when the Maximum von Mises Stress exceeds the material's yield strength. Additionally, the frame also should not experience major deflection. Moreover, FEA Static simulations were run to obtain data for maximum stress and deflection along with the mass of the arm.

Latin hypercube sampling was used to obtain the design variables' (arm width, thickness, length and density) values for the FEA simulations. This was conducted in MATLAB. In total, 40 simulations were run, with 10 sets of inputs for each material (of which there are four).

Material	fmincon	GA
Aluminium 6061 Alloy	0.0995	0.0826
ABS PC	0.0831	0.0696
Nylon 101	0.0844	0.0844
Carbon Fibre (Hexcel AS4C (3000 filaments))	-1.0368	-1.0346

Table 6: Optimised mass for each material (kg)

Variable	w_a	t_a	l_a	m_a
Optimal Value	22	2.7	180	0.0696

Table 7: Optimised variables for final mass

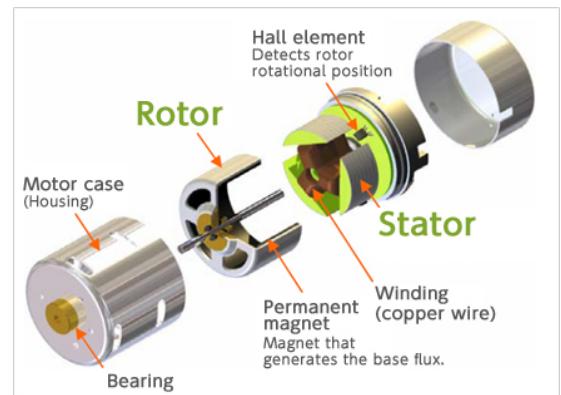


Figure 5: Diagram of BLDC motor parts

Genetic Algorithm

GA was much better at escaping local minima. Although it had a slightly longer runtime than fmincon, it was not sufficient to deem fmincon a better algorithm. Therefore, GA was selected for final results.

Due to the consistently low mass values produced with ABS, it was decided that this material would be taken forward for validation. The results were validated with further SolidWorks FEA simulations to verify that the structure did not fail or exceed the deflection constraint. This value for the mass of the arm, produces a total mass of 318 g for the drone frame.

3.5 Discussion

The final solution was successful in avoiding structural failure, with a maximum von Mises stress of 18.4 MPa, just over half its yield strength (30 MPa). Although the optimised variables lay within their bounds, the algorithm often output their lower bound. This was mitigated against by adding extra functional constraints, however they were rendered inactive within these bounds as they did not affect the algorithm's output.

Although the bounds of each variable were set via industry standards, in future, the range of the bounds should be increased as producing an optimal solution is of higher priority. In fact, had the CAD model of the frame been simpler, deriving functional constraints from its geometry would have been much easier

4 Subsystem 2: Motor

Motor selection is a key part of any drone design - matching the power required to drive the propeller while keeping the motor itself as light as possible is critical. Typically, propellers require low torque and high RPM for peak performance [8]. Aside from the battery, the four motors required in a quad-copter are a significant contributor to the overall weight and as such the overarching efficiency. This section seeks to find an optimal motor design which satisfies the requirements to generate sufficient thrust for the drone while minimising the weight. Figure 5 shows key motor terminology

4.1 Optimisation Formulation

Optimisation Formula

The equations opposite denote the negative null form of the optimisation problem described above. The optimisation is based on a set of design variables \vec{x} and parameters \vec{p} to minimise the motor mass, \vec{m}_{motor} .

Design Variables

The design variables cover the main defining factors in BLDC (Brushless DC) motor design. Firstly, with regards to stator design, n_t is the number of teeth per phase while w_t is the width of these teeth. h_s and r_s define the height and radius of the stator respectively which are the primary geometric variables in most motor designs. The number of coils of wire on the stator teeth is given by n_w and the wire radius is given by r_w . These are integral to the strength of the electromagnet and therefore the torque produced. Finally, the size of the air gap between the stator and rotor is given by t_{ag} and the thickness of the permanent magnets is t_m .

Design Parameters

For this optimisation problem the design parameters were assumed as follows. The maximum current draw from the battery was set to 5A at 12V. The cruising rpm and torque were set to 0.025 Nm and 11000 rpm respectively - requiring an output power of approximately 30W per propeller. To account for acceleration it was assumed that torques as high as 5 times the cruising torque would be required. Finally N42 neodymium magnets were assumed which have a magnet remanence of approximately 1T.

Constraints

Constraints g_1 and g_2 are geometric constraints to ensure feasible tooth widths based on the number of teeth and the stator radius as well as constraining the number of windings on the stator teeth. g_3 limits the current required to drive the propeller to less than what either the battery can supply or the limiting current of the windings wire. Likewise, g_4 ensures the motor can supply sufficient torque for all operating conditions. g_5 and g_6 are the upper and lower bounds of the design space respectively; limiting the design space to values that are physically possible and sensible.

4.2 Modelling Approach

Optimisation Formula

$$\min_{\vec{x}} f(\vec{x}, \vec{p}) = \|\vec{m}_{motor}\|_1$$

where $\vec{x} = (n_w, n_t, w_t, h_s, r_s, r_w, t_{ag}, t_m)^T$

$$\vec{p} = (I_{max}, V, \tau_{cruise}, \omega_{cruise}, m_\tau, \beta_0)^T$$

$$\vec{m}_{motor} = (m_t, m_w, m_{mag}, m_r, m_c)^T$$

$$\text{s.t. } g_1 : 0.4\pi r_s - 3n_t w_t \leq 0$$

$$g_2 : n_w - \frac{1}{2r_w} \sum_{i=1}^3 r_s$$

$$- \max\left(\frac{(0.5w_t + r_w + ir_w\sqrt{3})}{\tan \theta}, r_0\right) \leq 0$$

$$g_3 : \frac{V - \sqrt{V^2 - 4R_w \tau_{cruise} \omega_{cruise}}}{2R_w} - I_{op} \leq 0$$

$$g_4 : m_\tau \tau_{cruise} - 4\beta_s I_{op} (2n_w n_t (2w_t + 2h_s)) r_s \leq 0$$

$$g_5 : (5, 3, 2 \cdot 10^{-3}, 10 \cdot 10^{-3}, 10 \cdot 10^{-3}, 0.08 \cdot 10^{-3}, \\ 0.4 \cdot 10^{-3}, 2 \cdot 10^{-3})^T - \vec{x} \leq 0$$

$$g_6 : \vec{x} - (100, 15, 10 \cdot 10^{-3}, 50 \cdot 10^{-3},$$

$$50 \cdot 10^{-3}, 5 \cdot 10^{-3}, 2 \cdot 10^{-3}, 20 \cdot 10^{-3})^T \leq 0$$

$$h_1 : m_t - (w_t h_s (r_s 0.8 - 0.0025) + 0.0025 h_s$$

$$\cdot \left(\frac{2\pi r_s}{3n_w} - \frac{2\pi r_s}{200} \right)) \rho_{LI} = 0$$

$$h_2 : m_w - \text{cyl}(r_w, n_t (2w_t + 2h_s), \rho_c) = 0$$

$$h_3 : m_{mag} - \text{hcy}(r_s + t_{ag}, r_s + t_{ag} + t_m, h_s, \rho_m) = 0$$

$$h_4 : m_r - \text{h cyl}(r_s + t_{ag} + t_m, r_s + t_{ag} + t_m 2, \\ h_s, \rho_{ms}) = 0$$

$$h_5 : m_c - 2\text{cyl}(r_s + t_{ag} + t_m 2, 1 \cdot 10^{-3}, \rho_c) = 0$$

$$h_6 : R_w - \frac{(2n_t n_w (2w_t + 2h_s)) 1.68_{10^{-6}}}{\pi r_w^2} = 0$$

$$h_7 : \beta_s - \frac{\beta_0 t_m}{t_{ag} + t_m} = 0$$

$$h_8 : I_{op} - \min(8.73 \cdot 10^6 r_w^2 + 281 r_w - 0.0428, \\ I_{max}) = 0$$

$$h_9 : \theta - \frac{pi}{3n_t} = 0$$

First Principles

A number of basic equations were used to model the system constraints. The primary physical equations used are described in Equations (1 - 3).

$$F = I \vec{L} \times \beta \quad (1)$$

$$\tau = r_s \times (I \vec{L} \times \beta)$$

$$I_{in} V = \tau \omega + P_{losses} \quad (2)$$

$$= \tau \omega + I^2_{in} R_w$$

$$\beta_s = \frac{\beta_0 t_m}{t_m + t_{ag}} \quad (3)$$

These equations are applied to the model, taking into consideration the specific geometry of a BLDC motor, to develop valid constraints.

Assumptions

A number of assumptions have been made in order to arrive at the model developed. Primarily, it was assumed that losses are primarily copper losses (Of the form $I^2 R$) as opposed to hysteresis losses or cogging effects. The mathematical formulation of the mentioned losses was too complex for the scope of this project. The effects of cogging were mitigated by requiring at least 3 teeth per phase. The book Electric Machinery 3rd edition [8] suggests this is an acceptable assumption but further review of literature for small BLDC motors is needed. Optimal magnetic conditions were also assumed, in so far as magnetic saturation of the core was ignored and perfect linkage between the wires and the core and the core and the magnets was assumed. This is reasonable as only 3 layers of windings per tooth were allowed. Edge effects of the β field were also discounted.

To avoid having to use the limited number of optimisation methods available for non-linear constrained mixed integer programming problem due to n_t and n_w , I have assumed these to be continuous variables. Should an optimisation algorithm return non-integer values then the design space immediately surrounding will be explored and the best option chosen.

Monotonicity Analysis

The thickness of the air gap, t_{ag} , and the magnet thickness, t_m are both monotonically increasing in f across the design space and so by MP1 g_5 must be active bound the function from below. As there are no other constraints on these variables we can simplify the problem using equality constraints equal to their respective lower bound values. This reduces the design space to only 6 dimensions. MP2 is satisfied as all non-objective variables are constrain above and below by equality constraints (h_1 through h_8).

4.3 Optimisation

To find an optimal design, two different MATLAB optimisation algorithms were used, and the results compared. As the objective function is smooth and differentiable, gradient based methods were employed to find an optimum.

GA-SQP pipeline

First, a GA - SQP pipeline (Genetic Algorithm and Sequential Quadratic Programming) was used. GA is used to explore the design space efficiently using stochastic processes, which allow it to avoid local minima more effectively. The resultant design is then a strong initial point for the SQP algorithm to work from, exploiting the design space topology to find a local optima. It is expected that this is a good approximation of the global optima. The design defined by this is achieved a motor mass of 46.51g while successfully meeting all constraints. The design achieved by only using GA varies but, for one run, was 49.14g - demonstrating that GA is not guaranteed to find an optima.

Global Search

Secondly, the MATLAB GlobalSearch function was used. This repeatedly runs solvers across the design space to find the global optima - for this particular optimisation it requires between 12 and 48 solvers in order to find the global minima. While this method is much more likely to find the global optima it comes at a cost of being much slower to run. This was used to verify the result gained from the GA-SQP pipeline. The design achieved was a motor of mass 46.51g - exactly the same as the previous design.

Algorithm	GA	SQP	Global Search
n_w	5.0000	5.4280	5.4280
n_t	3.0000	3.1059	3.1059
w_t / mm	2.8661	2.0000	2.0000
h_s / mm	10.0029	10.0000	10.0000
r_s / mm	15.5412	14.8293	14.8293
r_w / mm	0.8611	0.8815	0.8815
t_{ag} / mm	0.4000	0.4000	0.4000
t_m / mm	2.0000	2.0000	2.0000
Motor Mass / g	49.14	46.51	46.51

Table 8: Optimum designs produced by GA, SQP and Global Search

4.4 Discussion

Overall, the optimisation of the motor would require further detail to provide a sufficiently robust design however serves as a good initial point for motor prototyping. The weight estimated likely higher than could be achieved as it doesn't consider optimising the part geometry beyond affecting the height and radius. However the performance of the motor regarding the torque produced and the power output are likely optimistic due to the model not considering hysteresis or cogging losses as well as ignoring magnetic saturation. The optimum design produced by each of the algorithms used is shown in Table 8.

5 Subsystem 3: Propulsion System

To optimise the efficiency of the propulsion system, the resistance torque will be minimized - whilst still providing the required thrust to allow the drone to fly. The propulsion system generates a certain thrust, depending on the geometry of the propeller and the RPM of the motor.

Propellers efficiency is measured by increasing the RPM and measuring the resistance torque to calculate mechanical power [9]. If mechanical power was to be optimised multicollinearity would occur. This is due to power being the product of a dependant (τ_M) and independent (R_M) variable, hence torque was chosen as the minimiser. Additionally, the data-set discussed in 5.2 (which contained power), had a VIF value of 21.2 requiring removal of the power term [10][11].

5.1 Optimisation Formulation

The required thrust for a single propeller (T_r) for the negative null form below was initially set at 5N, calculated from

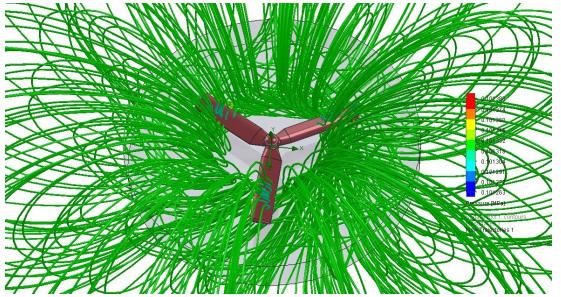


Figure 6: Initial SOLIDWORKS CFD Simulation

a general formula were ($T_r = 0.5gm_D$) [12]:

$$\begin{aligned} \min_{\vec{x}} \quad & f(\vec{x}) = \tau_M \\ \text{where } \vec{x} = & (L_P, P_P, N_B, R_M)^T \\ \tau_M = & \beta_{poly-\tau} \cdot \vec{x}_{poly} \\ T_R = & \beta_{poly-T} \cdot \vec{x}_{poly} \\ \text{s.t. } h_1(T_r) : & 5 - T_r = 0 \\ g_1(L_P) : & -300 + L_p \leq 0 \\ g_2(P_P) : & -400 + P_p \leq 0 \\ g_3(N_B) : & -4 + N_p \leq 0 \\ g_4(L_B) : & 2 + N_p \leq 0 \\ g_5(R_M, L_P) : & -R_M + \frac{4 \cdot 10^6}{L_P} \leq 0 \\ g_6(T_P, T_r) : & T_P - T_r \leq 0 \\ g_7(L_P, P_P) : & -3 + \frac{L_P}{P_P} \leq 0 \\ g_8(L_P, P_P) : & -1.5 + \frac{P_P}{L_P} \leq 0 \end{aligned}$$

Design Parameters and Variables

- 01) The geometric design variables (L_P, P_P, N_b) are commonly used to specify the geometry of different propellers .
- 02) The APC Thin Electric propeller was used for all studies giving a consistent aerofoil and therefore comparative data.
- 03) Propeller with a high L_P are prone to becoming unstable and unsafe at high speeds therefore constraints G_5 was implemented based of the manufacturer of the test propellers guidance [12].
- 04) A high ratio may perform theoretically well but is not physically possible due to P_P being restricted by L_P . The highest and lowest ratios were found from the data-set and applied in the form of constraint G_8 and G_9 .
- 05) L_P was limited to below 300mm, the upper bound of a commercial drone, forming constraint G_1 [14].

5.2 Model Development

Initially a SOLIDWORKS CFD model 6 was used to create the data-set for meta-modelling, however the complex geometry and small scale of the propellers meant that generated thrust values were much smaller when compared to experimental data from a propeller with the same dimension as the test.

Instead, experimental test data from a data-set of propellers with a varying L_P , P_P and N_b , tested at increasing R_M was used [15]. Only the tests marked E, where the electric variant was tested, could be used. The 2700 instances were copied into excel. The meta model was a better rep-

resentation of the real world problem as the data was not artificially generated.

Data Preparation

The dataset was analysed initially in excel and outliers were noticed. Model preparation went as follows.

Sampling: The highest and lowest 10% of τ_M results were removed to get rid of outliers.

Split: Regression methods used a 75:25 test train split, ANN used a 70:15:15 , test/train/validation.

Normalisation/Scaling: Data was normalised for the regression models using the and scaled between 0-1 for the ANN.

Scikit-learn in Python was used do to familiarity and code was written to directly print the objective functions for the regression methods, allowing for quick iterations between meta-modelling and optimisation .

Model Development

A comparison meta-modelling methods accuracy, measured by coefficient of determination calculated on the test data, can be seen below: An ($R^2 = 0.80$) is considered to mean

Meta-Modelling Method	Test Acc τ_M	Test Acc T_R
Linear Regression	0.480	0.510
Polynomial Regression	0.800	0.875
Keras ANN	0.721	0.941

Table 9: Comparison of Meta-Modelling Methods Accuracy(R^2 on test)

a large positive correlation [14]. Hence multi-varient polynomial regression is on the threshold for modelling the objective function, therefore it was chosen as the method to be improved. The Lasso regularization (L1 penalisation) method was used to reduce the number of features of the polynomial function, by equating irrelevant feature coefficients (values) to 0, depending on an value. Different degrees of polynomials were also tested to achieve the optimum accuracy, without over fitting the model. None of the tested polynomials showed an accuracy loss of more than 5% showing overfitting was not present [16]. Higher order polynomials provided better accuracy - possibly due to interdependency being better represented by the number of increased interaction terms. A third degree polynomial with a Lasso ($\alpha = 0.7$) was chosen as it provided an accurate meta-model without using too many features, that could become computationally expensive during optimisation.

5.3 Problem Space Exploration

Monotonicity Analysis

Monotonicity analysis was performed and it was found the problem was well bounded. Analysing the gradient of the objective function was difficult due to it's complexity and

Poly Degree	Lasso α	Test Acc τ_M	Test Acc T_P	Terms τ_M	Terms T_P
2	0.1	0.800	0.875	14	14
3	0.2	0.889	0.957	34	32
3	0.7	0.888	0.976	32	29
4	0.8	0.908	0.993	62	61

Table 10: Comparison of polynomial's accuracy

volatility. Some constraints were only partially active depending on the position in the design space, e.g. G_5 only became active at high values of L_P and low values for P_P .

2 Improving the Representation of T_R

Another observation, was that increasing L_P increased T_P however not T_R . Initially T_R was set at a constant estimated from a fixed m_D , when in reality increasing L_P would increase the m_D , this resulted in inaccurate optimisation results as the upper bound for L_P was always reached . An additional data-set was created, collating the L_P from the original data-set and their relative masses, from this a polynomial relationship ($R^2 = 0.99$) was found. Increasing L_P also meant a larger frame was required. The increase was derived from subsystem 1's data-set and a linear relationship between L_P and m_F ($R^2 = 0.78$) was found. The equation was implemented in place of the T_R in G_6 improving the overall accuracy of the model and providing a more optimum due to mass minimisation - effectively somewhat accounting for the subsystem interdependencies.

$$g_6(T_p, T_R) : T_p - T_R \leq 0$$

$$\text{Where: } T_R = \left(\frac{m_P L + m_D}{500} \right) g \\ m_D = (m_P + m_M)N_P + m_B + m_F \\ m_P = 0.5(0.0007L_P^2 - 0.19L_P + 18.661) \\ + 18.661)N_B \\ m_F = (200 + 4(\frac{L_P}{3.5}))$$
(4)

5.4 Optimisation

Four different non-linear numerical methods of optimisation, were implemented in MATLAB.

GA-SQP

Initially the GA was implemented to stochastically search the design space, proving a close estimate of the global optima. This result could then be used as the initial starting point for a SQP to find the local optima (using a gradient based method). As the objective functions nature was difficult to interoperate the results of the GA were often sporadic sometimes failing to find a suitable starting point. The N_B for SQP had to be rounded to the closest integer and G_6 manually tested to validate the result.

Multistart & Black-Box (Surrogateopt)

An alternative global optimisation method multi-start (which finds multiple local minima's) was used. The objective function was not used for multi-start, instead only the raw data was processed. A successful result was found, however the validation process for G_6 was required again. The Surrogate method provided the most successful result as a starting point could be specified, as well as strict bounds. Therefore the objective function could be optimised in a smooth, more stable area, rather than the volatile area (near the edges of the data-set). An integer constraint for N_B could also be implemented. A comparison of methods can be seen below:

*Below results are non-normalised. GA&SQP require normalised objective and nonlinear equation to compensate for volatility of non-normalised equation.

Optimisation Method	L_P	P_P	N_B	R_M	τ_M	T_P
GA	300	219	2	8000	140	21.5
SQP	100	150	2	26617	193	6.1
MultiStart	220	151	4	27000	181	18.7
Black-Box	125	181	3	23575	102	12.3

Table 11: Comparison of Optimisation Methods

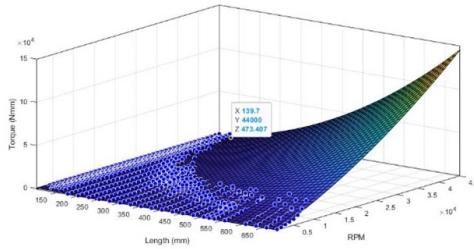


Figure 7: Polynomial fit showing correlation of (L_p, R_m, t_m)

5.5 Discussion

Influencing Factors

Resistance torque was influenced most by L_P . Mechanically a larger L_P means greater inertia, therefore more force is required to turn the propeller 6. The output of the optimised propeller is close to that of the propeller with the highest ratio of T_p to τ_M (x(157,67,2,2800)) which is positive sign. Furthermore, when visually analysing the relationship between the input variables and M it is clear that minimising L_P and R_m , will optimise τ_M 7:

Limitation of Meta Modelling

The strength of the optimisation relies on the accuracy of the model. Despite experimental data providing the most accurate representation of a drone propulsion system, the validity of the data is down to the provider, hence it can't be confirmed that values are 100% accurate. Meta Modelling provided the most significant challenge, due to the complex relationship and interdependency.

Optimisation Result

It is not surprising that the surrogate model provided the best results as it is recommended for problems were the nature of the objective function is difficult to understand and many local minima's may occur.

6 Subsystem 4: Drone Trajectory

To optimise the power consumption of a drone trip between the Dyson building and the Hackspace, the goal is to minimise travel time by determining the optimal coordinates of three waypoints. Additionally, extending the battery life is of interest (especially SOC). Logically, optimising travel time will inherently reduce energy depletion of the battery. Generating a wind profile to study, will allow the creation of an environmental and battery-aware optimisation model.

6.1 Optimisation Formulation

A generalised form of the objective function is:

$$f(\mathbf{x}, \mathbf{y}, \mathbf{u}, \mathbf{v}) = \frac{\int_{t_i}^{t_f} \sqrt{(\dot{x}(t))^2 + (\dot{y}(t))^2} dt}{V_d + \int_C F \bullet dr} \quad (5)$$

After discretisation of the problem into three waypoints, the negative null form is:

$$\begin{aligned} \min_{\vec{x}} \quad & f(\vec{x}) = \sum_{i=1}^3 \left(\frac{\int_{t_{i-1}}^{t_i} \sqrt{(\dot{x}_i)^2 + (\dot{y}_i)^2} dt}{V_d + \int_{C_i} F \bullet dr_i} \right) \\ \text{where } & r(t) = (x(t), y(t)) \\ & \dot{r}(t) = (\dot{x}(t), \dot{y}(t)) \\ & x(t) = x_0 + \int_{t_i}^{t_f} F(x(u)) du \\ & y(t) = y_0 + \int_{t_i}^{t_f} F(y(v)) dv \\ & \mathbf{F}(x, y) = \langle u(x, y) \ v(x, y) \rangle \\ & \int_C F \bullet dr = \int_{t_i}^{t_f} F(x(t), y(t)) \bullet \dot{r}(t) dt \\ & \int_C F \bullet dr = \langle \mathbf{F}(x, y) | r(t) \rangle \\ \text{s.t. } & h_1 : N_w - 3 = 0 \\ & h_2 : \dot{r}(t) - F(r(t)) = 0 \\ & h_3 : V_d - 15 = 0 \\ & g_1 : x_1 - 4 \leq 0 \\ & g_2 : x_2 - 4 \leq 0 \\ & g_3 : x_3 - 4 \leq 0 \\ & g_4 : y_1 - 2 \leq 0 \\ & g_5 : y_2 - 2 \leq 0 \\ & g_6 : y_3 - 2 \leq 0 \\ & g_7 : x_1 - x_2 \leq 0 \\ & g_8 : x_2 - x_3 \leq 0 \\ & g_9 : y_1 - y_2 \leq 0 \\ & g_{10} : y_2 - y_3 \leq 0 \\ & g_{11} : v - 20 \leq 0 \\ & g_{12} : u - 20 \leq 0 \\ & g_{13} : -v - 20 \leq 0 \\ & g_{14} : -u - 20 \leq 0 \\ & g_{15} : \sqrt{v^2 + u^2} - 30 \leq 0 \end{aligned}$$

The objective function is a minimisation of time and is related to the drone's coordinates as well as its speed in the generated wind vector space. It is subject to multiple constraints which are divided into the following:

- Number of waypoints: h_1
- Dimensionality: $g_1 - g_15$ and h_3
- Velocity: $g_1 - g_15$ and h_3
- Vector field and flow line: h_2

6.2 Model Development

First Principles

Taking into account wind, the motion of the drone as an initial negative null form equation system is:

$$\begin{aligned} \min_{\theta} \quad & f(\vec{x}) = (t_f - t_i) \\ \text{s.t.} \quad & \dot{x}(t) = d + \sin(\theta(t)) + W_x \\ & \dot{y}(t) = d + \cos(\theta(t)) + W_y \\ & (x(t_i), (t_i)) = (x_i, y_i) \\ & (x(t_f), (t_f)) = (x_f, y_f) \end{aligned}$$

The control parameter of the optimal path is the heading angle θ . By considering $s(X(t), a(t))$ as the true speed of the drone taking into account the wind:

$$\begin{aligned} \min_{a \in A} \quad & f(\vec{x}) = (t_f - t_i) \\ \text{s.t.} \quad & a(t) = (\sin(\theta(t)), \cos(\theta(t))) \\ & \dot{X} = s(X(t), a(t)) \\ & X(t_0) = X_0 \\ & X(t_f) = X_f \end{aligned}$$

This is a standard formulation of optimal control problems solvable as a Hamilton-Jacobi equation which is based on Bellman's optimality principle in discrete case:

$$\min a \in A \{ \nabla u(X).s(X, a) \} = 1 \quad (6)$$

The resolution of this PDE gives a necessary and sufficient condition for optimality and can be done by decomposing it recursively into linked sub-problems as in DP. The resolution can be seen as a front expansion problem where the wavefront represents the minimum time to reach the arrival point. This resembles a problem of recursive stochastic gradient descent. It is based on Huygens's principle as:

$$\|\nabla u(\mathbf{X})\| = \min a \in A \left\{ \frac{\nabla u(X)}{\nabla u(\mathbf{X})} s(X, (\mathbf{a})) \right\} = 1 \quad (7)$$

Simplification of the Problem

Computing and solving such an equation is beyond the scope of this report. To simplify the problem computationally, it is discretised by implementing three waypoints which become the optimisable variables: $x_1, x_2, x_3, y_1, y_2, y_3$. The equation to optimise becomes a simple kinematics relationship

$$\Delta t = \frac{dP}{dv_{total}} \quad (8)$$

The position (distance between two waypoints) is given by the line integral of the curve in the wind vector field. The total velocity equates to the drone's velocity added to the tailwind/headwind components of the wind velocity. Instead of determining an equation for the vector field representing the wind profile, it is randomly generated using MATLAB. Regardless, to go through monotonicity analysis, a vector field and flow line are set:

$$\mathbf{F}(x, y) = \langle u(x, y) | v(x, y) \rangle \quad (9)$$

$$r(t) = (x(t), y(t)) \quad (10)$$

Function	x_1	x_2	x_3	y_1	y_2	y_3	u	v
f	+	+	+	+	+	+	?	?
g_1	+							
g_2		+						
g_3			+					
g_4				+				
g_5					+			
g_6						+		
g_7	+	-						
g_8		+	-					
g_9				+	-			
g_{10}					+	-		
g_{11}							+	
g_{12}							+	
g_{13}							-	
g_{14}							-	
g_{15}							+	+

Table 12: Monotonicity Analysis

Algorithm	SQP	active set	Interior point	GA
Optimal time	13.54	13.62	13.54	13.54
x_1	20	20	20	20
y_1	7.49	6.93	7.49	7.51
x_2	29.45	23.31	29.45	29.7
y_2	12.63	9.19	12.63	12.78
x_3	33.07	21.91	33.06	33.16
y_3	13.03	12.5	12.03	13.07

Table 13: Algorithm Comparison

6.3 Problem Space Exploration

Monotonicity analysis was performed on the inequality constraints of the model. However, due to complexity, some constraints might be overlapping on certain portions of the working space. To that extent, it was decided to keep them although they might be partially active. In Table 12, the rows g_2-g_3 and g_5-g_6 are inactive while the other rows are active or potentially active and related to dimensionality constraints (g_1, g_4 and g_7-g_{10}) and velocity constraints ($g_{11}-g_{15}$).

6.4 Optimisation

MATLAB was used to optimise the objective function with given constraints. Firstly, the curve is interpolated by using the discretised waypoints to generate a continuous path. Secondly, this function is used to interpolate the wind vector field at all the points in the path. Thirdly, the direction vector is calculated from the difference between each point of the extrapolated path function. These small differences are added together to find the curve's arc length. Then, the tailwind and headwind contributions at each given point are obtained by doing the dot product of the wind vector field with the direction vector (5)(6). Finally, the time between each point is calculated (8) and the sum of those are optimised to find the waypoints' coordinates. Different algorithms are used to do so (Table 13), and the final results display fmincon (sqp) as the best solver.

To demonstrate the optimality of the result, the time taken a dataset of 25 drone trajectories is generated and analysed and results are presented in table Table 14.

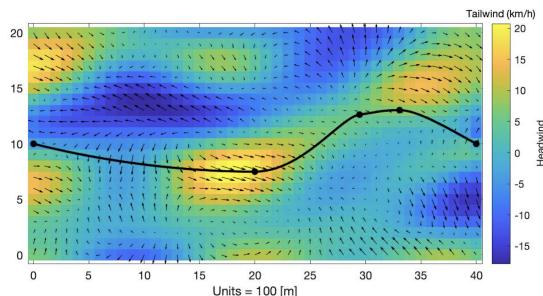


Figure 8: Optimal trajectory in a generated wind field

Metric	Time (min)	Δ to optima
Average	22.5	39.9%
Median	18.5	27.0%
Max	66.7	79.8%
Min	13.6	0.7%
Straight line	16.9	20.1%

Table 14: Time to path comparison

6.5 Discussion

Difference Between Optimiser Algorithms

Initially, fmincon with an SQP algorithm was used to solve the optimisation problem. Then, two other fmincon solvers were used (active-set and interior-point) demonstrating similar results. Moreover, using a genetic algorithm showcased similar results to fmincon with a higher solving time. This is because fmincon acts as a local optimiser compared to genetic algorithms which are global optimisers. However, results being that similar would hereby present fmincon as a better solver. If z was defined as an optimisable variable, then the increase in complexity of the problem may mean that a ga algorithm becomes better at finding the optimal solution.

A Battery Aware Model

Concerning the relationship with the battery, five cells which are often used in drones (Ultimate PX-04 LIPO) are connected in series. Doing so increases the total capacity to 5000 mAh. The specs display the discharge curves at different C-rates and a Python program is used to discretise the curves into points and interpolate a fifth-degree polynomial equation to fit them (11):

$$V(C) = a_1C^5 + a_2C^4 + a_3C^3 + a_4C^2 + a_5C + a_6 \quad (11)$$

Calculating the area under the curve with an integral between initial capacity at starting voltage and final capacity at cut-off voltage allows to determine the battery's energy depletion as a function of capacity (12):

$$E(C) = \int V(C) dC \quad (12)$$

By considering that the battery outputs 12V the weights of the function $V(C)$ can be computed through interpolation giving us an array for the values a_1-a_6 of: [-7.2e-14; 1.6e-10; -1.3e-07; 5.0e-05; -1.0e-02; 12]. By accounting for the optimal flight power consumption of 170W as well as the total energy available, total energy is:

$$E = P\Delta t \mid E_{total} = E(5[Ah]) = 59.88Wh \quad (13)$$

Path	Time (min)	Energy Use (Wh)	SOC
Paths	13.5	38.25	63.88 %
Straight	16.9	47.9	79.99 %
Average	22.5	63.8	106.55%
Median	18.5	52.4	87.51%
Total	21.1	59.88	100.00%

Table 15: SOC path comparison

sult in about 64% depletion of the battery's energy while demonstrating that the average path would not even manage a one-way trip between the Dyson building and the Hackspace.

7 System Level Discussion

Overall, the optimal solution uses ABS for its frame and weighs 318 g. The drone's motors each weigh 46.5g and the propellers have been optimised to reduce the cruising torque to 102 Nmm. Finally, an optimal path was created for the drone's journey between the Dyson Building and the White City Hackspace, depleting only 64% of the battery's energy, significantly lower than for a straight or median path. When comparing these results to the Dji phantom 4 pro v2 drone model, it is clear that an optimal system has been created. The Dji phantom 4 pro v2 drone weighs 1375g and requires a torque of 134 Nmm. This demonstrates this new design's capability to compete against market competitors.

In regard to how each subsystem has informed system design, each analysis was conducted concurrently, creating mutually beneficial relationships between them. Due to the scope of this report, simplifications of physical systems was required to aid modelling. These simplifications inherently reduced the number of variables, which decreased overall system accuracy. For example, in the propulsion subsystem, accounting for all the variables of the aerofoil profile could have had a large impact on that subsystem's optimal result. With more time, each subsystem's model could have been more complex and accounted for more variables.

8 Conclusion

Through the analysis of four subsystems, power consumption of the drone in the studied model was decreased substantially. Indeed, the modelled drone would manage to fly efficiently between the Dyson building and the Hackspace for the delivery of lightweight parts. Nevertheless, we cannot know for sure the actual result of the optimisation on a system-level as interdependencies between the subsystems are not necessarily linear and cannot be summed together. Hence, this system optimisation is only as good as the models we have created.

References

- [1] Seifert et al (2019), Influence of Drone Altitude, Image Overlap, and Optical Sensor Resolution on Multi-View Reconstruction of Forest Images, MDPI
- [2] Eicker et al (2018) An Optimization Problem for Quadcopter Reference Flight Trajectory Generation, Wiley, Hindawi
- [3] Elsayed, M.S, and Mohamed, M. (2019) A Multi-Objective Optimization of Autonomous Drones Solar Energy Charging Stations Utilizing BIPV Urban Upgrade, Proceedings of the 54th Canadian Transportation Research.
- [4] Chen et al (2018) A Case for a Battery-Aware Model of Drone Energy Consumption, Department of Control and Computer Engineering, Polytechnic of Turin
- [5] Chen et al (2017) A circuit-equivalent battery model accounting for the dependency on load frequency, Department of Control and Computer Engineering, Polytechnic of Turin
- [6] Girarget and Lapasset et al (N/a) Wind-Optimal path planning: application to aircraft trajectories, Capgemini Technology Services
- [7] Mary Fenelon (2010) Finding an Optimal Path
- [8] ANNETT FA. Chapter 6: Synchronous Machines, Steady State. In: Electrical Machinery Third editionM. 3rd ed. New York: McGraw-Hill Book Co.; 1950. p. 283–329. (McGRAW HILL Electrical and Electronic Engineering series).
- [9] C. Blouin (2020) How to measure brushless motor and propeller efficiency, RCbenchmark, <https://www.rcbenchmark.com/blogs/articles/how-to-measure-brushless-motor-and-propeller-efficiency>
- [10] [Internet]. Ijamejournals.com. 2020 [cited 15 December 2020]. Available from: <http://www.ijamejournals.com/pdf/rpm11174.pdf>
- [11] B. Szyk (N/a) Drone Motor Calculator, Omni Calculator, <https://www.omnicalculator.com/other/drone-motor>: :text=In
- [12] Author: N/a (2013) Handling Multicollinearity in Regression Analysis, Minitab <https://blog.minitab.com/blog/understanding-statistics/handling-multicollinearity-in-regression-analysis>
- [13] APC Propellers (2016) APC Propeller Speed Specification, Landing Products Inc. <https://www.apcprop.com/wp-content/uploads/2017/06/APC-Propeller-RPM-Limits.pdf>
- [14] Author: N/a (2018) All about multirotor drone FPV propellers,
- [15] APC Propellers (2019) APC Propeller Nomenclature, Landing Products Inc. <https://49u9ei2todm6vjegs45ceqp1-wpengine.netdna-ssl.com/wp-content/uploads/2019/01/APC-Propeller-Nomenclature.pdf> <https://www.getfpv.com/learn/new-to-fpv/all-about-multirotor-fpv-drone-propellers/>: :text=Based
- [16] Author N/a (N/a) Linear Correlation, <https://condor.depaul.edu/sjost/it223/documents>
- [17] Author: N/a (N/a) Overfitting, Corporate Finance Institute <https://corporatefinanceinstitute.com/resources/knowledge>
- [18] Author: N/a (N/a) APC Propeller Static Thrust VS. RPM Predictions STATIC-2, <https://www.apcprop.com/files/>

9 Nomenclature

All values taken to be in S.I. units unless otherwise specified

Symbol	Meaning
w_a	Width of Arm
t_a	Thickness of Arm
l_a	Length of Arm
r_a	Radius of Arm
m_a	Mass of Arm
l_b	Length of Base Plate
t_b	Thickness of Base Plate
l_{def}	Deflection
V_b	Volume of Base Plate
ρ	Density
P_{losses}	Power expended through losses
BLDC	Brushless DC motor - a type of synchronous electric motor
GA	Genetic Algorithm
SQP	Sequential Quadratic Programming
MP1	Monotonicity Principle 1: In a well-constrained minimisation problem, every increasing variable is bounded below by at least one non-increasing active constraint.
MP2	In a well-constrained minimisation problem every relevant nonobjective variable is bounded both below and above.

Table 16: Subsystem 1 Nomenclature

Symbol	Meaning
m_{motor}	Mass of the motor
n_w	Number of windings per tooth
n_t	Number of teeth per phase
w_t	Tooth width
h_s	Stator height
r_s	Stator radius
r_w	Radius of winding wire
t_{ag}	Thickness of air gap
t_m	Thickness of magnets
I_{max}	Maximum current draw from battery
V	Battery voltage
τ_{cruise}	Torque required drive propeller to move drone at $5ms^{-1}$
ω_{cruise}	Angular velocity required by propeller to move drone at $5ms^{-1}$
m_τ	Maximum torque as a multiple of τ_{cruise}
β_0	Surface remanence of N42 neodymium magnets
m_t	Mass of stator teeth
m_w	Mass of windings
m_{mag}	Mass of permanent magnets
m_r	Mass of rotor
m_c	Mass of can
θ	Angle given to one tooth
R_w	Total winding resistance
I_{op}	Maximum operating current based on battery and wire radius
$cyl(r, h, \rho)$	Utility function that calculates mass of a cylinder of material
$hcyl(r_i, r_o, h, \rho)$	Utility function that calculates mass of a hollow cylinder of material
ρ_{li}	Density of laminated iron
ρ_c	Density of copper
ρ_m	Density of neodymium
ρ_{m^s}	Density of magnet steel
ρ_a	Density of aluminium
β_s	Magnetic field strength at the stator
F	Force generated by current moving through a magnetic field
L	Vector of the length of wire carrying current

Table 17: Subsystem 2 Nomenclature

Symbol	Meaning
L_P	Length of the propeller measured as the maximum diameter
P_P	Pitch of the propeller
N_B	Number of blades of the propeller
R_M	Revolutions per minute of the motor
m_D	Overall mass of the drone (excl. payload)
m_F	Mass of the frame
m_B	Mass of the battery
N_P	Number of Propellers
m_P	Mass of a single propeller
T_r	Thrust required for the drone to fly
T_p	Thrust produced by the propulsion system
x_{poly}	Polynomial feature vector of x
β_{poly}	Coefficients of polynomial model. See code for details
τ_M	Resistance torque subjected to the motor by the propeller

Table 18: Subsystem 3 Nomenclature

Symbol	Meaning
t_i	Time of departure
t_f	Time of arrival
F	Wind vector field
V_d	Drone speed
t_0, t_1, t_2, t_3	Time between each waypoint
x_1, x_2, x_3	X coordinate of each waypoint
y_1, y_2, y_3	Y coordinate of each waypoint
r_1, r_2, r_3	Flow line equation between waypoints
C_1, C_2, C_3	Curve between each waypoint
u	Horizontal wind velocity
v	Vertical wind velocity
W_x	Horizontal wind profile
W_y	Vertical wind velocity
θ	Heading angle
$s(X, \mathbf{n})$	Speed vector of the drone with wind
\mathbf{n}	Outward unit vector normal to the front at X
X_0	Initial position
X_f	Final position
$u(X)$	Minimum time to reach end point starting from X
N_w	Number of waypoints
V	Voltage
C	Capacity
E	Battery energy
E_{total}	Total energy of the battery pack
SOC	State of Charge
PDE	Partial Differential Equation
DP	Dynamic Programming

Table 19: Subsystem 4 Nomenclature