Aircraft Sizing Algorithm Documentation

Team 7:
Cooper LeComp, John A. Papas Dennerline,
Ian Greene, Christos Levy, Daniel Qi, Hans Suganda

27th August 2022

Contents

1 Rationale

We need to determine 2 very important things:

1. The thrust to weight ratio affects:
    - acceleration: A larger thrust to weight ratio will affect this positively. Based on Newton's 2^{nd} law, if we have a larger force due to a larger thrust, the object (aircraft) should accelerate more.
    - climb rate: A larger thrust to weight ratio affects this positively. If we have a larger thrust to weight ratio, then the aircraft can put out more potential energy into its height at a faster rate.
    - maximum speed: Larger thrust to weight ratio gives a larger maximum speed. The more thrust an aircraft can produce, the more drag it can overcome.
    - turn rate: If the aircraft has a larger thrust to weight ratio, it can sustain higher speeds and thus have incredible turn rates.
    - Aircraft Endurance: Thrust to weight ratio will affect this parameter negatively. If an aircraft produces alot of power for its own weight, it will consume its fuel and cannot stay flying for too long.
2. The stall speed is directly determined by the wing loading and the maximum lift coefficient. Wing Loading will affect:
    - stall speed: If the wing is smaller for the aircraft's weight, stall speed has to increase because the aircraft has to maintain the same amount of lift to remain flying.
    - induced drag: If the wing is smaller, then at lower speeds the lift coefficient would be much higher. Since drag is proportional to lift coefficient squared, this would mean a smaller wing would produce a much larger induced drag at lower speeds.
    - take-off distance: A smaller wing is less capable of producing lift at low speeds. Therefore, the aircraft needs more speed, increasing take-off distance.

These 2 parameters, the wing loading and the thrust to weight ratio determines much of the aircraft's flight characteristic which is why it is incredibly important to base the sizing of the aircraft to these 2 parameters. These 2 parameters can be deduced or approximated based on the requirements of the RFP which is the first step in our sizing algorithm.

Alright, supposing we could determine the wing loading and the thrust to weight ratio given the requirements what next? We have to center our sizing based on discrete-sized components that are available in the market. Motor-propeller combinations would be a good example in this case. Suppose we were to use a sizing method that is based on the target payload, and we get a required motor power rating that is x watts in power. What happens if the market will only sell $0.3x$ watt rating or $2.3x$ watt power rating with nothing in between? Then all of the decisions that we have made thus far is very unoptimised because we would be forced to use more underpowered motors or an overpowered motor. We could attempt to use multiple engines instead of one, but that deviates from the original plan.

Therefore, a more intelligent way is to base the sizing on discrete parts we are dependent on outsourced manufacturers and then attempt to control all the other components that we can truly control in a continuous way. For example, fuselages, wings, structure all can be controlled. Yes we can attempt to make an aircraft that is 720 grams or an aircraft that is instead 630 grams. Those variables are in our control in a continuous non-discrete way. So if our algorithm eventually tells us to make a wing of $0.89\,m^2$ we can indeed make a wing close to that value.

That is the core principle of the sizing algorithm implemented within this team. The sizing algorithm relies on discrete parts, which in our case would be primarily the motor-propeller combination.

2 Sizing Methodology

1. From the RFP, compute 2 of the most arguably important parameters: Wing Loading and Thrust to Weight Ratio.
   (a) Firstly use the stall speed to determine wing loading. The stall speed should be independent of engine size.
   (b) After figuring out wing loading, determine the thrust to weight ratio based on the specified climb rate.
2. Choose a motor-propeller combination that exists within the market, and figure out:
   (a) The thrust that the setup can produce.
   (b) The power that the motor will consume at cruise.
3. Based on the Thrust to Weight Ratio and the Thrust that the motor-propeller setup produces, figure out the loaded weight of the entire aircraft.
4. Based on the power consumption of the motor and the required endurance, as well as the specific energy of the batteries, determine the weight of the batteries.
5. Based on the Wing Loading, and the loaded weight of the aircraft, figure out the weight of the wings.
   (a) Based on the wing loading and weight of the loaded aircraft, figure out the area of the aircraft's main wings.
   (b) Based on the area of the aircraft's main wings and the material of choice, estimate the weight of the wings.
6. We already know a few things: Weight of the propulsion system (batteries, motor, propeller), Weight of the Lift Devices (Weight of the Wing), Weight of the Payload, Weight of the Aircraft. Subtract the weight of all the known components from the total weight of the aircraft. This is essentially the weight for the fuselage and tail and excess margin mass. We seek to figure out a reasonable number for this fuselage weight.
7. If the fuselage and excess weight quota is too low, choose a propeller motor combination that gives a larger thrust. If the weight quota is too high, or the wing area is just too large and won't fit in the box, then choose a propeller motor combination that gives a lower thrust value. Re-iterate all the steps by just changing the motor-propeller values using the `Matlab` script.

3 Algorithm Structure

The chosen programming language is `Matlab` due to the team's most familiarity with it. We are implementing a programming style that is somewhat similar to object-oriented programming wherein we have a namespace and we have functions which operate on all of the variables within that name space. Although in our case, our namespace is the `global` namespace.

The reason we do it this way is so that we are not going to spend more time debugging making sure that we pass all the correct named variables to smaller functions. Instead we have one and only one name for a physical variable and all scripts can read off those variables. This allows for rapid development of the algorithm.

The `Matlab` script below defines the global variables that is needed for this entire simulation. This script is called by all of the other scripts to make sure that all of the variables listed here is indeed visible to all the other scripts. This script primarily just serves a house-keeping script to keep the other scripts running smoothly.

```
%%Global variable list
global g; %This is local gravitational acceleration constant (m/s^2)
global L_wi; %This is a wing loading (N/m^2)
global T_W; %This is the thrust to weight ratio (dimensionless)
global rho; %This is the air density (kg/m^3)
global v_stall; %This is the stall speed of our aircraft (m/s)
global c_Lmax; %This is an estimation of how good is our biggest possible lift coefff
global q; %This is the dynamic pressure
global c_D0; %This is the drag coefficient near stall speed
global A_r; %Aspect ratio of the main wing
global e; %efficiency of the wing due to deviating from elliptic lift distribution
global G; %Gradient of ascent = rise of flight/run of flight
global endurance; %This is how long the aircraft should be flying in terms of seconds
global Motor_Power; %This is how much power the motor consumes (J/s)
global Batt_Specific_Energy; %This has units of (J/kg)
global Batt_Mass; %This has units of kg, this is the mass of the battery
global Engine_Thrust; %This is the engine thrust of our aircraft (Newtons)
global Total_Aircraft_Weight; %This is the weight of the total aircraft (N)
global Total_Aircraft_Mass; %This is the mass of the total aircraft (kg)
global Main_Wing_Area; %This is the total wing area of the main wing (m^2)
global payload_mass; %This is the mass of the payload in (kg)
global Main_Wing_Thickness; %This is the thickness of the main wing (meters)
global Ave_Density_Main_Wing; %This is the average density of the Main Wing's material (kg/m^3)
global mass_main_wing; %Approximate Mass of the main wing (kg)
global weight_main_wing; %Approximate Weight of the main wing (N)
global Spare_Aircraft_Weight; %The final weight for everything else (N)
global Spare_Aircraft_Mass; %The final mass for everything else (kg)
global Motor_Prop_Mass; %This unit is in kg and is the combined mass of the motor and propeller
```

The script below handles the global variable initializations. This is an important script. This is where we input all of our data, such as the RFP requirements, the market for the propellers, our estimated parameters such as cruise lift coefficients and stall drag coefficients. This script is what the user should modify if they choose to consider a different design with different motor-propeller combinations or a different wing material and so forth.

```
%%Give Definition of Global variable
Global_Variables;

%%Variables which are known/guessed (independent)

%Variables Needed to Compute the Wing Loading and Thrust to Weight Ratio
g = 9.81; %This is the local gravitational accel (m/s^2)
rho = 1.225; %This is the density of the air
v_stall = 8; %This is the estimation for stall speed
c_Lmax = 1.2; %the maximum lift coefficient we can get
climb_angle = 25; %This is from the RFP
c_D0 = 0.03; %This is the drag coefficient near stall speed
A_r = 9; %Aspect ratio of the main wing, typical values
e = 0.8; %efficiency of the wing due to deviating from elliptic lift distribution
```

```
%Variables that are needed to compute the total weight of the aircraft
Engine_Thrust = 40; %This is force produced by engine, units are in Newtons (N)

%Variables that are needed to compute the mass of the batteries
endurance = 900; %seconds of flight. This corresponds to 15 minutes of flight
Motor_Power = 200; %This is the power of the motor (W), has to be in cruise conditions
Batt_Specific_Energy = 0.655*10^6; %This is Energy/Mass, so its unit has to be (J/kg)

%Variables that are needed to compute the Lift Devices
Main_Wing_Thickness = 0.07; %Units in meters
Ave_Density_Main_Wing = 24.82862; % Density of Wing Material (kg/m^3)

%%Directly Known Masses of the Aircraft

%Mass of the Motor and Propeller of Aircraft
Motor_Prop_Mass = 0.2; %This is in kg and taken from manufacturer

%Mass of the Prescribed Avionics
mass_pixhawk = 0.037;  % Units in kg
mass_servos = 0.03 * 5; % Units in kg

%Mass of the Payload
payload_mass = 1.5*0.45;

%%Variables that are dependent on other variables (dependent)
q = 1/2*rho*v_stall^2; %This is the dynamic pressure
G = tan(deg2rad(climb_angle)); %Gradient of ascent = rise of flight/run of flight
mass_avionics = mass_pixhawk + mass_servos; % Units in kg
```

Of course, with every single Matlab project, there has to be a Main function which calls on all of the other scripts. This is our Main function. Its sole task is to set-up the variables correctly, call all of the other scripts which process the data and call the scripts that give us the useful informations we care about.

```
%%Give Definition of Global variable
Global_Variables;

%%Basic Housekeeping Before Running Simulation
clear; clc;

%%Load All of the Inputs for this simulation
Initializations; %Initialize the global variables

%%Compute the 2 important parameters based on RFP requirements
Wing_Loading; %Compute the wing loadings
Thrust_Weight_Ratio; %Compute the Thrust to Weight Ratio

%%Compute the Power-plant weight
Aircraft_Total_Mass; %Compute the total aircraft mass based on thrust to weight and engine choice
Get_Batt_Mass; %Compute the battery mass needed for correct flight endurance

%%Compute the Weight of the lift devices
Aircraft_Wing_Area; %Compute the Area of the Main Wing Device
Main_Wing_Mass; %Compute the mass of the Main Wing

%%Compute Spare Weight Quota Available
Spare_Weight_Available; %Computes whatever mass is left for all the other components
```

```
%%Show All Results
Post_Processing; %Print out All of the things we want to know about
```

The rest of this document documentation will proceed in the same order the scripts within `Main.m` are called.

4 Stall Speed Constraint

Wing loading is basically the weight of the aircraft divided by the total wing area.

$$L_{wi} = \frac{W}{S} = \frac{1}{2}\rho v_{stall}^2 c_{L,max} \tag{1}$$

Our typical would be around $c_{L,max} \approx 1.2 \to 1.5$, Sweep only reduces your maximum coefficient of lift. The `Matlab` implementation is shown below,

```
%%Give Definition of Global variable
Global_Variables;

%This Matlab script is used to compute the wing loading using the stall speed of the aircraft.
%We are getting the stall speed from the Requirements for the aircraft.
L_wi = 1/2*rho*(v_stall^2)*c_Lmax;
```

5 Climb Rate Constraint

The cimb rate G is defined as the ratio between vertical and horizontal distance travelled when the aircraft is climbing D is going to represent drag.

$$\frac{D}{W} = \frac{T}{W} - G \tag{2}$$

There is also another relation for drag to weight ratio,

$$\frac{D}{W} = \frac{qc_{D,0}}{L_{wi}} + L_{wi}\frac{1}{q\pi A_r e} \tag{3}$$

wherein A_r is the aspect ratio of the wings, and e represents the efficiency factor due to deviating from the elliptic lift distribution. Substituting equation 2 into equation 3 and solving for $\frac{T}{W}$,

$$\frac{D}{W} = \frac{T}{W} - G = \frac{qc_{D,0}}{L_{wi}} + L_{wi}\frac{1}{q\pi A_r e}$$

$$\frac{T}{W} = \frac{qc_{D,0}}{L_{wi}} + L_{wi}\frac{1}{q\pi A_r e} + G$$

wherein $c_{D,0}$ represents the zero-lift drag coefficient of an aircraft. For a clean propeller aircraft, $c_{D,0} = 0.02$. For a dirty propeller aircraft, then $c_{D,0} = 0.03$. The expression for thrust to weight ratio computations are implemented below,

```
%%Give Definition of Global variable
Global_Variables;

%This computes the thrust to weight ratio based on the climb rate
T_W  = (q*c_D0)/(L_wi) + L_wi/(q*pi*A_r*e) + G;
```

6 Computing Aircraft Total Mass

Supposing we know thrust to weight,

$$T_w = \frac{T}{W}$$

and we know the thrust T, it is a simple operation to get the total weight of the aircraft (including the payload)

$$W = T \times \frac{W}{T} = T \times \frac{1}{T_w} = \frac{T}{T_w}$$

That is what is being implemented in the `Matlab` script below. Here `T_W` is a variable that corresponds to thrust to weight ratio. `Engine_Thrust` literally means how much thrust the engine makes, since we only have 1 engine, this will be the thrust for the entire aircraft.

```
%%Give Definition of Global variable
Global_Variables;

Total_Aircraft_Weight = Engine_Thrust/T_W; %Plane weight in Newtons. Weight = Thrust/(Thrust/Weight)
Total_Aircraft_Mass = Total_Aircraft_Weight/g; %Mass in kg, Mass = Weight/g
```

7 Computing Aircraft Batteries

This is where the sizing algorithm approximates the truth but could be developed further. We hope to know the thrust needed for the aircraft at cruise conditions or could estimate it well. We also know that we want the aircraft to have some level of endurance in the air based on the RFP. The first step is determining how much total energy the batteries in the aircraft needs to store. To do that,

$$E_{batt} = P_{motor} T_e$$

wherein E_{batt} represents the amount of energy the battery has to store, and P_{motor} is the power of the motor during cruise and T_e is the endurance time of the aircraft. After figuring out how much energy the battery has to hold in total, we can determine the battery weight if we also know the specific battery energy. The batteries we chose are li-po batteries and the average of the lower bound and upper-bound of the value in Wikipedia was used as an estimate for the specific battery energy of our li-po batteries.

$$M_{batt} = E_{batt} \times \frac{M_{batt}}{E_{batt}} = E_{batt} \times \frac{1}{\rho_{batt,E}} = \frac{E_{batt}}{\rho_{batt,E}}$$

wherein M_{batt} represens the mass of the batteries, $\rho_{batt,E}$ represents the specific battery energy density. This is what the `Matlab` script used to estimate the battery size is doing,

```
%%Give Definition of Global variable
Global_Variables;

%Based on Motor power draw, figure out total energy needed
Batt_Energy = Motor_Power*endurance; %Batt_Energy must have units of J = J/s*s

%Based on how much energy the battery must contain and the specific energy of battery,
Batt_Mass = Batt_Energy/Batt_Specific_Energy; %Units of kg = J/(J/kg)
```

The variable names are extremely self-explanatory.

8 Computing Aircraft Main Wing

Before we attempt to compute weight or mass of the aircraft's main wing, we have to compute the area of the aircraft's main wing. This is possible because we know the total loaded weight of the aircrft, and we also know the prescribed wing loading for the aircraft. Reiterating the definition for wing loading (equation 1),

$$L_{wi} = \frac{W}{S} = \frac{1}{2}\rho v_{stall}^2 c_{L,max}$$

Here L_{wi} represents the wing loading, W represents the total loaded weight of the aircraft and S represents the surface area of the lifting surfaces. By extremely simple algebraic manipulation,

$$S = \frac{W}{L_{wi}}$$

Therefore, if we know the aircraft weight W and also the prescribed wing loading L_{wi} then we can compute the surface area of the main wing. The `Matlab` implementation for the main wing's surface area is shown below,

```
%%Give Definition of Global variable
Global_Variables;

Main_Wing_Area = Total_Aircraft_Weight/L_wi; %Area must be in m^2
```

After computing the surface area of the main wing, we also assume that the wing is a particular thickness, this should be studied carefully to fit to structural integrity, but for now we believe this thickness is reasonable for a wing. From this thickness, we can compute the approximate wing volume and since we have decided to use a foam composite wing, we can multiply density of foam and resin to the volume of the wing to estimate the weight of the main wing.

$$M_{m,wing} = V_{m,wing} \times \rho_{m,wing}$$

wherein $M_{m,wing}$ represents the mass of the main wing, $V_{m,wing}$ represents the volume of the main wing, $\rho_{m,wing}$ represents the approximate density of the main wing. This is kind of hand-wavy and can be further studied to improve the accuracy. The `Matlab` implementation of this,

```
%%Give Definition of Global variable
Global_Variables;

Main_Wing_Volume = Main_Wing_Area * Main_Wing_Thickness; % Units in m^3
mass_main_wing = Ave_Density_Main_Wing * Main_Wing_Volume; %Units in kg
weight_main_wing = mass_main_wing*g; %Units in Newtons
```

9 Computing Spare Weight Available

Computing the spare weight is basically just taking the aircraft's total loaded weight and subtracting all of the components we already know. The remaining mass should be whatever is left for the fuselage, tail and whatnot. So we use this algorithm as a sanity check. If we are investing this much weight for the main wings, can we make a corresponding fuselage with this much remaining weight? And so on. The `Matlab` script which computes the spare weight is shown below,

```
%%Give Definition of Global variable
Global_Variables;
```

```
%Initialize the Spare Weight to the Total Aircraft Weight
Spare_Aircraft_Weight = Total_Aircraft_Weight;

%Subtract the Propulsion System
Spare_Aircraft_Weight = Spare_Aircraft_Weight - Batt_Mass*g - Motor_Prop_Mass*g;

%Subtract the Lift Devices
Spare_Aircraft_Weight = Spare_Aircraft_Weight - weight_main_wing;

%Subtract the Control System
Spare_Aircraft_Weight = Spare_Aircraft_Weight - mass_avionics*g;

%Subtract the Intended Payload
Spare_Aircraft_Weight = Spare_Aircraft_Weight - payload_mass*g;

%Convert the Spare Weight into spare Mass
Spare_Aircraft_Mass = Spare_Aircraft_Weight/g;
```

10 Post-Processing

This `Matlab` script handles the text output of the variables we are interested in.

```
%%Give Definition of Global variable
Global_Variables;

fprintf("##################################################################\n");
fprintf("Wing Loading for Aircraft: %f (N/m^2)\n", L_wi);
fprintf("Thrust to Weight Ratio: %f (dimless)\n", T_W);
fprintf("##################################################################\n");
fprintf("Battery Mass: %f kg\n", Batt_Mass);
fprintf("##################################################################\n");
fprintf("Aircraft Wing Area: %f m^2\n", Main_Wing_Area);
fprintf("Main Aircraft Wing Mass: %f kg\n", mass_main_wing);
fprintf("##################################################################\n");
fprintf("The Total Mass of the Aircraft: %f kg\n", Total_Aircraft_Mass);
fprintf("The Spare Mass of the Aircraft: %f kg\n", Spare_Aircraft_Mass);
fprintf("##################################################################\n");
fprintf("# End of Analysis\n")
fprintf("##################################################################\n");
```

11 Algorithm Outputs

Below shows the current output of the entire sizing algorithm,

```
##################################################################
Wing Loading for Aircraft: 47.040000 (N/m^2)
Thrust to Weight Ratio: 0.544359 (dimless)
##################################################################
Battery Mass: 0.274809 kg
##################################################################
Aircraft Wing Area: 1.562094 m^2
Main Aircraft Wing Mass: 2.714924 kg
##################################################################
The Total Mass of the Aircraft: 7.490406 kg
The Spare Mass of the Aircraft: 3.438673 kg
##################################################################
```

```
# End of Analysis
##################################################################
>>
```

12 Further Developments

Definitely many parts of the algorithm can be improved to increase the accuracy. The most important thing though, is to size the horizontal stabilizers next. We also need to figure out pretty soon which airfoil we would like to use because much of the design is very much dependent on the main wing's airfoils, including stability. If the main airfoil has an adverse pitching moment, we have to think about how large the tail has to be and what effect this will have on the overall wing-loading of the aircraft. Also power motor ratings should be thought more clearly.

13 Estimation of Motor Power Rating

Currently, this is not too important although it may change in the future. We have not had the time to implement this feature into the code-base. The equation below was found in page 118,

$$\frac{T}{W} = \left(\frac{\eta_p}{V}\right)\left(\frac{P}{W}\right)$$

wherein η_p represents the propeller efficiency, P represents power of the engines, W represents the weight of the aircraft, V represents the true air speed of the aircraft.

14 Appendix

14.1 Common Theoretical Definitions

q is the dynamic pressure of an aircraft flying. It is defined as

$$q = \frac{1}{2}\rho v^2$$

wherein ρ represents the density of the air, v represents the velocity the aircraft is flying at.

14.2 Textual References

1. Page 117 has table for thrust to weight of typical aircrafts.
2. Page 119 has a table for power to weight ratio of typical aircrafts.
3. Page 124 has a tabel for typical wing loadings.
4. Page 126 has description on typical $c_{L,max}$ values.
5. Page 135 has descriptions of what $c_{D,0}$ should be.