**AER1216**: Fundamentals of UAVs Project

Arnav Wadekar, 1003883443

Task: Design an Unmanned Aerial Vehicle (UAV) for a surveying task. The task requires the UAV to look for brown spots on a green field patch of 0.2 km x 0.5 km.

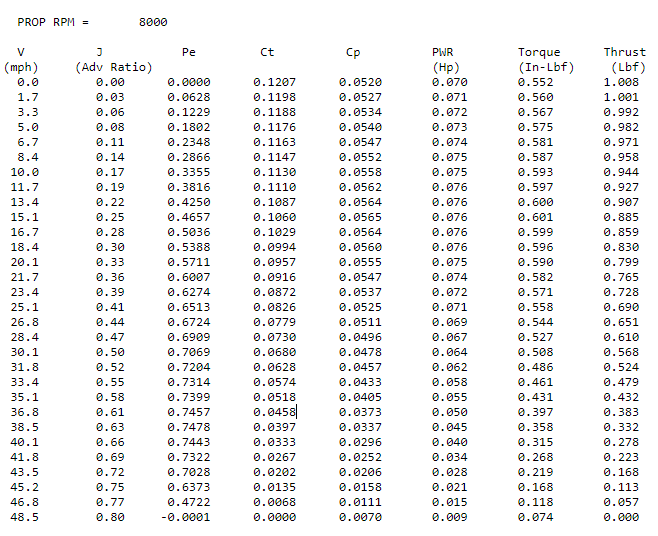
Details: The UAV has a camera attached to it when flying over the field. There are different flight paths that can be taken by the UAV and the goal is to find out which path and combinations result in optimal flight conditions and time. The camera can survey a width of 5 meters when flying at a constant height over the field and can record at a maximum speed of 15 m/s successfully.

Choices: The following table highlights the customization choices available and the choices made:

|  |  |  |
| --- | --- | --- |
| **Component/Parameter:** | **Choices Available:** | **Chosen:** |
| Configuration | Quadrotor/Fixed Wing | Quadrotor |
| Propellers | APC8x4.5MR  APC8x4.7SF  APC9x3.8SF  APC9x4.5MR  APC9x6SF  APC9x6E | APC9x6SF |
| Motors | KDE-2315-F885  AXI 2212/34v2  AXI 2212/26v2  AXI 2212/20v2  AXI 2212/12v2 | KDE-2315-F885, Kv = 885 RPM/V, rm = 0.127 Ohms, i0 = 0.5 A, imax = 24 A. |
| Battery | Turnigy LiPo Battery (2, 3, 4 cells XXX, 30C) | 3600 mAh, 11.1 V, 3 cell, 30C, 0.321 kg |
| Camera | N/A | 0.5 kg, Δx = 5 m |
| Frame | No Fairings  With Fairings | 350 mm, With Fairings, 0.035 m2, 1.33 kg |

Choice Reasoning: Each choice had a reasoning and starting from top to bottom:

The frame was chosen to be with fairings as even though that added weight, the reduction in coefficient of drag was hypothesized to be more beneficial to the power required for the motors and thus how much charge the quadrotor would use to propel itself. Next, the camera was by default stated as 0.5 kg and thus no choice was made. The battery was chosen to be the Turnigy LiPo Battery with 3600 mAh. Although it is heavier than the 1500 mAh by almost twice the weight, it has twice the capacity and it was considered crucial so that the quadrotor needed to stop and charge minimal number of times during its flight. The motor was chosen to be the KDE-2315-F885 because of its characteristics of having a relatively lower Kv as well as a low r­m. Having these values lower helps the efficiency. In addition to that, the motor is a well known one in the industry and has been performing well even in expensive recreational quadrotors. The propeller was chosen using the data obtained from the manufacturer’s website as well as the data on the motors. The form of the data was in the following format (8x4.5MR, rest available in appendix) for every 1000 RPM increment:

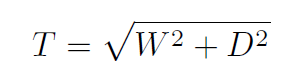


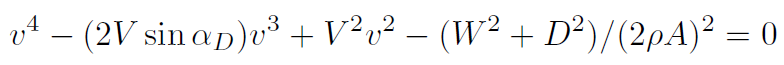
Keeping in mind that 15 m/s was approximately 33 mph, the RPM where the maximum efficiency (Pe) of the propeller was achieved at approximately 33 mph was chosen. Doing this for all propellers, the following table can be created:

|  |  |  |
| --- | --- | --- |
| **Propeller** | **RPM** | **Efficiency (PE)** |
| 8x4.5MR | 8000 | 0.7314 |
| 8x4.7SF | 7000 | 0.7308 |
| 9x3.8SF | 8000 | 0.7067 |
| 9x4.5MR | 7000 | 0.7301 |
| 9x6SF | 5000 | 0.7610 |
| 9x6E | 6000 | 0.7636 |

Using the table above and the charts available with the motor’s performance specifications, the 9x6SF was chosen because of it has high efficiency at even lower RPMs. In addition to that, the thrust produced was high even at the lower RPMs compared to the 8 series and another 9 series.

This meant that now the masses could be added up (ignoring mass of propeller) and becomes Mtotal = 4\*0.075 + 0.335 + 0.5 + 1.33 = 2.465 kg. Thus, the total weight of the quadrotor UAV is 24.1817 N with g = 9.81 m/s2.

The methodology to calculate the total power for each flight path is applied to all of them. First, there is the general power calculations of the quadrotor when it is travelling in a straight line. Using forward flight momentum theory, this equation is required for steady state forward flight: .

The tilt angle of the quadrotor then can be found by  where D is the drag force = 0.5\*rho\*CdS\*Vi^2. Once those are found, the parameters can be plugged into the following equation: 

The roots for the induced velocity are then found by solving the above equation. For the chosen design of this quadrator, the induced velocity was calculated to be 2.2127 m/s. Next, assuming that the following flight path calculations are already done, the induced velocity can be used in this equation to find the total power of level straight flight:



Where T was calculated earlier. For this quadrotor, it was found to be 54.5599 W.

Next, the losses need to be considered and thus the battery statistics come into play. Using the following equations, the parameters are calculated for power needed to hover with losses (calc) and the power needed to hover with actuator disc theory (theory):

T\_hover = Wt/4;

n\_hover = sqrt(T\_hover/(rho\*C\_t\*propdiam^4));

n\_hover\_rpm = n\_hover\*60;

P\_hover\_calc = C\_p\*rho\*n\_hover^3\*propdiam^5;

i\_m = (Q/K\_t)+i\_0;

v\_m = ((n\*2\*pi)/K\_v\_motor)+i\_m\*r\_m;

omega\_elec = K\_v\_motor\*(v\_m-i\_m\*r\_m);

P\_in = omega\_elec\*Q;

i\_e0 = 4\*i\_m;

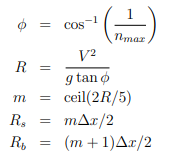
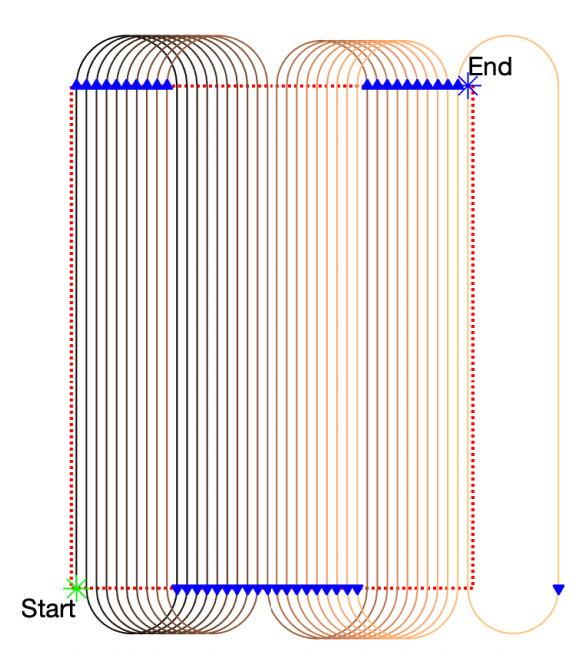
v\_e0 = v\_m + i\_e0\*ESC\_r\_e;

P\_hover\_theory = sqrt((Wt^3)/2\*rho\*totproparea);

Totproparea is the propeller area of all 4 propellers. These calculated powers, along with the calculated ESC and motor efficiencies (code in appendix) are then divided by each other to find the reduction in power. The effect of hover vs 15 m/s is also considered a 107.5 increase in required power and that is also applied. The reduction in power for this quadrotor was calculated to be 0.7914. This means that the power in level flight will be increased by 1.26 times to account for the losses not covered by the theory used to calculate the power. These steps are then repeated for each flight path and will be covered below. Once the total power required for each flight path is found, it is divided by this reduction in power calculated using the hover theory to simulate the losses incurred and thus that is how the final total power of each flight path is calculated.

Flight Paths: There were 4 available flight paths with a constant speed, and since a quadrotor configuration was chosen, an additional flight path was also considered. These flight paths will be referred to as A, B, C, D, and E with E being the additional path and A, B, C, D being the ones explained in the project handout.

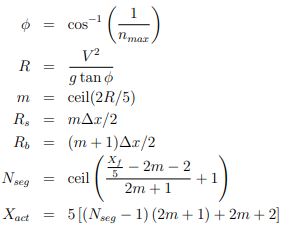
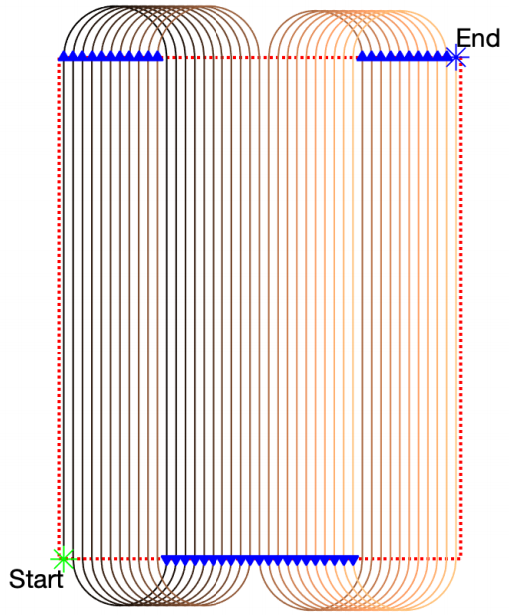
Flight Path A: Flight path A can be seen below:



This flight path starts at the start point, does 1 segment which is half of the field and then begins on the right-hand side of the diagram. It ends with one loop outside the field and at the end point. The equations on the right are used to calculate the smaller and bigger turning radii. The turn angle is used in the calculations for the power used while turning. To ensure the load factor for both turns do not exceed 1.5, they are calculated using

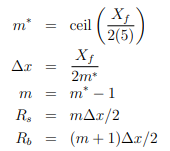
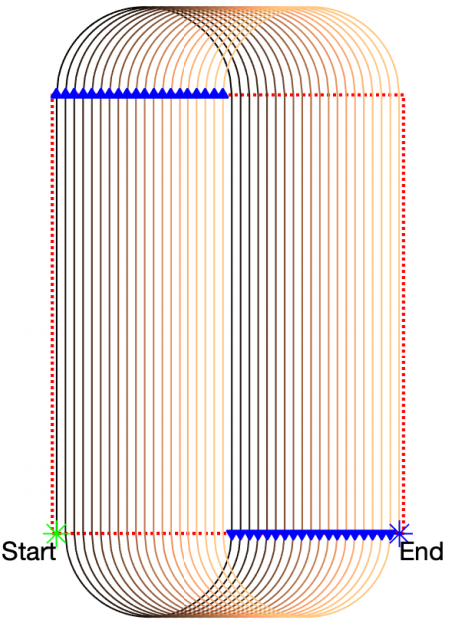
n\_a\_s = sqrt(((Vi^2/(g\*R\_a\_s))^2)+1); where n\_a\_s is the load factor for a small turn, and R\_a\_s is the radius of the small turn. None of the flight paths exceeded a load factor of 1.5 during a turn and hence the calculations could be continued. Next, the load factor is used along with the calculated tilt angle to find out the thrust require to turn. This thrust is then used to find the required power with the total power equation mentioned above. The same steps are repeated for each flight path except E which will be talked about later. The diagrams for B, C, and D are below:

Flight path B:



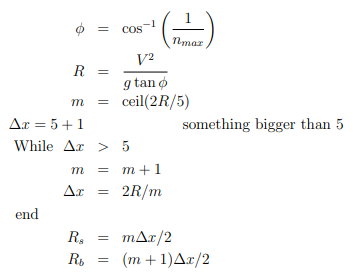
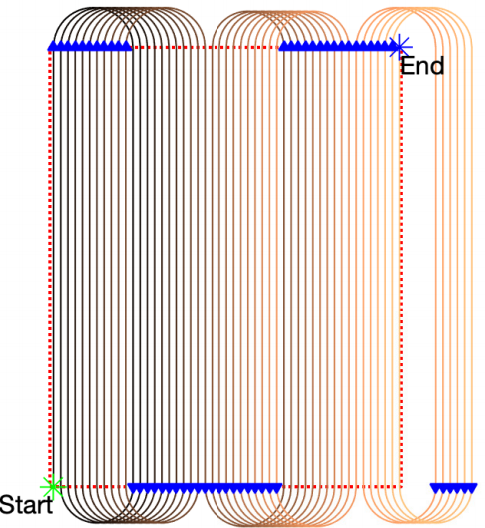
This flight path attempts something like flight path A but does not have the excess run off towards the end of its journey.

Flight Path C:



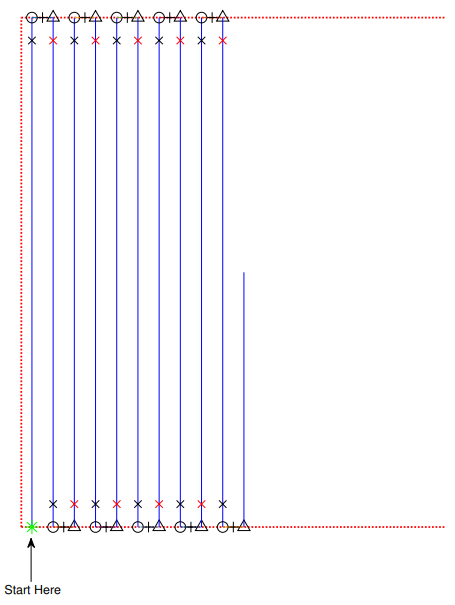
This flight path attempts to cover the entire field without breaking it into sections. It does an oval loop and shifts to the side on every pass.

Flight Path D:



This flight path is an addition to A and has 3 major segments with a lot of additional passes outside the green field required.

Flight Path E:



This flight path is unique as it does not require turns to navigate over the field. It is also unique because it is not at a constant velocity. It enters with a velocity at every black cross and plus sign and then begins to stop. At every black triangle and circle, it begins to accelerate. These maneuvers need extra power to speed up and slow down every time but are extremely time efficient and can navigate the field without extra passes and in one go. For this flight path, a code was provided that was used to calculate the power required for each acceleration and deceleration event on the path. Dynamic inflow model in the code was used. The straight-line power between the black and red crosses is calculated using the same method as before. The power calculated for each event can be summarised by the table below:

|  |  |  |  |
| --- | --- | --- | --- |
| Event | Acceleration/Deceleration (m/s2) | Time (s) | Power (W) |
| Acceleration along width (circle to plus) | 5 | 1 | 179.8745 |
| Deceleration along width (plus to triangle) | 5 | 1 | 142.9732 |
| Acceleration along length (triangle to red cross) | 11.25 | 1.3333 | 544.8815 |
| Deceleration along length | 11.25 | 1.3333 | 89.5903 |

Conclusion: For all the flight paths, the distances are calculated by using the number of turns times the distance turned (pi\*R for perimeter of outside of semicircle) along with the distances travelled in a straight line and adding them up. The time can then be calculated by dividing the total distance by the velocity. For flight path E, the straight distances are added up by the amount of parallel distance. The distance of the turns and width are also added up to that value to then find the total distance. The time for the straight distance is the same and then the calculated time above is used to add to the final value. Below is a summary of distance and time:

|  |  |  |
| --- | --- | --- |
| Flight Path | Distance Total (m) | Total Time (s) |
| A | 23484.51 | 1565.634 |
| B | 24729.27 | 1648.618 |
| C | 25976.88 | 1731.792 |
| D | 31160.64 | 2077.376 |
| E | 18915 | 181.9974 |

Now that all the calculations for power have been done for respective flight paths and the general steady flight, they can be added up to form the following table:

|  |  |  |
| --- | --- | --- |
| Flight Path | Total Power Required (W) | Total Power Required with losses (W) |
| A | 487.369 | 724.5415 |
| B | 482.1113 | 716.7252 |
| C | 408.5692 | 607.3946 |
| D | 507.2291 | 754.0662 |
| E | 1084.231 | 1611.859 |

As visible above, the least power required is for flight path C. This makes sense as it does the least amount of turns out of the flight paths that have turns however it is close to the other flight paths in terms of power consumed. Flight path E although the least distance, is the most power consuming for one segment because of its nature of having to slow down and speed up again whereas the other flight paths keep going at a constant speed thus requiring less continuous power. Where flight path E excels, however, is the time taken to complete the survey as can be seen above.

Using the 0th model battery equation and multiplying the total power by time to find total energy, the following table can be made:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Flight Path | Total Power Required (W) | Total Power Required with losses (W) | Total Energy Required (J) | 0th Order Energy (J) | Energy ratio |
| A | 487.3690116 | 724.5414852 | 1134366.93 | 143856 | 7.885433556 |
| B | 482.1113114 | 716.7251862 | 1181605.804 | 143856 | 8.213809674 |
| C | 408.5691924 | 607.3946485 | 1051881.194 | 143856 | 7.312042556 |
| D | 507.2290528 | 754.0661851 | 1566478.865 | 143856 | 10.88921466 |
| E | 1084.231001 | 1611.859436 | 293354.2264 | 143856 | 2.039221349 |

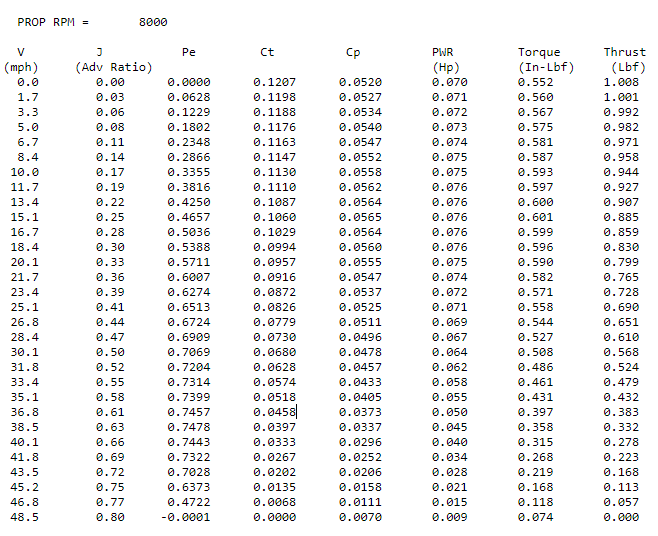
This table is a conclusion to the calculations and all flight plans. As can be seen on the very right column, the energy ratio is how many times more energy the drone requires to complete each segment. Although flight path E had the highest power required, it has the lowest energy ratio with only 2.039 charges required to complete the flight plan. Although this would mean 3 full charges are required, it is still much lower than the next one with 8 charges for C and A. Flight path D is the least effective with 11 charges required to complete a survey. In order of most efficient to least efficient, the flight paths are E, C, A, B, D.

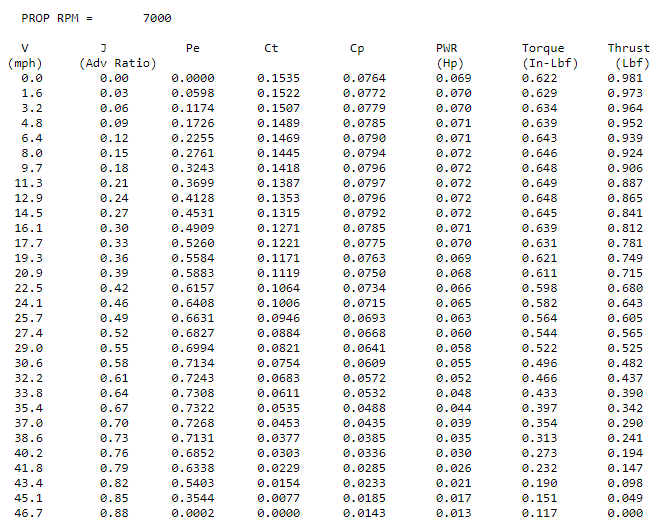
Since charging the quadrotor will be required, the take off and landing methods will have to be considered. The start point could serve as a charging station dock as well as the command center of the drone. Once it is close to the end of its battery life, the drone could move straight (least distance) to the dock from wherever it is and land to begin charging. It would then take off and take the same path to its last surveyed point and restart its cycle. Since flight path E only requires 3 charges, it would be possible to split it into 3 segments and then code the drone to perform each segment and come back to charge when it is low on battery. The conditions required for this design to work should be close to ideal. That is, a day with not too much crosswind/turbulence, no rain, relatively clear skies so the camera can take high quality images and no trees around the perimeter of the green field.

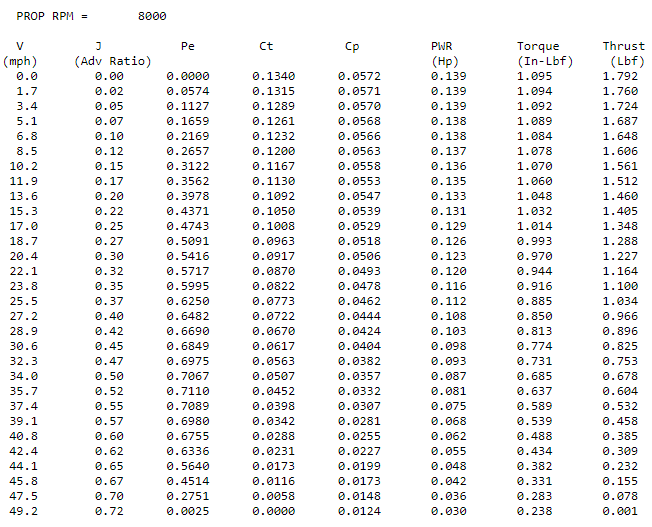
In conclusion, this design performs well as only 3 charges are required for flight path E. It completes the task in a short amount of time excluding the take off, charging, and landing times. Improvements to this analysis could be done in the selection method with the propeller and iteratively solving for each propeller and motor. In addition to that, the deceleration distance for flight path E could be reduced even further however that would result in the load factor coming close to the maximum and thus would have to be done carefully.

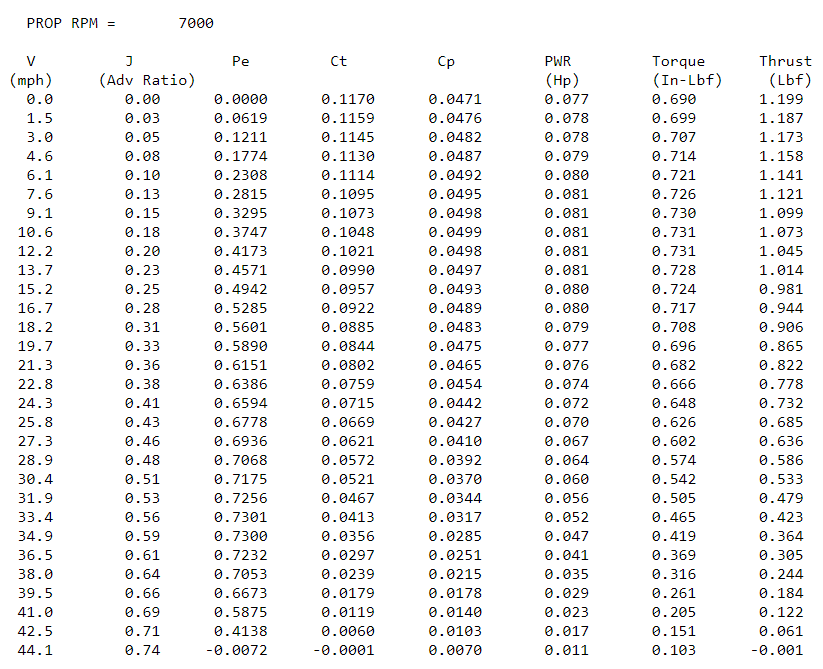
**Appendix:**

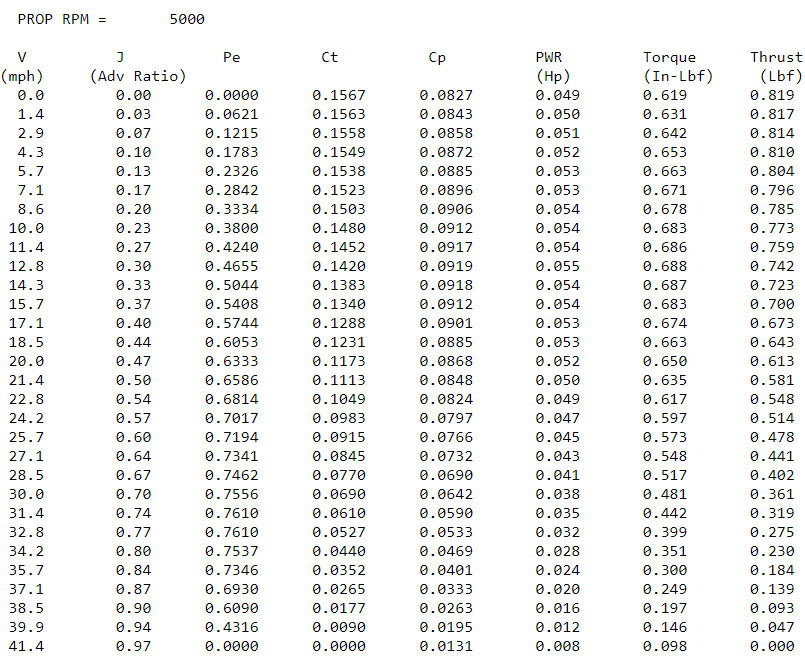
Propeller Data:

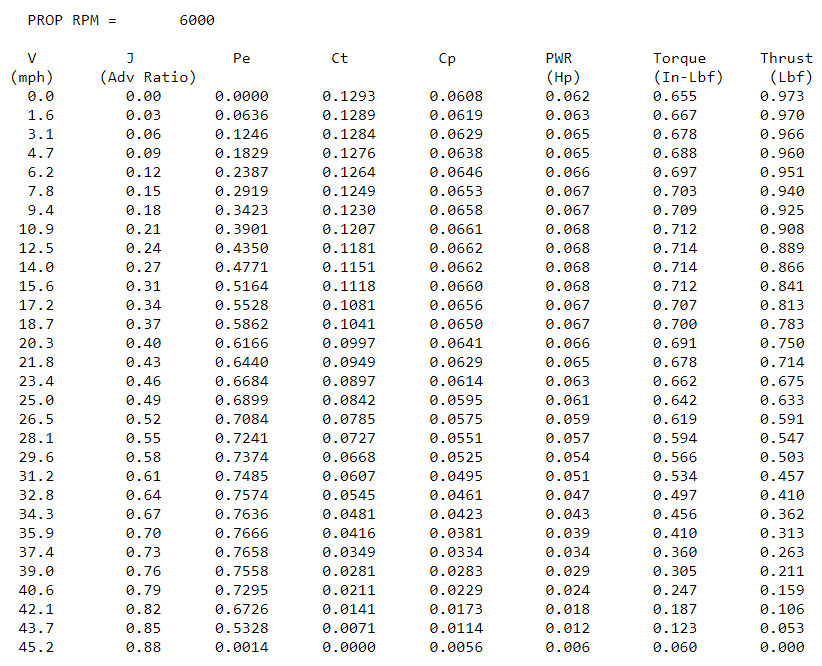
8x4.5MR:  


8x4.7SF:  
 

9x3.8SF:  
 

9x4.5MR:  


9x6SF:  
 

9x6E:  


tic

clear

clc

%With Fairings

rho = 1.225;

C\_DS = 0.0350; %m^2

M\_drone = 1.3300; %kg

M\_cam = 0.5; %kg

M\_batt = 0.335; %kg

M\_motor = 0.075\*4;

M = M\_drone + M\_cam + M\_batt + M\_motor; %kg

Vi = 15; %m/s

n\_max = 1.5;

g = 9.81;

X\_f = 200; %m

Wt = M\*g;

%propeller

propdiam = 0.1524\*2;

proparea = pi\*propdiam^2/4;

totproparea = 4\*proparea;

%motor chosen with low voltage and amp requirements: AXI2212/34v2

K\_v\_motor = (885/60)\*2\*pi; %RPM/V

r\_m = 0.127; %Ohm

i\_0 = 0.5; %Amps

i\_max = 24; %Amps

K\_t = 1/K\_v\_motor;

nu\_e = 0.85;

nu\_m = 0.95;

ESC\_r\_e = 0.5;

omega = 7000;

n = omega/60;

C\_p = 0.0903;

C\_t = 0.1577;

C\_q = C\_p/(2\*pi);

Q = C\_q\*rho\*n^2\*propdiam^5;

T\_hover = Wt/4;

n\_hover = sqrt(T\_hover/(rho\*C\_t\*propdiam^4));

n\_hover\_rpm = n\_hover\*60;

P\_hover\_calc = C\_p\*rho\*n\_hover^3\*propdiam^5;

i\_m = (Q/K\_t)+i\_0;

v\_m = ((n\*2\*pi)/K\_v\_motor)+i\_m\*r\_m;

omega\_elec = K\_v\_motor\*(v\_m-i\_m\*r\_m);

P\_in = omega\_elec\*Q;

i\_e0 = 4\*i\_m;

v\_e0 = v\_m + i\_e0\*ESC\_r\_e;

P\_hover\_theory = sqrt((Wt^3)/2\*rho\*totproparea);

motor\_eff = P\_in/(i\_m\*v\_m);

esc\_eff = v\_m/v\_e0;

P\_reduction = P\_hover\_theory/(P\_hover\_calc);

%misc

drag = 0.5\*rho\*C\_DS\*Vi^2;

thrust = sqrt(Wt^2+drag^2);

alpha\_d\_straight = -atan(drag/Wt);

%coefv\_ind = [1, -(2\*Vi\*sin(alpha\_d\_straight)), Vi^2, 0, -((Wt^2+drag^2)/2\*rho\*totproparea)];

%v\_ind = roots(coefv\_ind);

V = Vi;

for i=1:length(V)

alpha(i) = atan(-drag/Wt);

T(i) = sqrt(Wt^2+drag^2);

C(1) = 1;

C(2) = -2\*V(i)\*sin(alpha(i));

C(3) = V(i)^2;

C(4) = 0.0;

C(5) = -1\*T(i)^2/(2\*rho\*totproparea)^2;

R = roots(C);

%

% find positive real root

%

tol = 1e-9;

for j=1:4

if(abs(imag(R(j))) < tol)

if real(R(j)) > 0

%

% induced velocity

%

v(i) = real(R(j));

break;

end

end

end

end

v\_ind = v;

%Flight Path A (done twice)

delta\_x\_a = 5;

phi\_a = acos(1/n\_max);

R\_a = (Vi^2)/(g\*tan(phi\_a));

m\_a = ceil(2\*R\_a/5);

R\_a\_s = m\_a\*delta\_x\_a/2;

R\_a\_b = (m\_a+1)\*delta\_x\_a/2;

n\_a\_s = sqrt(((Vi^2/(g\*R\_a\_s))^2)+1);

n\_a\_b = sqrt(((Vi^2/(g\*R\_a\_b))^2)+1);

phi\_a\_s = 1/cos(n\_a\_s);

phi\_a\_b = 1/cos(n\_a\_b);

alpha\_d\_a\_s = -atan(drag/Wt\*n\_a\_s);

alpha\_d\_a\_b = -atan(drag/Wt\*n\_a\_b);

thrust\_a\_s = sqrt(Wt^2\*n\_a\_s+drag^2);

thrust\_a\_b = sqrt(Wt^2\*n\_a\_b+drag^2);

P\_turn\_a\_s = thrust\_a\_s\*(v\_ind-Vi\*sin(alpha\_d\_a\_s));

P\_turn\_a\_b = thrust\_a\_b\*(v\_ind-Vi\*sin(alpha\_d\_a\_b));

P\_ind\_a\_st = thrust\*v\_ind;

P\_tot\_a\_st = thrust\*(v\_ind-Vi\*sin(alpha\_d\_straight));

P\_tot\_a = P\_tot\_a\_st + P\_turn\_a\_s + P\_turn\_a\_b;

P\_tot\_a\_loss = P\_tot\_a/(P\_reduction\*nu\_e);

%Flight Path B

delta\_x\_b = 5;

phi\_b = acos(1/n\_max);

R\_b = (Vi^2)/(g\*tan(phi\_b));

m\_b = ceil(2\*R\_b/5);

R\_b\_s = m\_b\*delta\_x\_b/2;

R\_b\_b = (m\_b+1)\*delta\_x\_b/2;

N\_seg = ceil((((X\_f/5)-2\*m\_b-2)/(2\*m\_b+1))+1);

X\_act = 5\*((N\_seg-1)\*(2\*m\_b+1)+2\*m\_b+2);

delta\_x\_b = 5 + 1; %something larger than 5

N\_seg\_new=N\_seg-1;

while delta\_x\_b > 5

m\_b\_new = m\_b+1;

delta\_x\_b =(X\_f)/((N\_seg\_new-1)\*(2\*m\_b\_new+1)+2\*m\_b\_new+2);

end

R\_b\_s\_new = m\_b\_new\*delta\_x\_b/2;

R\_b\_b\_new = (m\_b\_new+1)\*delta\_x\_b/2;

n\_b\_s = sqrt(((Vi^2/(g\*R\_b\_s\_new))^2)+1);

n\_b\_b = sqrt(((Vi^2/(g\*R\_b\_b\_new))^2)+1);

phi\_b\_s = 1/cos(n\_b\_s);

phi\_b\_b = 1/cos(n\_b\_b);

alpha\_d\_b\_s = -atan(drag/Wt\*n\_b\_s);

alpha\_d\_b\_b = -atan(drag/Wt\*n\_b\_b);

thrust\_b\_s = sqrt(Wt^2\*n\_b\_s+drag^2);

thrust\_b\_b = sqrt(Wt^2\*n\_b\_b+drag^2);

P\_turn\_b\_s = thrust\_b\_s\*(v\_ind-Vi\*sin(alpha\_d\_b\_s));

P\_turn\_b\_b = thrust\_b\_b\*(v\_ind-Vi\*sin(alpha\_d\_b\_b));

P\_ind\_b\_st = thrust\*v\_ind;

P\_tot\_b\_st = thrust\*(v\_ind-Vi\*sin(alpha\_d\_straight));

P\_tot\_b = P\_tot\_b\_st + P\_turn\_b\_s + P\_turn\_b\_b;

P\_tot\_b\_loss = P\_tot\_b/(P\_reduction\*nu\_e);

%Flight Path C

m\_star = ceil(X\_f/(2\*5));

delta\_x\_c = X\_f/(2\*m\_star);

m\_c = m\_star-1;

R\_c\_s = m\_c\*delta\_x\_c/2;

R\_c\_b = (m\_c+1)\*delta\_x\_c/2;

n\_c\_s = sqrt(((Vi^2/(g\*R\_c\_s))^2)+1);

n\_c\_b = sqrt(((Vi^2/(g\*R\_c\_b))^2)+1);

phi\_c\_s = 1/cos(n\_c\_s);

phi\_c\_b = 1/cos(n\_c\_b);

alpha\_d\_c\_s = -atan(drag/Wt\*n\_c\_s);

alpha\_d\_c\_b = -atan(drag/Wt\*n\_c\_b);

thrust\_c\_s = sqrt(Wt^2\*n\_c\_s+drag^2);

thrust\_c\_b = sqrt(Wt^2\*n\_c\_b+drag^2);

P\_turn\_c\_s = thrust\_c\_s\*(v\_ind-Vi\*sin(alpha\_d\_c\_s));

P\_turn\_c\_b = thrust\_c\_b\*(v\_ind-Vi\*sin(alpha\_d\_c\_b));

P\_ind\_c\_st = thrust\*v\_ind;

P\_tot\_c\_st = thrust\*(v\_ind-Vi\*sin(alpha\_d\_straight));

P\_tot\_c = P\_tot\_c\_st + P\_turn\_c\_s + P\_turn\_c\_b;

P\_tot\_c\_loss = P\_tot\_c/(P\_reduction\*nu\_e);

%Flight Path D

phi\_d = acos(1/n\_max);

R\_d = (Vi^2)/(g\*tan(phi\_d));

m\_d = ceil(2\*R\_d/5);

delta\_x\_d = 5+1;

while delta\_x\_d > 5

m\_d\_new = m\_d + 1;

delta\_x\_d = (2\*R\_d)/m\_d\_new;

end

R\_d\_s\_new = m\_d\_new\*delta\_x\_d/2;

R\_d\_b\_new = (m\_d\_new+1)\*delta\_x\_d/2;

n\_d\_s = sqrt(((Vi^2/(g\*R\_d\_s\_new))^2)+1);

n\_d\_b = sqrt(((Vi^2/(g\*R\_d\_b\_new))^2)+1);

phi\_d\_s = 1/cos(n\_d\_s);

phi\_d\_b = 1/cos(n\_d\_b);

alpha\_d\_d\_s = -atan(drag/Wt\*n\_d\_s);

alpha\_d\_d\_b = -atan(drag/Wt\*n\_d\_b);

thrust\_d\_s = sqrt(Wt^2\*n\_d\_s+drag^2);

thrust\_d\_b = sqrt(Wt^2\*n\_d\_b+drag^2);

P\_turn\_d\_s = thrust\_d\_s\*(v\_ind-Vi\*sin(alpha\_d\_d\_s));

P\_turn\_d\_b = thrust\_d\_b\*(v\_ind-Vi\*sin(alpha\_d\_d\_b));

P\_ind\_d\_st = thrust\*v\_ind;

P\_tot\_d\_st = thrust\*(v\_ind-Vi\*sin(alpha\_d\_straight));

P\_tot\_d = P\_tot\_d\_st + P\_turn\_d\_s + P\_turn\_d\_b;

P\_tot\_d\_loss = P\_tot\_d/(P\_reduction\*nu\_e);

%Flight Path E

acc\_to\_Vi = 11.25;

dec\_to\_Vi = -11.25;

dist\_accdec\_st = 10;

P\_acc\_e\_st = 544.8815;

P\_dec\_e\_st = 89.5903;

Vint = 5;

Acc\_to\_Vint = 5;

Decc\_to\_Vint = 5;

dist\_accdec\_turn=2.5;

P\_acc\_e\_turn = 179.8745;

P\_dec\_e\_turn = 142.9732;

P\_tot\_e\_st = thrust\*(v\_ind-Vi\*sin(alpha\_d\_straight));

P\_tot\_e = P\_tot\_e\_st+P\_acc\_e\_st+P\_dec\_e\_st+P\_acc\_e\_turn+P\_dec\_e\_turn;

P\_tot\_e\_loss = P\_tot\_e/(P\_reduction\*nu\_e);

%Distance Traversed

%Flight Path A

distance\_turns\_a = 20\*pi\*R\_a\_b+20\*pi\*R\_a\_s;

distance\_length\_a = 41\*500;

distance\_tot\_a = distance\_turns\_a+distance\_length\_a;

time\_taken\_a = distance\_tot\_a/Vi;

%Flight Path B

distance\_turns\_b = 22\*pi\*R\_b\_b\_new+20\*pi\*R\_b\_s\_new;

distance\_length\_b = 43\*500;

distance\_tot\_b = distance\_turns\_b+distance\_length\_b;

time\_taken\_b = distance\_tot\_b/Vi;

%Flight Path C

distance\_turns\_c = 20\*pi\*R\_c\_b+19\*pi\*R\_c\_s;

distance\_length\_c = 40\*500;

distance\_tot\_c = distance\_turns\_c+distance\_length\_c;

time\_taken\_c = distance\_tot\_c/Vi;

%Flight Path D

distance\_turns\_d = 28\*pi\*R\_d\_b\_new+26\*pi\*R\_d\_s\_new;

distance\_length\_d = 55\*500;

distance\_tot\_d = distance\_turns\_d+distance\_length\_d;

time\_taken\_d = distance\_tot\_d/Vi;

%Flight Path E

delta\_x\_e = 5;

number\_e = (X\_f - 5)/delta\_x\_e;

distance\_length\_e = 480\*(X\_f - 5)/delta\_x\_e;

distance\_turns\_acc\_e = 2.5\*(((X\_f - 5)/delta\_x\_e));

distance\_turns\_dec\_e = 2.5\*(X\_f - 5)/delta\_x\_e;

distance\_tot\_e = distance\_length\_e+distance\_turns\_acc\_e+distance\_turns\_dec\_e;

time\_acc\_turn = 1;

time\_dec\_turn = 1;

time\_acc\_st = 1.3333;

time\_dec\_st = 1.3333;

time\_taken\_e = number\_e\*time\_dec\_turn+number\_e\*(time\_acc\_st+time\_dec\_st)+(number\_e)\*time\_acc\_turn;

%Energy

E\_b = 3\*3.7\*(3600/1000)\*3600;

t\_e = E\_b/(P\_tot\_e\_loss);

toc