Weekly report

1. **My *Goals* from last week**

* I would want to solve a linear programming problem. The objective function would be to minimize aX1 + bX2 where
* a = $200 (Smart Dart)
* b = $800 (hexapod)
* X1 = No. of Smart Darts needed
* X2 = No. of Hexapods needed

*How do I obtain other constraints? How do I bring in the Hexacopter flight time, cost into the constraints.*

* + - I would want to do some coding on MATLAB. If there is any possibility for this.
    - Make progress with the quadcopter pixhawk control.
    - Produce a basic outline for the conference paper.

1. **My *Accomplishments* this week**

* I have made a MATLAB program which partially fulfils the desired output. I have attached the code here.
* Prepared a presentation for Houston Robotics Day.
* Read about using Python to control the Quadcopter.

1. **My *Goals* for next week**

* Improve the code and make it paper ready. Get approval from Dr.Becker regarding the MATLAB code.
* Accomplish a flight test and obtain a video of GPS waypoint control.
  1. Meeting with Dr. Becker on Friday 11 A.M.

1. **What I need Dr. Becker to do:**

* Assistance with the MATALB code, how do I perform parallel animation? Correcting the time parameter and if possible improving the algorithm.

MATLAB code :-

function[] = Hetero\_Sensors(x,y,No\_Drones,No\_Hex)

global test

[X,Y] = meshgrid(0:10:x,0:10:y);

home\_base =[x/2,y/2];

plot(home\_base(1,1),home\_base(1,2),'rO');

xlabel 'X-distance (m)';

ylabel 'Y-distance (m)';

title 'Heterogenous Sensors Deployment Time';

hold on

set(gca,'Xlim',[0,100],'Ylim',[0,100]);

grid on;

Q=[];

for i=1:(x/10)+1

for j=1:(y/10)+1

d = norm(home\_base-[X(i,j),Y(i,j)],2); %norm dist from home base to other points

Q = [Q;X(i,j),Y(i,j),d]; % [x\_cord, y\_cord, euclid\_dist]

end

end

[~,I] = sort(Q(:,3));

test=[];

for i = 1:length(Q)

test = [test;Q(I(i,1),:)];

end

test = test(2:end,:);

%disp(test);

num =0;

t = 0;

while isempty(test) ~= 1

num = num+1;

[test,t] = Drone\_loc(test,No\_Drones,No\_Hex,home\_base,num,t);

end

disp('Total Time');

disp(t);

end

function[test,t] = Drone\_loc(test,No\_Drones,No\_Hex,home\_base,num,t)

A=zeros(2,2);

if num==1

for j =1:No\_Hex

A(1,:)=home\_base;

A(2,:) = test(j,1:2);

t1 = line\_equation(A(1,1),A(1,2),A(2,1),A(2,2),0);

end

t = t + t1;

test = test(No\_Hex+1:end,:);

%disp(length(test));

end

for i =1:No\_Drones

A(1,:)=home\_base;

A(2,:) = test(i,1:2);

t2 = line\_equation(A(1,1),A(1,2),A(2,1),A(2,2),1);

end

test = test(No\_Drones+1:end,:);

% disp(length(test));

t = t + t2;

end

function[t] = line\_equation(x1,y1,x2,y2,c)

v1 = 6;%(m/s) Speed of the hexapod

v2 = 10;%(m/s) Speed of the Quadcopter

if c==0

x =[x1,x2];

y =[y1,y2];

p1 =[x1,y1];

p2 =[x2,y2];

h = animatedline(x,y,'Color','b','LineWidth',3);

t = (norm(p1-p2,2)/v1);

plot(x2,y2,'bs');

txt1 = num2str(t);

text(x2,y2,txt1)

pause(0.1);

clearpoints(h);

else

x =[x1,x2];

y =[y1,y2];

X =[x2,x1];

Y =[y2,y1];

p1 =[x1,y1];

p2 =[x2,y2];

h1 = animatedline(x,y,'Color','r','LineWidth',3);

pause(0.1);

clearpoints(h1);

h2 = animatedline(x,y,'Color','g','LineWidth',3);

pause(0.1);

clearpoints(h2);

plot(x2,y2,'md');

t = ((2\*norm(p1-p2,2))/v2);

txt1 = num2str(t);

text(x2,y2,txt1)

pause(0.1);

end

end

% function[] = line\_equation(x1,y1,x2,y2)

% p1 =[x1,y1];

% p2=[x2,y2];

% a = ((p2(1,2)-p1(1,2))/(p2(1,1)-p1(1,1)));

% x = linspace(p1(1,1),p2(1,1),norm(p1-p2,2)\*10);

% y = (a\*(x-p1(1,1))) + p1(1,2);

% ax = gca;

% h = hgtransform('Parent',ax);

% hold on

% plot(x(1),y(1),'ks','Parent',h);

% hold off

% for k = 2:length(x)

% m = makehgtform('translate',x(k)-x(1),y(k)-y(1),0);

% h.Matrix = m;

% pause(0.01);

% end

% end