xin=input('enter the x coordinates in an order');

yin=input('enter the y coordinates in an order');

if(length(xin)~=length(yin))

error('i/p wrong');

end

xin(end+1)=xin(1);

yin(end+1)=yin(1);

[x1,y1]=perimeter\_points(xin,yin);

disp(x1);

disp(y1);

for j=1:length(x1)

a=double(x1(j));

b=double(y1(j));

path=[a -20;a b];

robotInitialLocation = path(1,:);

robotGoal = path(end,:);

initialOrientation = 0;

robotCurrentPose = [robotInitialLocation initialOrientation];

robot= differentialDriveKinematics("TrackWidth",1,"vehicleinputs","VehicleSpeedHeadingRate");

plot(path(:,1), path(:,2),'k--d');

controller = controllerPurePursuit;

controller.Waypoints = path;

controller.DesiredLinearVelocity = 0.6;

controller.MaxAngularVelocity = 2;

controller.LookaheadDistance = 0.3;

goalRadius = 0.1;

distanceToGoal = norm(robotInitialLocation - robotGoal);

sampleTime = 0.1;

vizRate = rateControl(1/sampleTime);

frameSize = robot.TrackWidth/0.8;

end

function [x1,y1]=perimeter\_points(xin,yin)

polyin = polyshape({xin},{yin});

x=xin(1);

y=yin(1);

for i=1:length(xin)-1

xdiff=(xin(i)-xin(i+1));

ydiff=(yin(i)-yin(i+1));

len=sqrt(xdiff^2 + ydiff^2);

disp(len);

[xi,yi]=filline([xin(i) yin(i)],[xin(i+1) yin(i+1)],abs(len)+1);

disp(xi);

disp(yi);

x=cat(2,x,xi);

y=cat(2,y,yi);

end

points=[x;y];

disp(points);

points=unique(points.','rows');

disp("the coordinates of all points on the polygon are");

x1=points(:,1);

y1=points(:,2);

disp(points);

figure;

plot(polyin)

hold on

plot(x1(1),y1(1),'r\*');

end

function [xx,yy]=filline(startp,endp,pts)

m=(endp(2)-startp(2))/(endp(1)-startp(1));

disp(m);

if (m==Inf||m==-inf)

xx(1:pts)=startp(1);

yy(1:pts)=linspace(startp(2),endp(2),pts);

elseif m==0

xx(1:pts)=linspace(startp(1),endp(1),pts);

yy(1:pts)=startp(2);

else

x=[startp(1),endp(1)];

y=[startp(2),endp(2)];

syms m c

eqs=[y(1)==m\*x(1)+c,y(2)==m\*x(2)+c];

[m1,c1]=solve(eqs,[m,c]);

xx=linspace(startp(1),endp(1),pts);

yy=m1\*(xx)+c1(1);

end

end