Assignment 3

ME766A

Sourit Saha, 200998

- **Code:**

generate_obstacles.m

```
function obstacle_space = generate_obstacles(sizex,sizey)
%function to define obstacles in the C-space. obstacle_space(x,y) = true,
%if an obstacle is present at that (x,y)
[x,y] = meshgrid(1:sizex,1:sizey);
obstacle_space = zeros(sizex,sizey);

'''p1 & c1''';
%obstacle_space(150:350,50:100)= 1;
%obstacle_space(75:150,100:200)= 1;
%obstacle_space(275:350,300:310)= 1;

%t = ((x - 250).^2 + (y - 200).^2) < 50^2;
%obstacle_space(t) = 1;

'''p2''';
%obstacle_space(250:300,275:325)= 1;
%obstacle_space(20:100,75:175)= 1;

%t = ((x - 300).^2 + (y - 100).^2) < 50^2;
%obstacle_space(t) = 1;

%t = ((x - 100).^2 + (y - 300).^2) < 50^2;
%obstacle_space(t) = 1;

'''p3''';
obstacle_space(300:350,25:75)= 1;
obstacle_space(20:100,75:175)= 1;

t = ((x - 250).^2 + (y - 300).^2) < 50^2;
obstacle_space(t) = 1;

t = ((x - 300).^2 + (y - 100).^2) < 50^2;
obstacle_space(t) = 1;

'''c2''';
%obstacle_space(300:350,250:300)= 1;
%obstacle_space(20:100,75:175)= 1;

%t = ((x - 300).^2 + (y - 220).^2) < 50^2;
%obstacle_space(t) = 1;

%t = ((x - 100).^2 + (y - 300).^2) < 50^2;
%obstacle_space(t) = 1;

end
```

PotentialMethod.m

```
function path = PotentialMethod(start,goal,potential,max_it)
[force_x, force_y] = gradient(-potential);
path = [start];
for i=1:max_it
    %start from last updated point
    current = path(end,:);
    if (abs(norm(current - goal)) < 5.0)
        disp('Robot in close proximity of goal');
        break;
    end
    ix = round(current(2));
    iy = round(current(1));
    delx = force_x(ix,iy);
    dely = force_y(ix,iy);
    delt = 1/ norm([delx,dely]);
    next_point = current + delt*[delx,dely];
    path = vertcat(path,next_point);
    %add new point to path
end
end
```

show_C_space.m

```
function show_C_space(obstacle_space,start,goal,sizex,sizey)
contourf(obstacle_space);
hold on;
plot(start(1),start(2),'g.', 'MarkerSize', 25);
plot(goal(1),goal(2),'r.', 'MarkerSize', 25);
hold off;
axis([0 sizex 0 sizey]);
axis xy;
axis on;
title('C-space');
xlabel('X');
ylabel('Y');
legend('C-Space','Start','Goal');
end
```

main.m

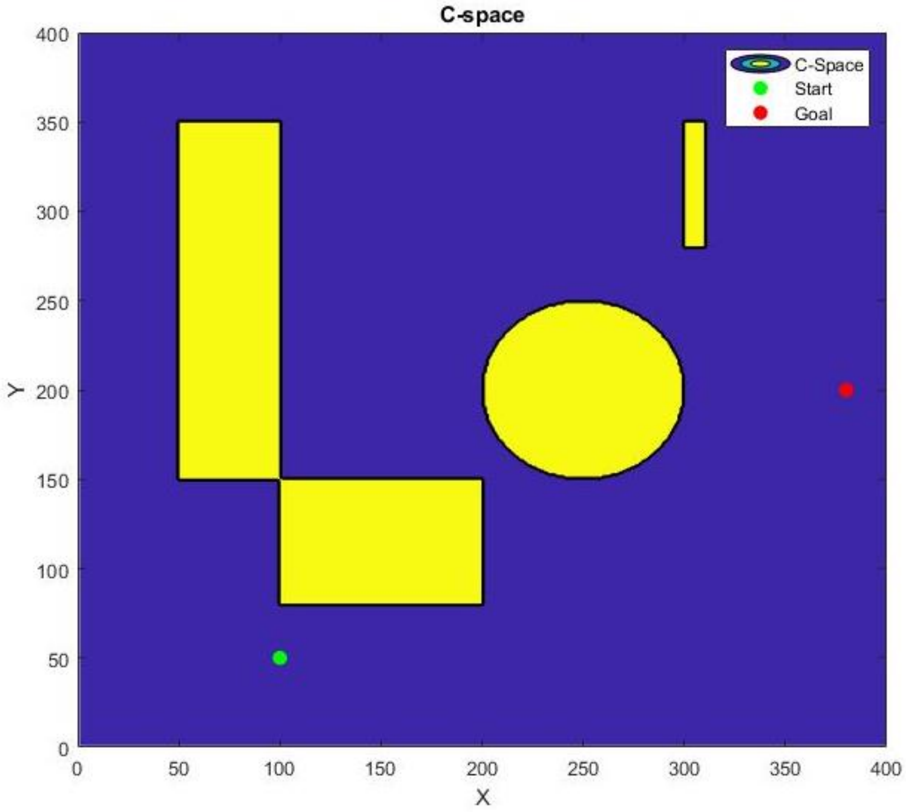
```
%define domain
sizex = 400;
sizey = 400;

[x,y] = meshgrid(1:sizey,1:sizex);

obstacle_space = generate_obstacles(sizex,sizey);

%to define attractive force we need start point and goal point

'''p1''';
%start = [100,50];
%goal = [380,200];

'''c1''';
%start = [20,80];
%goal = [380,200];

'''rest cases''';
start = [50,150];
goal = [380,350];

show_C_space(obstacle_space,start,goal,sizex,sizey);

%use function bwdist to get dist from current point to closest non zero
%point in domain

euc_d = bwdist(obstacle_space);

d_new = (euc_d/100);
d_new = d_new+1;

%base distance till which repulsive force acts
d0 = 2;

%parameter to control motion
p1 = 500;

rep = p1*((1./d_new - 1/d0).^2);

rep (d_new > d0) = 0;

%parameter to control attractive force

p2 = 1/400;

att = p2*((x-goal(1)).^2 + (y-goal(2)).^2);

%adding potentials

potential = att + rep;
path = PotentialMethod(start,goal,potential,1000);
```

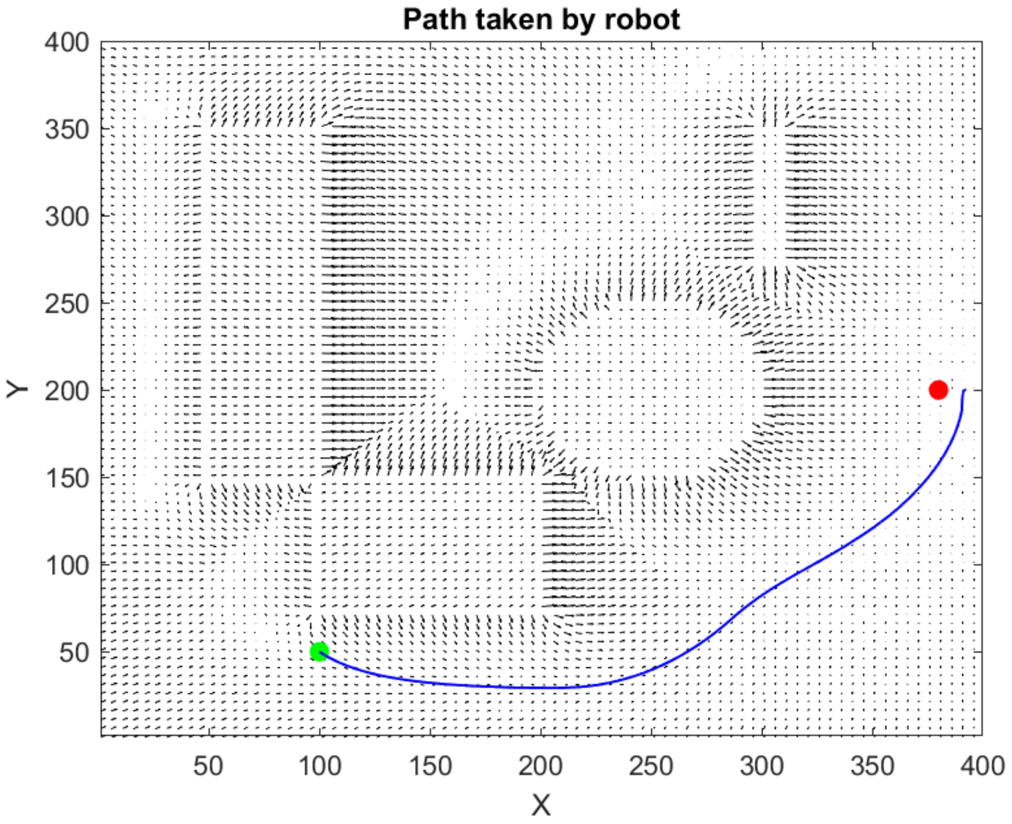
```
[force_x, force_y] = gradient (-potential);

figure;

xidx = 1:5:sizey;
yidx = 1:5:sizex;

quiver(x(yidx,xidx), y(yidx,xidx), force_x(yidx,xidx), force_y(yidx,xidx),1,'k');

axis ([1 sizey 1 sizex]);

hold on;

plot(start(1), start(2), 'g.', 'MarkerSize', 25);
plot(goal(1), goal(2), 'r.', 'MarkerSize', 25);
plot (path(:,1), path(:,2), 'b', 'LineWidth', 1);
title('Path taken by robot');
xlabel('X')
ylabel('Y')
```

- **Plots:**

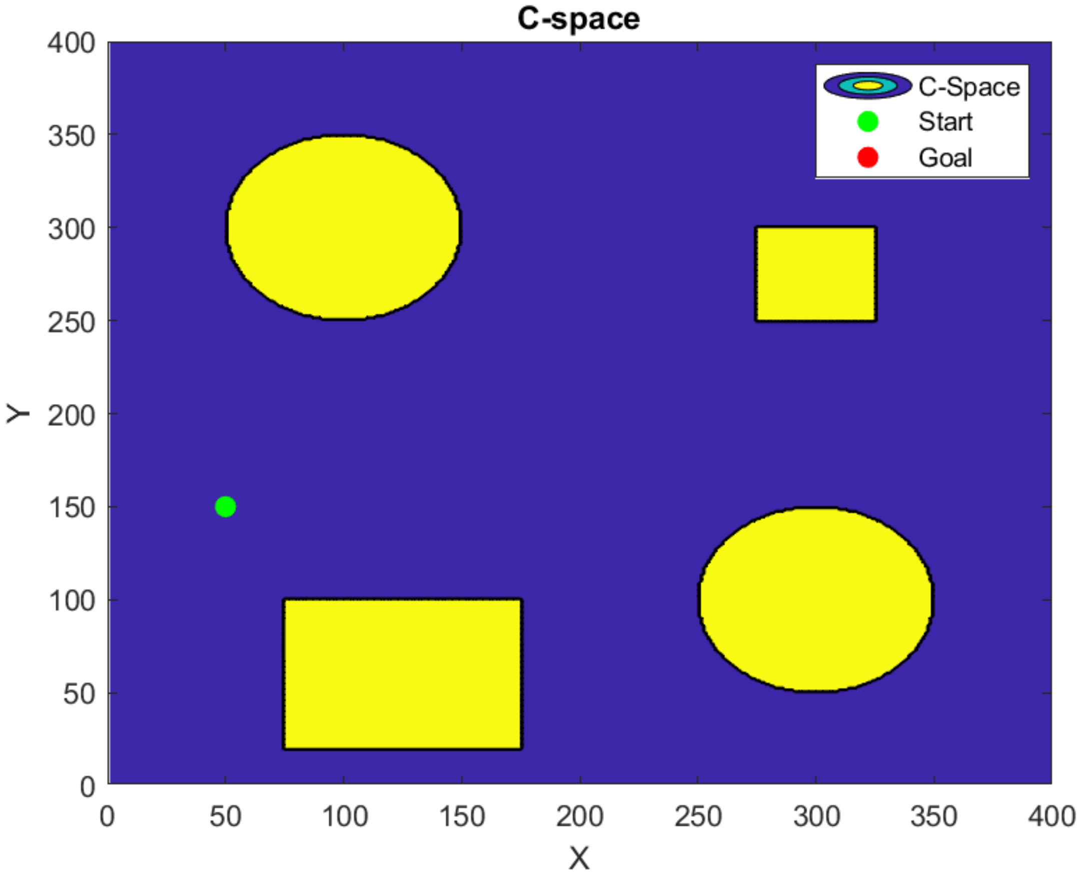
1. **Path Found Successfully:**

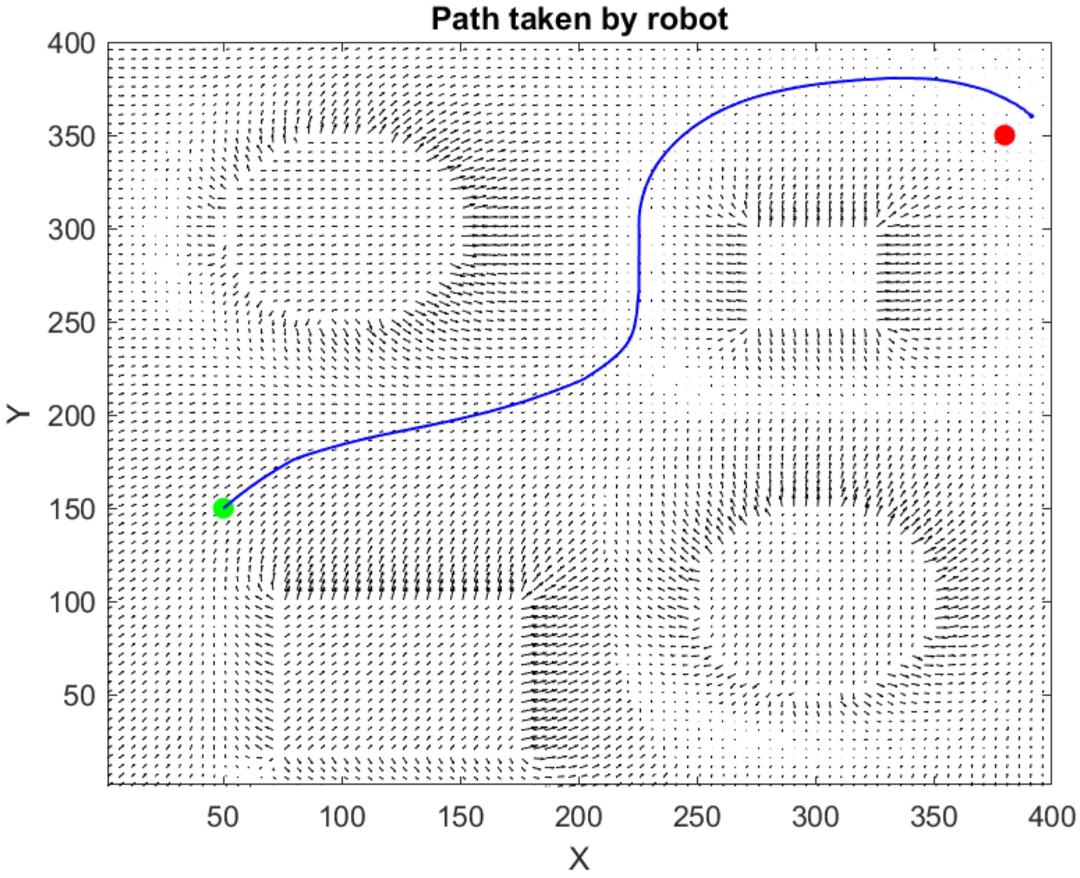
Case 1.



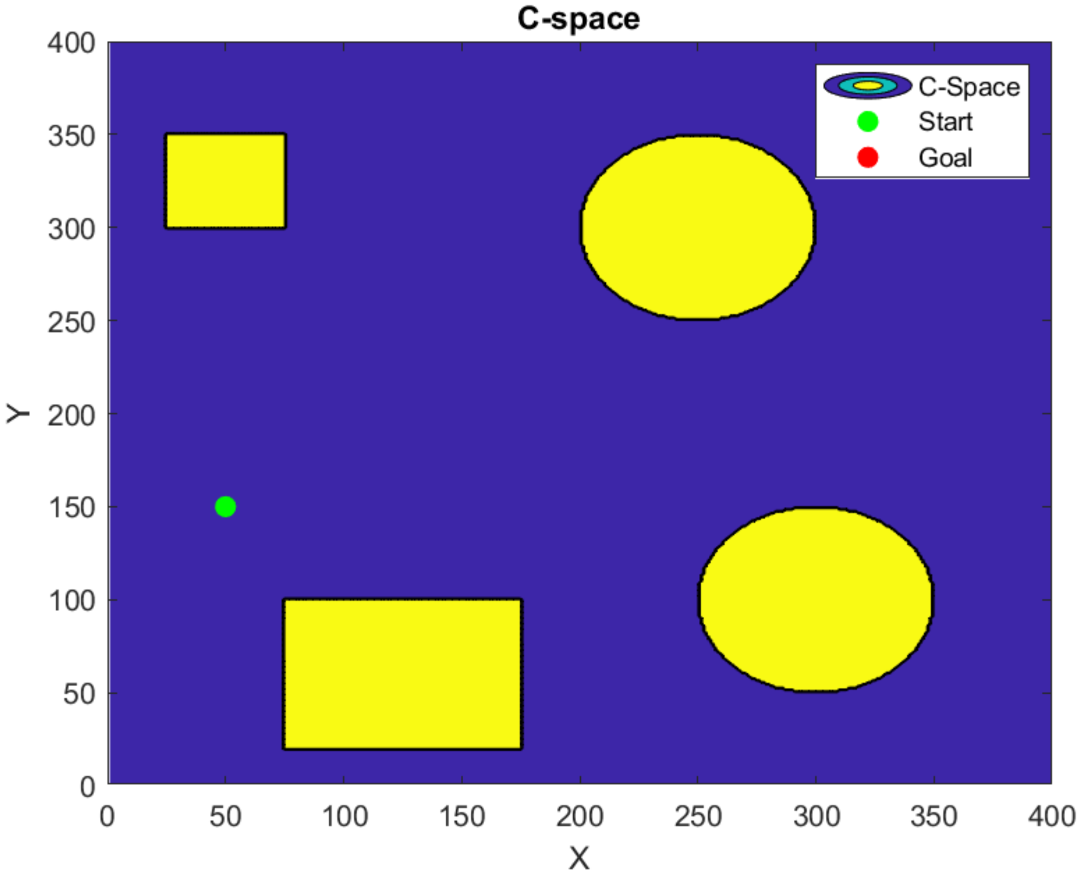


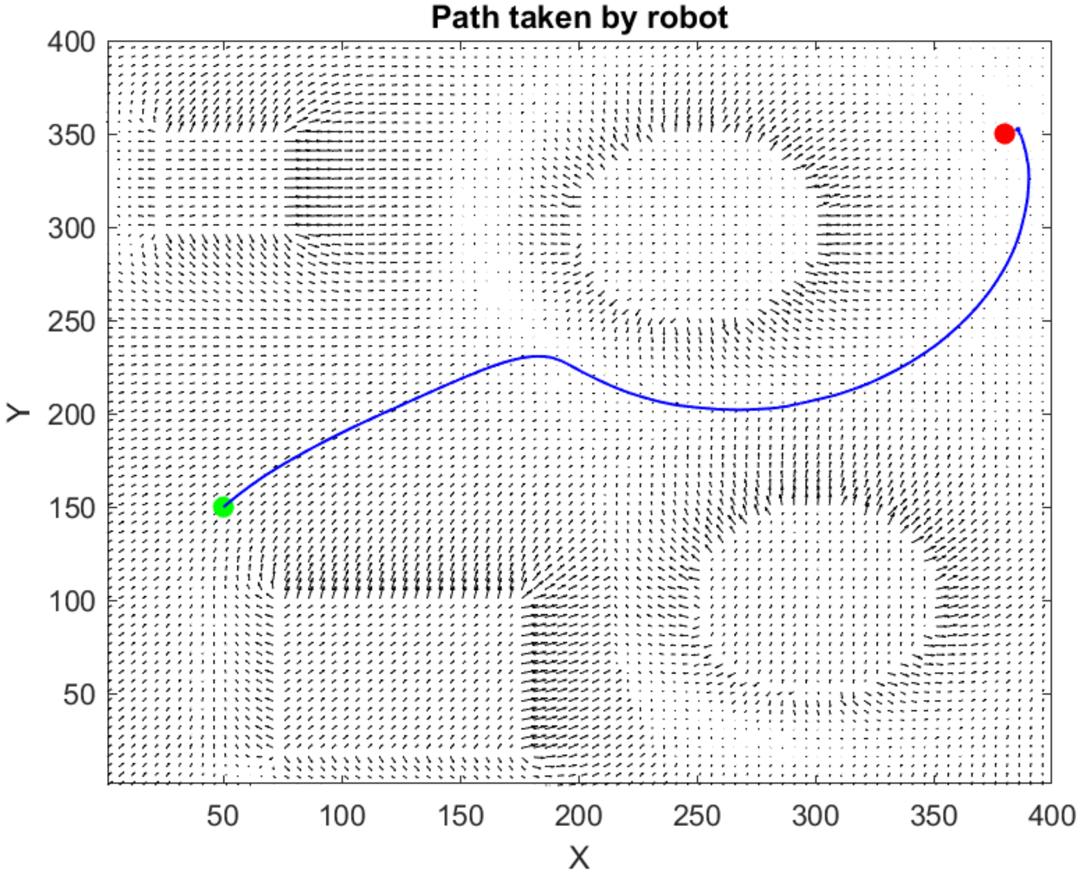
Case 2.


C-space
C-Space
Start
Goal
400
350
300
250
200
150
100
50
0
Y
0
50
100
150
200
250
300
350
400
X


Path taken by robot
400
350
300
250
200
150
100
50
Y
50
100
150
200
250
300
350
400
X

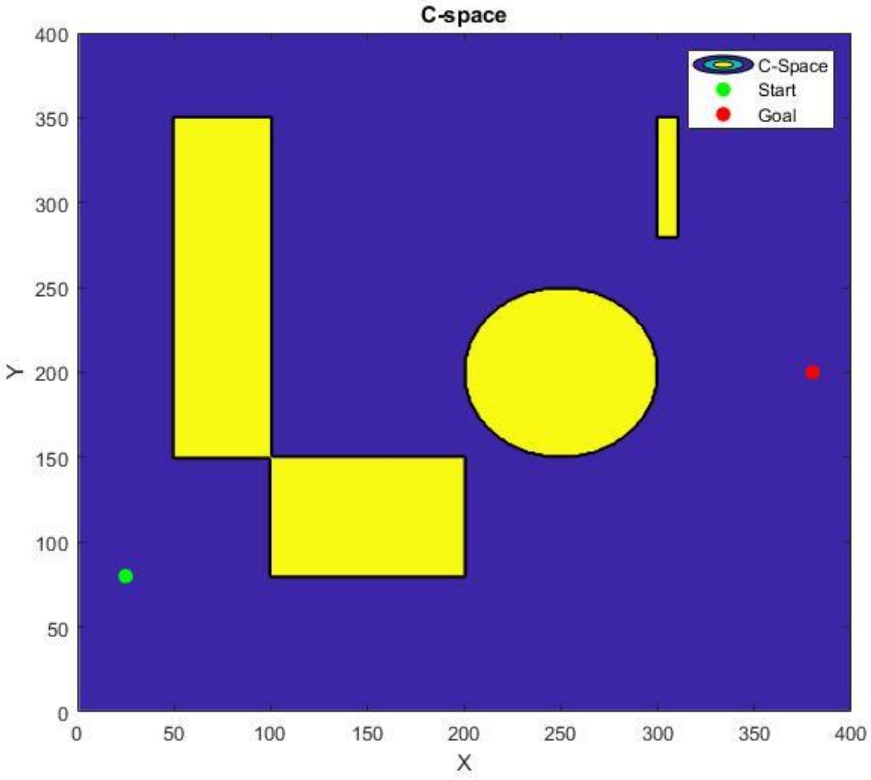
Case 3.

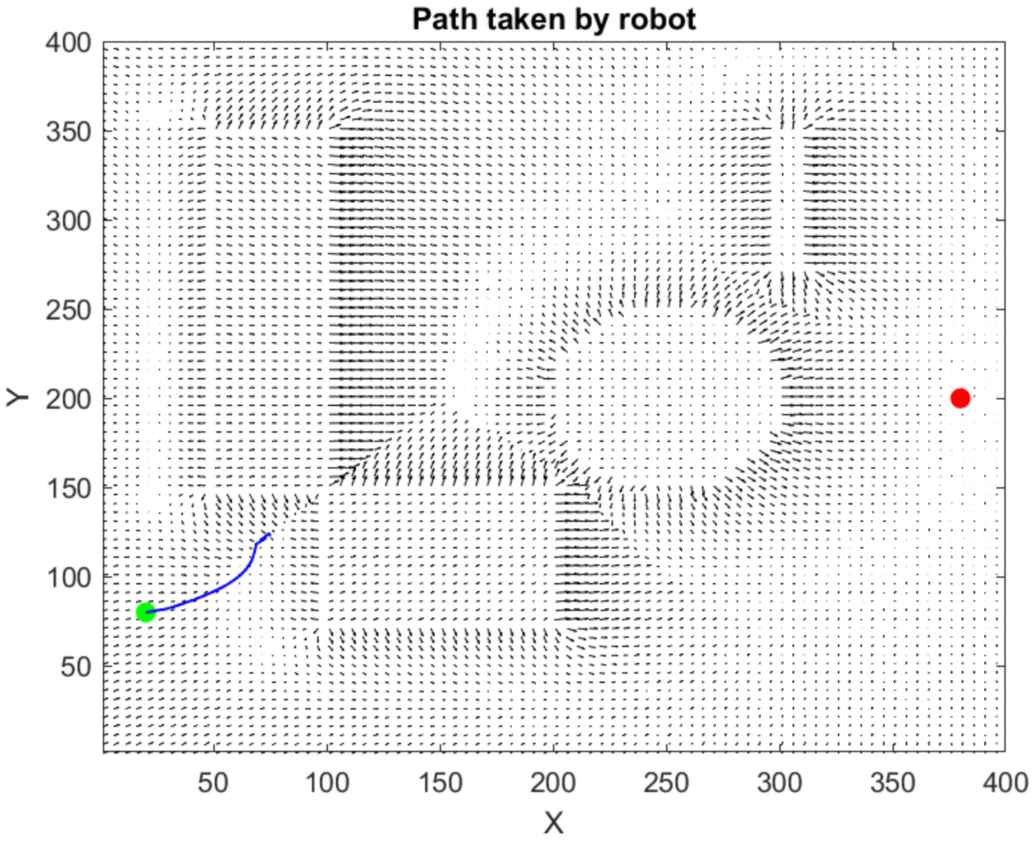

C-space
C-Space
Start
Goal
400
350
300
250
200
150
100
50
0
Y
0
50
100
150
200
250
300
350
400
X


Path taken by robot
400
350
300
250
200
150
100
50
Y
50
100
150
200
250
300
350
400
X

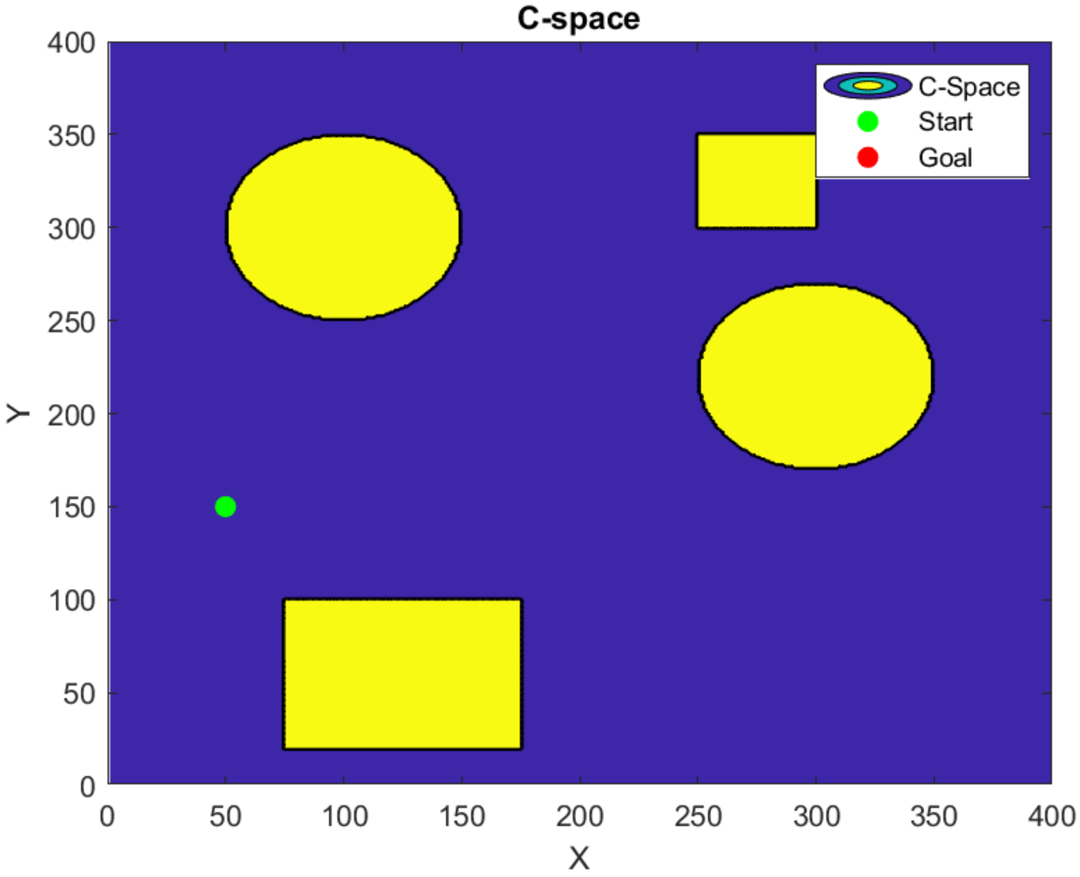
2. Failed Cases:

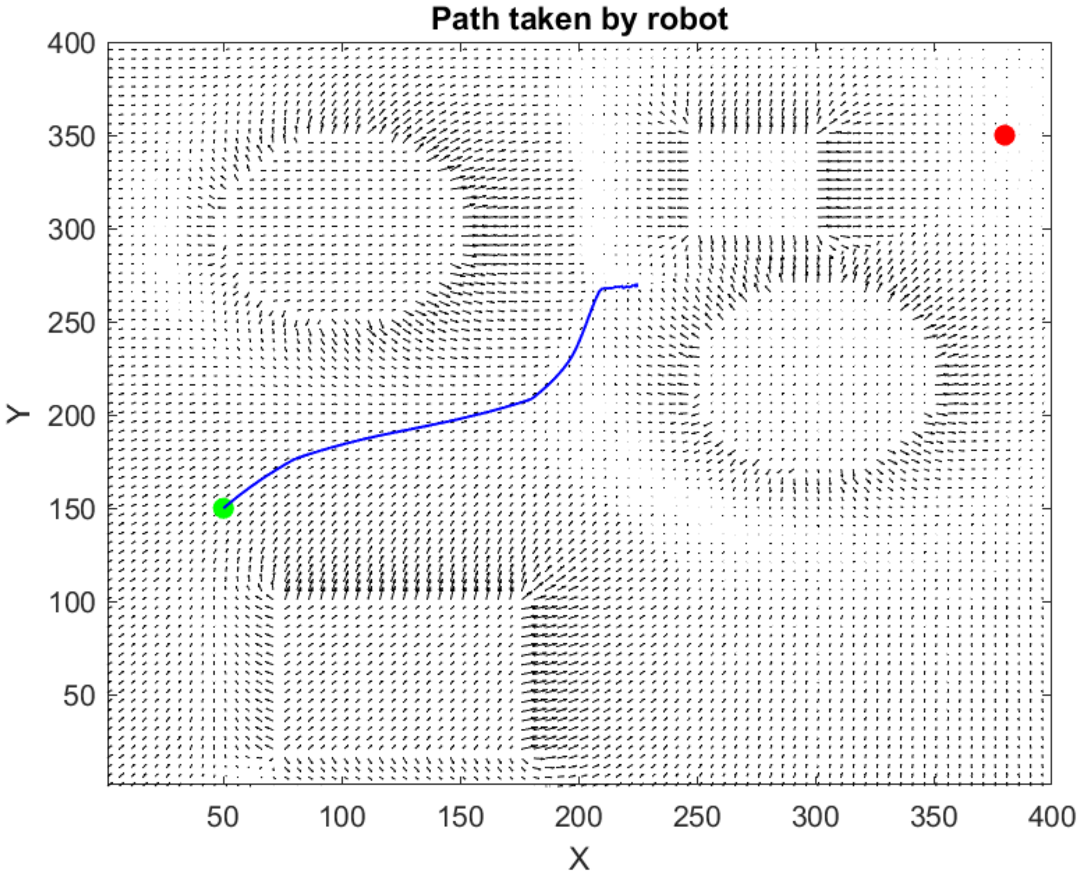
Case 1.





Case 2.


C-space
C-Space
Start
Goal
400
350
300
250
200
150
100
50
0
Y
0
50
100
150
200
250
300
350
400
X


Path taken by robot
400
350
300
250
200
150
100
50
Y
50
100
150
200
250
300
350
400
X