Term 081

EE 656: Robotics & Control

HW7

Potential Field Motion Planning

By

Mohammad Shahab

227598

For

Dr. Ahmad Masoud

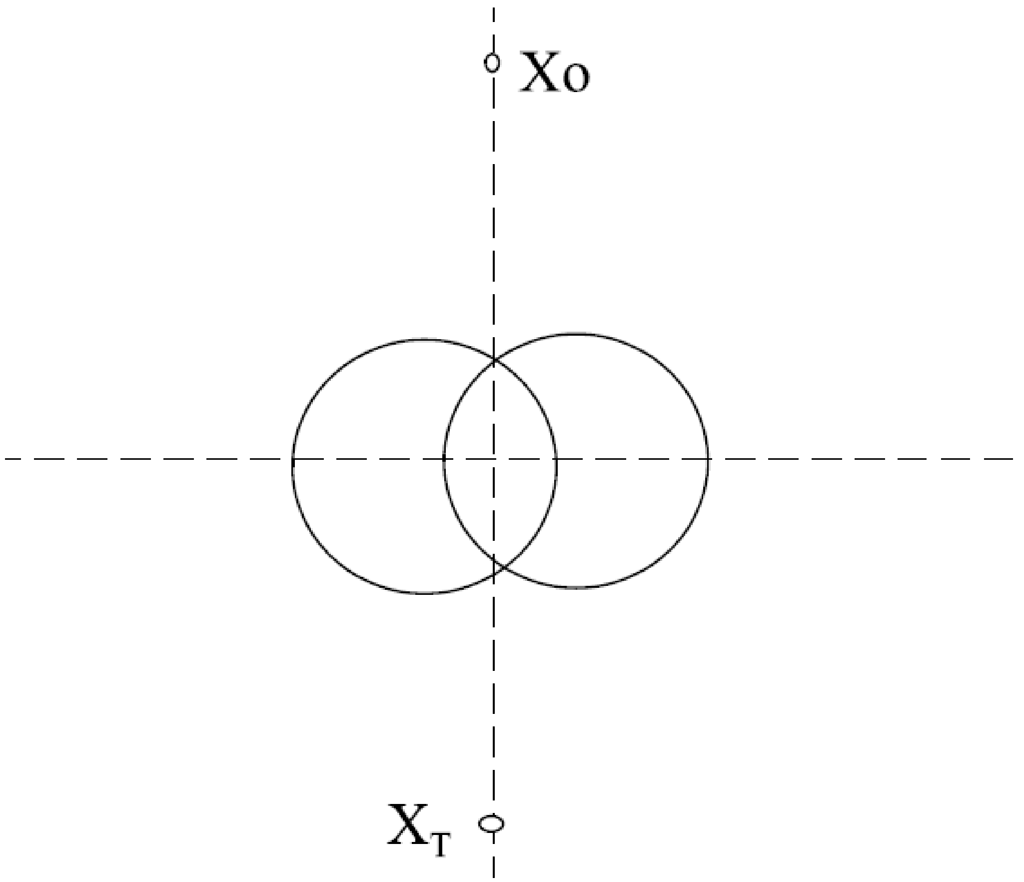
17 January 2009

Term 081

EE 656: Robotics & Control

HW7 – Potential Field Planning

We are to plan the motion for the environment in below figure



With initial position of

$$x_0 = (x_0, y_0) = (0, 4)$$

And final (target) position of

$$x_T = (x_T, y_T) = (0, -4)$$

The planning should take care of obstacles resembled by the two overlapping circles above. With radius = 2. The circles' centers are at (1, 0) and (-1, 0) respectively.

1) Develop the Artificial Potential Field

To develop the field of the environment of concern, we need to construct 3 different ‘surfaces’ to make the required planning. The three are reflecting the: starting position, final position and the obstacles.

a. Repelling Force @ start position

With the provided starting point, a repelling force should be generated at the initial position. We can construct the repelling ‘surface’ as function of x, y:

$$V_r(x, y) = \frac{K_r}{\sqrt{(x - x_0)^2 + (y - y_0)^2}}$$

With K_r a parameter controlling the shape of the ‘surface’. So, we can build the artificial forces at x, y as

$$f_{xr}(x, y) = -\frac{\partial V_r}{\partial x}$$

$$f_{yr}(x, y) = -\frac{\partial V_r}{\partial y}$$

b. Attracting Force @ target position

With the provided target point, an attracting force should be generated at the target position. We can construct the attracting ‘surface’ as function of x, y:

$$V_a(x, y) = \tfrac{1}{2}K_a[(x - x_T)^2 + (y - y_T)^2]$$

With K_a a parameter controlling the ‘shape’ of the ‘surface’. So, we can build the artificial forces at x, y as

$$f_{xa}(x, y) = -\frac{\partial V_a}{\partial x} = -K_a(x - x_T)$$

$$f_{ya}(x, y) = -\frac{\partial V_a}{\partial y} = -K_a(y - y_T)$$

c. Repelling Force around the Obstacle

With described obstacles, we generate forces at each point on border of the obstacle. However, it is enough to build the forces only on samples on the border. Specifically, we construct repelling forces at *i* points (x_{oi}, y_{oi}) with potential 'surfaces' of

$$V_{oi}(x,y) = \frac{K_o}{\sqrt{(x-x_{oi})^2 + (y-y_{oi})^2}}$$

With K_o a parameter controlling the 'shape' of the 'surface'. We construct the integrated 'surface' of

$$V_o(x,y) = \sum_i V_{oi}(x,y)$$

So, we can build the artificial forces at x, y as

$$f_{xo}(x,y) = -\frac{\partial V_o}{\partial x}$$

$$f_{yo}(x,y) = -\frac{\partial V_o}{\partial y}$$

d. Total

With described forces above we can construct the full force field at x, y as

$$f_x(x,y) = f_{xa}(x,y) + f_{xr}(x,y) - f_{yo}(x,y)$$

$$f_y(x,y) = f_{ya}(x,y) + f_{yr}(x,y) + f_{xo}(x,y)$$

With the 4th term resembling cycling (clockwise) force around the obstacle.

2) Generating the Potential Field

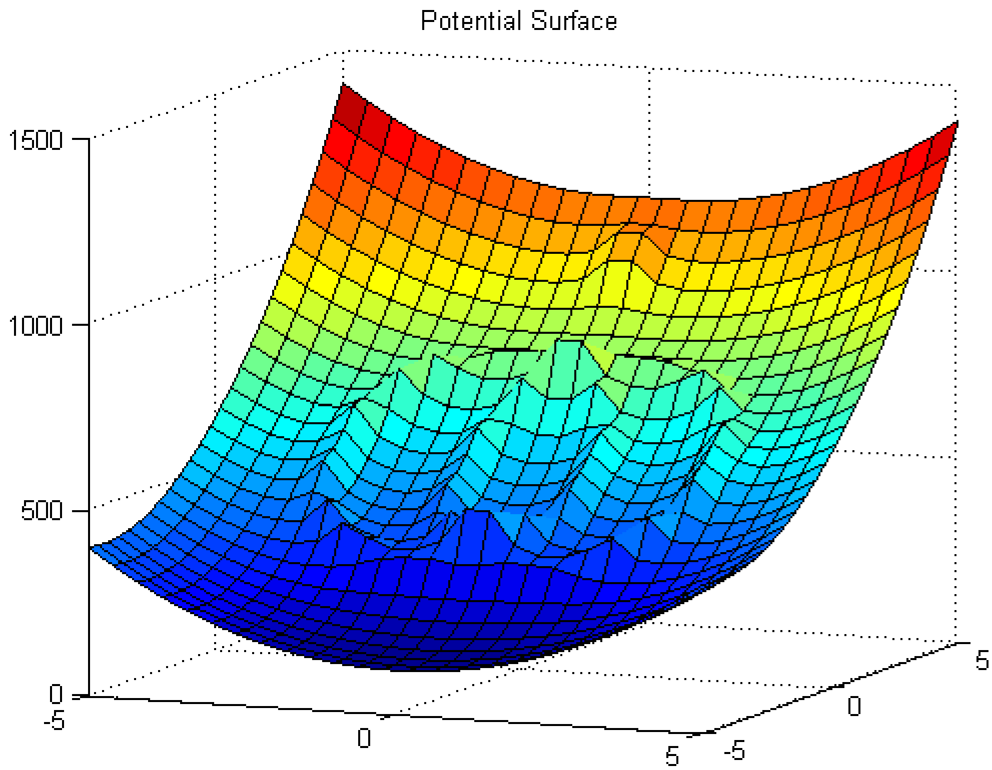
We can select the best values of parameters as

$$K_a = 25$$

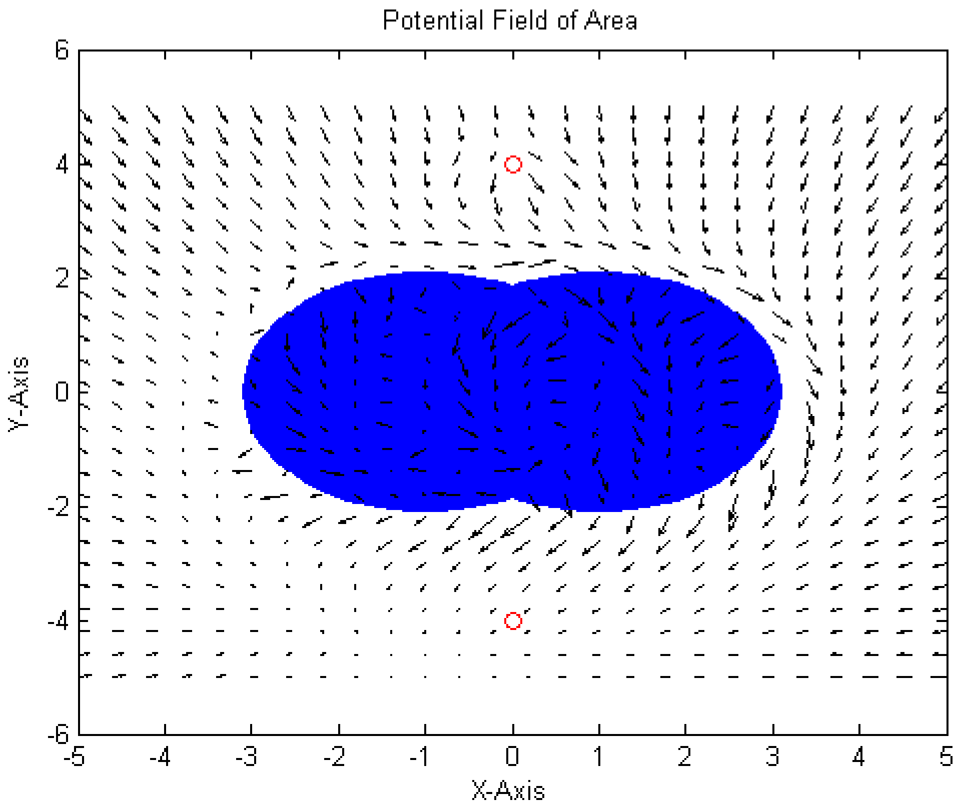
$$K_r = 30$$

$$K_o = 5$$

The potential surface is generated to be



We spacing of **0.4**, the gradient vector field of forces is



3) Motion Planning

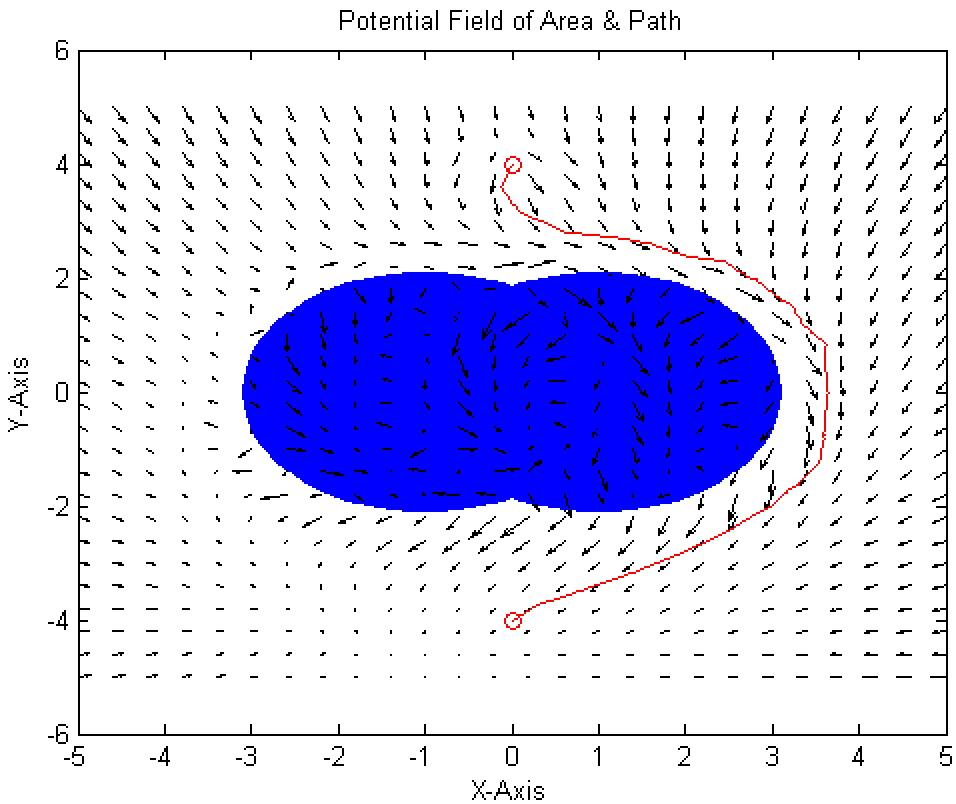
It is found that the best values are

$$K_a = 25$$

$$K_r = 30$$

$$K_o = 5$$

These values resulted in the motion of



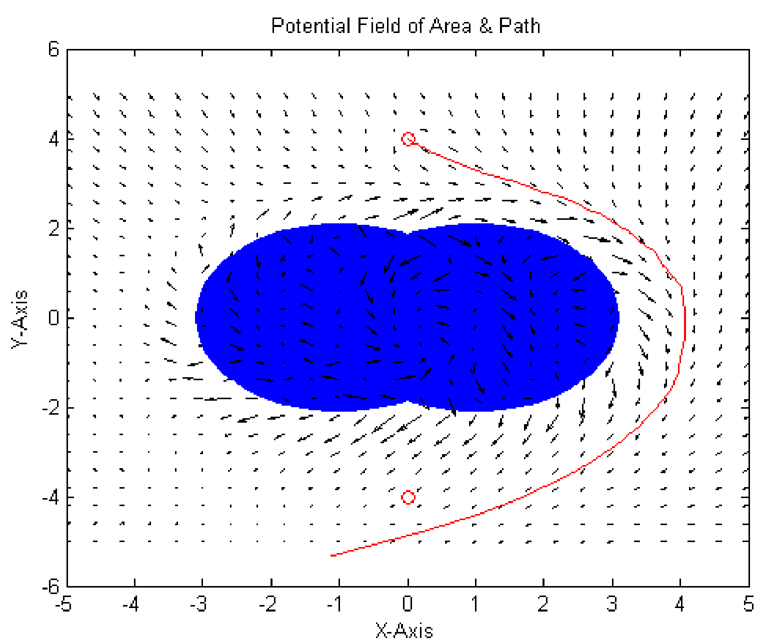
It is seen that tuning the forces are tricky in terms of convergence. It is tested that with values of

$$K_a = 25$$

$$K_r = 30$$

$$K_o = 10$$

The plan will **NOT CONVERGE** as seen below.



APPENDIX: MATLAB Program

```
clear all
clc

dh=.4;
X=-5:dh:5;
Y=-5:dh:5;

[X,Y]=meshgrid(X,Y);

x0=[0;4];
xT=[0;-4];

Kr=30;
Vr=Kr./sqrt((X-x0(1)).^2+(Y-x0(2)).^2);

[fxr,fyr]=gradient(Vr,dh,dh);


Ka=25;
Va=.5.*Ka.*((X-xT(1)).^2+(Y-xT(2)).^2);


[fxa,fya]=gradient(Va,dh,dh);

cx1=1;
cy1=0;
cx2=-1;
cy2=0;

rad=2+.1;

th=(0:.1*dh:2).*pi;

xo=rad.*cos(th);
yo=rad.*sin(th);

xo1=xo-cx1;
yo1=yo-cy1;
xo2=xo-cx2;
yo2=yo-cy2;

Vo1=[];
Vo2=[];
Vo=zeros(size(Vr));

Ko=5;

for i=1:length(th)
    Vo1=Ko./sqrt((X-xo1(i)).^2+(Y-yo1(i)).^2);
    Vo2=Ko./sqrt((X-xo2(i)).^2+(Y-yo2(i)).^2);

    Vo=Vo+Vo1+Vo2;


end

[fxo,fyo]=gradient(Vo,dh,dh);

fX=-fxr-fxa-fxo-fyo;
fY=-fyr-fya-fyo+fxo;
```

```
quiver(X,Y,fX,fY,'k')
title('Potential Field of Area')
xlabel('X-Axis')
ylabel('Y-Axis')
hold on
plot(x0(1),x0(2),'ro',xT(1),xT(2),'ro')
fill(xo1,yo1,'b',xo2,yo2,'b')
plot(xo1,yo1,'b',xo2,yo2,'b')
hold off


ss=1;
k=1;
xp=[];
yp=[];
xp(1)=x0(1);
yp(1)=x0(2);
ix=[];
iy=[];
jx=[];
jy=[];

fxx=0;
fyy=0;

while ss

    Pw=sqrt(((X-xp(k)).^2)+((Y-yp(k)).^2));
    xw(k)=min(min(Pw));

    [iix,iiy]=find(Pw==xw(k));

         ix(k)=iix(1);
         iy(k)=iiy(1);


         fx1=fX(ix(k),iy(k));
         fy1=fY(ix(k),iy(k));

    fxx(k)=fx1./norm(fX);
    fyy(k)=fy1./norm(fY);


    ff(k,:)=[fX(ix(k),iy(k)),fY(ix(k),iy(k))];

    xp(k+1)=xp(k)+dh*(fxx(k));
    yp(k+1)=yp(k)+dh*(fyy(k));


    if (sqrt((xp(k+1)-xT(1)).^2+(yp(k+1)-xT(2)).^2)<=0.4)
        ss=0;
    end

    k=k+1;
end

hold
plot([xp],[yp],'r')

figure
surf(X,Y,(Vr)+(Va)+(Vo))
title('Potential Surface')
```