AE 225

KRISHNA WADHWANI- 160010031

NOVEMBER 1, 2017

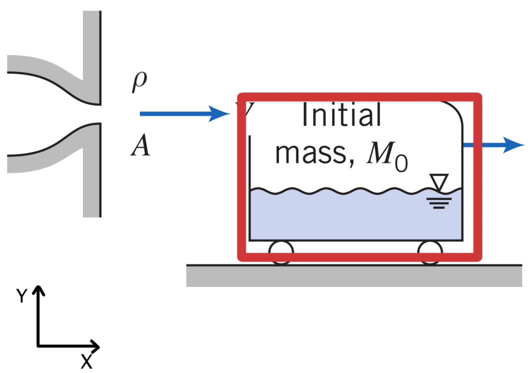
1 QUESTION 1

Assumptions:

- Liquid is incompressible
- Flow of liquid jet is 1-D at inlet
- No fluid motion within the control volume relative to it

<u>Step 1: Choosing Control Volume</u>

I am taking the tank as my control-volume (highlighted in red in the figure below). The selected control volume is moving with an accelerated velocity. At a give time t, it's velocity is $\underline{U}$. Mass of the system at any time t is M. Liquid jet is entering the tank with velocity $V\ \hat{\underline{i}}$.



<u>Step 2: Mass Conservation</u>

From Reynold Transport Theorem for mass consevation: $\frac{dm_{sys}}{dt} = \frac{\partial}{\partial t}\int\limits_{CV} \rho d\forall + \int\limits_{CS} \rho(\underline{W}.\hat{n})dA$

As mass of the system is conserved: $\frac{dm_{sys}}{dt} = 0$

$\implies \dfrac{dM}{dt} + \int\limits_{in} \rho(\underline{W}.\hat{n})dA + \int\limits_{out} \rho(\underline{W}.\hat{n})dA = 0$

$\int\limits_{out} \rho(\underline{W}.\hat{n})dA = 0$ as air is deflected into the tank and not exiting it

$\int\limits_{in} \rho(\underline{W}.\hat{n})dA = -\rho A(V-U)$ as flow is 1-D

So, $\dfrac{dM}{dt} - \rho A(V-U) = 0 \quad \ldots eq^n 1$

Step 3: Momentum Conservation

From Reynold Transport Theorem for momentum consevation: $\underline{F} = \frac{\partial}{\partial t}\int\limits_{CV}\rho\underline{V}d\cancel{V} + \int\limits_{CS}\rho(\underline{W}.\hat{n})\underline{V}dA$

Given negligible resistance, there is no net force on the control volume. So, what we get is(Assuming negligible flow of liquid inside the tank relative to the tank):

$$\frac{d(MU)}{dt} + \int\limits_{in}\rho(\underline{W}.\hat{n})\underline{V}dA = 0$$

$$\implies M\frac{dU}{dt} + U\frac{dM}{dt} - \rho(V-U)VA = 0$$

Using $eq^n 1$, we get:

$$\frac{dU}{dt} = \frac{\rho A(V-U)^2}{M} \qquad \ldots eq^n 2$$

Now using $eq^n 2$ for solving $eq^n 1$:

$$dM = \rho A(V-U)\frac{dU}{dU/dt}$$

$$\implies \int\limits_{M_0}^{M}\frac{dM}{M} = \int\limits_{0}^{U}\frac{dU}{V-U}$$

$$\implies ln\frac{M}{M_0} = -ln\frac{|V-U|}{V}$$

$$\implies \mathbf{M} = \frac{\mathbf{M_0}}{\mathbf{V}-\mathbf{U}}$$

Using the value of M, we get:

$$\frac{dU}{dt} = \frac{\rho A(V-U)^3}{M_0 V}$$

$$\implies \int\limits_{0}^{U}\frac{dU}{(V-U)^3} = \int\limits_{0}^{t}\frac{\rho A dt}{M_0 V}$$

$$\implies \frac{1}{2}\Big[\frac{1}{(V-U)^2} - \frac{1}{V^2}\Big] = \frac{\rho A t}{M_0 V}$$

$$\implies \frac{1}{\left(1-\frac{U}{V}\right)^2} = \frac{2\rho AVt}{M_0} + \frac{1}{v^2}$$

$$\implies \frac{\mathbf{U}}{\mathbf{V}} = \mathbf{1} - \frac{\mathbf{1}}{\left(\mathbf{1} + \frac{\mathbf{2\rho AVt}}{\mathbf{M_0}}\right)^{\mathbf{1/2}}}$$

2 QUESTION 2

Assumptions:

- Air is incompressible at standard condition
- Flow is steady
- Flow is 1-D

To calculate pressure at inlet, I am using Bernoulli's equation at a point on the atmosphere and a point on inlet at the same streamline:

$$\frac{P}{\rho} + \frac{v^2}{2} + gz = c$$

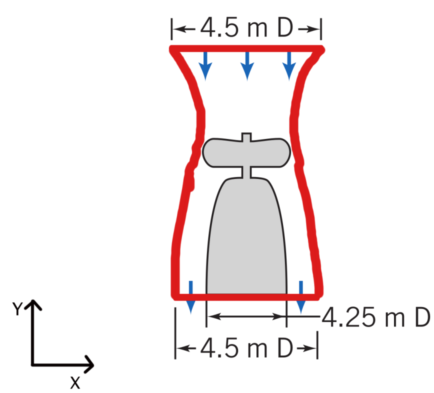
Assuming velocity of the air at atmosphere to be zero and taking the two points at same height:

$$P_{atm} = P_1 + \frac{\rho V_1^2}{2}$$

$$\implies P_{in,gauge} = -\frac{\rho V_1^2}{2}$$

Step 1: Choosing control volume

The chosen control volume is as follows (highlighted in red in the figure below). Chosen control volume is perpendicular to the flow. Air is entering the control volume with a velocity $-V_{in}\ \underline{\hat{j}}$ and exiting the control volume with velocity $-V_{out}\ \underline{\hat{j}}$



Step 2: Mass Conservation

From Reynold Transport Theorem for mass consevation: $\frac{dm_{sys}}{dt} = \frac{\partial}{\partial t}\int\limits_{CV} \rho d\forall + \int\limits_{CS} \rho(\underline{W}.\hat{n})dA$

As mass of the system is conserved: $\frac{dm_{sys}}{dt} = 0$

As flow is steady: $\dfrac{\partial}{\partial t}\int\limits_{CV} \rho d\forall = 0$

As flow is 1D and incompressible: $\int\limits_{CS} \rho(\underline{W}.\hat{n})dA = \rho A_{out}V_{out} - \rho A_{in}V_{in}$

$$\implies A_{out}V_{out} = A_{in}V_{in} \quad \ldots eq^n 1$$

Step 3: Momentum Conservation

Momentum consevation along y:

$$\underline{F} = \frac{\partial}{\partial t}\int\limits_{CV} \rho\underline{V}d\forall + \int\limits_{CS} \rho(\underline{W}.\hat{n})\underline{V}dA$$

As flow is steady: $\frac{\partial}{\partial t}\int\limits_{CV} \rho\underline{V}d\forall = 0$

As flow is 1-D and incompressible: $\int\limits_{CS} \rho(\underline{W}.\hat{n})\underline{V}dA = \rho A_{in}V_{in}^2 - \rho A_{out}V_{out}^2$

Net force on the control volume is equal to the weight of the helicopter $= -mg - P_{in,gauge}A_{in} + P_{out,gauge}A_{out}$

$\implies -mg - (-\frac{\rho V^2}{2})A_{in} + 0 = \rho A_{in}V_{in}^2 - \rho A_{out}V_{out}^2$

From $eq^n 1$:

$\implies -mg + \frac{\rho V_{in}^2}{2}A_{in} = \rho A_{in}V_{in}^2(1 - \frac{A_{in}}{A_{out}})$

$\implies V_{in}^2 = -\frac{mg}{-\frac{\rho A_{in}}{2} + \rho A_{in}(1 - \frac{A_{in}}{A_{out}})}$

Given:

$A_{in} = \frac{\pi(4.5)^2}{4} = 15.904m^2$

$A_{out} = \frac{\pi[(4.5)^2 - (4.25)^2]}{4} = 1.718m^2$

$m = 1000kg$

Taking: $g = 9.81m/s^2, \ \ \rho = 1.225kg/m^3$

Using these values in the above expression, we get: $V_{in} = 7.58 \ m/s^2$

So, we get speed of the air leaving the craft V_{out} as:

$$V_{out} = \frac{A_{in}V_{in}}{A_{out}}$$

$\implies \mathbf{V_{out} = 70.2 \ m/s^2}$

Step 4: Energy Conservation

For the minimum power(denoted by $\dot{\mathscr{W}}_{shaft,net,in}$) delivered by the propeller, we will use energy conservation equation as:

$$\frac{d}{dt}\int_{CV} e\rho \, d\mathcal{V} + \int_{CS}(\breve{u} + \frac{p}{\rho} + \frac{V^2}{2} + gz)\rho(W.\hat{n})dA = \dot{\mathscr{W}}_{shaft,net,in} + \dot{\mathscr{Q}}_{net,in}$$

Applying assumptions of 1D inlet/outlet, steady, incompressibility and no heat input, we get:

$$\rho A_{in}V_{in}(\breve{u}_{out} - \breve{u}_{in} + \frac{V_{out}^2 - V_{in}^2}{2} + g\Delta z + \frac{P_{out} - P_{in}}{\rho}) = \dot{\mathscr{W}}_{shaft,net,in}$$

Using equation of state:$P = \rho RT$: $T_1 = 288.1K$ and $T_2 = 288.0026K$. As change in temperature is small, we can neglect changes in internal energy as internal energy is a function of temperature.

$P_{out} - P_{in} = \frac{\rho V_1^2}{2}$

Ignoring changes in gravitational potential energy:

$\implies \dot{\mathscr{W}}_{shaft,net,in} = \rho A_{out}\frac{V_{out}^3}{2}$

Placing the values, We get:

$\dot{\mathscr{W}}_{\mathbf{shaft,net,in}} = \mathbf{363.89 \ kW}$

3 QUESTION 3

Incompressible Navier-Stoke's equations:

Mass conservation: $\nabla.\underline{u} = 0$
Momentum Conservation: $\rho\frac{\partial u}{\partial t} + \rho(\underline{u}.\nabla)\underline{u}) = -\nabla p + \underline{f}_{body} + \mu\nabla^2\underline{u}$

In cylindrical coordinates:

Mass conservation: $\frac{1}{r}\frac{\partial(ru_r)}{\partial r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial u_z}{\partial z} = 0$
Momentum conservation:
Along r: $\rho(\frac{\partial u_r}{\partial t} + u_r\frac{\partial u_r}{\partial r} + \frac{u_\theta}{r}\frac{\partial u_r}{\partial \theta} - \frac{u_\theta^2}{r} + u_z\frac{\partial u_r}{\partial z}) = -\frac{\partial P}{\partial r} + \rho g_r + \mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_r}{\partial r}) - \frac{u_r}{r^2} + \frac{1}{r^2}\frac{\partial^2 u_r}{\partial \theta^2} - \frac{2}{r^2}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial^2 u_r}{\partial z^2}]$

Along θ: $\rho(\frac{\partial u_\theta}{\partial t} + u_r\frac{\partial u_\theta}{\partial r} + \frac{u_\theta}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{u_\theta u_r}{r} + u_z\frac{\partial u_\theta}{\partial z}) = -\frac{1}{r}\frac{\partial P}{\partial \theta} + \rho g_\theta + \mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_\theta}{\partial r}) - \frac{u_\theta}{r^2} + \frac{1}{r^2}\frac{\partial^2 u_\theta}{\partial \theta^2} + \frac{2}{r^2}\frac{\partial u_r}{\partial \theta} + \frac{\partial^2 u_\theta}{\partial z^2}]$

Along z: $\rho(\frac{\partial u_z}{\partial t} + u_r\frac{\partial u_z}{\partial r} + \frac{u_\theta}{r}\frac{\partial u_z}{\partial \theta} + u_z\frac{\partial u_z}{\partial z}) = -\frac{\partial P}{\partial z} + \rho g_z + \mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_z}{\partial r}) + \frac{1}{r^2}\frac{\partial^2 u_z}{\partial \theta^2} + \frac{\partial^2 u_z}{\partial z^2}]$

Assumptions to obtain exact solution for cylinderical coulette flow:

- Laminar
- Incompressible
- Steady
- Rotationally Symmetric ($\approx$ Fully developed)
- Simple Geometry
- Ignoring gravity

From the above conditions:

$$\frac{\partial P}{\partial \theta} = 0, \frac{\partial P}{\partial z} = 0$$

Although rotationally assymetric solution also exists, we concern our self with rotationally symmetric solution for our problem. As flow is rotationally symmetric and there is no flow in axial direction, velocity will depend only on r and $u_z = 0$. So, our velocity is of the form:
$\underline{u} = u_r(r)\hat{e}_r + u_\theta(r)\hat{e}_\theta$

Applying mass conservation:
$\nabla.(\underline{u}) = 0$
$\implies \frac{1}{r}\frac{\partial(ru_r)}{\partial r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial u_z}{\partial z} = 0$
Now,
$\frac{\partial u_\theta}{\partial \theta} = 0$ as u_θ is a function of r due to rotational symmetry
$\frac{\partial u_z}{\partial z} = 0$ as u_z due to no flow in axial direction
$\implies \frac{1}{r}\frac{d(ru_r)}{dr} = 0$
Integrating with respect to r:
$ru_r = k$

As $u_r = 0$ at $r = R_1$ and $r = R_2$,
$u_r \equiv 0$
<u>Momentum conservation along r:</u>

$$\rho(\frac{\partial u_r}{\partial t}+u_r\frac{\partial u_r}{\partial r}+\frac{u_\theta}{r}\frac{\partial u_r}{\partial \theta}-\frac{u_\theta^2}{r}+u_z\frac{\partial u_r}{\partial z}) = -\frac{\partial P}{\partial r}+\rho g_r+\mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_r}{\partial r})-\frac{u_r}{r^2}+\frac{1}{r^2}\frac{\partial^2 u_r}{\partial \theta^2}-\frac{2}{r^2}\frac{\partial u_\theta}{\partial \theta}+\frac{\partial^2 u_r}{\partial z^2}]$$

$\rho\frac{\partial u_r}{\partial t} = 0$ as flow is steady
$u_r\frac{\partial u_r}{\partial r} = 0$ as $u_r = 0$
$\frac{u_\theta}{r}\frac{\partial u_r}{\partial \theta} = 0$ as $u_r = u_r(r) = 0$
$u_z\frac{\partial u_r}{\partial z} = 0$ as $u_r = u_r(r) = 0$
$\rho g_r = 0$ as g is along z
$\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_r}{\partial r})$ as $u_r = 0$
$\frac{u_r}{r^2} = 0$ as $u_r = 0$
$\frac{\partial^2 u_r}{\partial \theta^2} = 0$ as $u_r = u_r(r) = 0$
$\frac{\partial u_\theta}{\partial \theta} = 0$ as $u_\theta = u_\theta(r)$
$\frac{\partial^2 u_r}{\partial z^2} = 0$ as there is no flow in radial direction

So, we get:

$$\rho\frac{u_\theta}{r^2} = -\frac{\partial P}{\partial r}$$

<u>Momentum conservation along z:</u>

$$\rho(\frac{\partial u_z}{\partial t} + u_r\frac{\partial u_z}{\partial r} + \frac{u_\theta}{r}\frac{\partial u_z}{\partial \theta} + u_z\frac{\partial u_z}{\partial z}) = -\frac{\partial P}{\partial z} + \rho g_z + \mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_z}{\partial r}) + \frac{1}{r^2}\frac{\partial^2 u_z}{\partial \theta^2} + \frac{\partial^2 u_z}{\partial z^2}]$$

$\rho\frac{\partial u_z}{\partial t} = 0$ as flow is steady
$u_r\frac{\partial u_z}{\partial r} = \frac{u_\theta}{r}\frac{\partial u_z}{\partial \theta} = u_z\frac{\partial u_z}{\partial z}) = \frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_z}{\partial r}) = \frac{1}{r^2}\frac{\partial^2 u_z}{\partial \theta^2} = \frac{\partial^2 u_z}{\partial z^2} = 0$ as $u_z = 0$ (No flow in z direction)
$\frac{\partial P}{\partial z} = 0$ as give that there is no flow in axial direction
$\rho g_z = 0$ as we have ignored gravity

<u>Momentum conservation along θ:</u>

$$\rho(\frac{\partial u_\theta}{\partial t}+u_r\frac{\partial u_\theta}{\partial r}+\frac{u_\theta}{r}\frac{\partial u_\theta}{\partial \theta}+\frac{u_\theta u_r}{r}+u_z\frac{\partial u_\theta}{\partial z}) = -\frac{1}{r}\frac{\partial P}{\partial \theta}+\rho g_\theta+\mu[\frac{1}{r}\frac{\partial}{\partial r}(r\frac{\partial u_\theta}{\partial r})-\frac{u_\theta}{r^2}+\frac{1}{r^2}\frac{\partial^2 u_\theta}{\partial \theta^2}+\frac{2}{r^2}\frac{\partial u_r}{\partial \theta}+\frac{\partial^2 u_\theta}{\partial z^2}]$$

$\rho\frac{\partial u_\theta}{\partial t} = 0$ as flow is steady
$u_r\frac{\partial u_\theta}{\partial r} = \frac{u_\theta u_r}{r} = \frac{2}{r^2}\frac{\partial u_r}{\partial \theta} = 0$ as $u_r = u_r(r) = 0$
$\frac{u_\theta}{r}\frac{\partial u_\theta}{\partial \theta} = u_z\frac{\partial u_\theta}{\partial z}) = \frac{1}{r^2}\frac{\partial^2 u_\theta}{\partial \theta^2} = \frac{\partial^2 u_\theta}{\partial z^2} = 0$ as $u_\theta = u_\theta(r)$
$\rho g_\theta = 0$ as g is along z

So, we get :
$$\frac{\partial}{\partial r}\Big(\frac{1}{r}(\frac{\partial(ru_\theta)}{\partial r})\Big) = 0$$
As u_θ is a function of r:
$$\frac{d}{dr}\Big(\frac{1}{r}(\frac{d(ru_\theta)}{dr})\Big) = 0$$

Integrating with respect to r:

$\dfrac{1}{r}\dfrac{d(ru_\theta)}{dr} = c_1$

$\implies u_\theta(r) = \dfrac{c_1 r}{2} + \dfrac{c_2}{r}$

Applying boundary conditions(No-slip condition):

$u_\theta = \omega_1 R_1$ for $r = R_1$

$u_\theta = \omega_2 R_2$ for $r = R_2$

From these 2 conditions, we get:

$c_1 = 2\dfrac{\omega_1 R_1^2 - \omega_2 R_2^2}{R_1^2 - R_2^2}$

$c_2 = -R_1^2 R_2^2 \dfrac{\omega_1 - \omega_2}{R_1^2 - R_2^2}$

With these values, we obtain the velocity field of the cylinder as:

$$\underline{\mathbf{u}} = \boldsymbol{\frac{\omega_2 R_2^2 - \omega_1 R_1^2}{R_2^2 - R_1^2} r} + \boldsymbol{\frac{1}{r}\frac{R_1^2 R_2^2(\omega_1 - \omega_2)}{R_2^2 - R_1^2}}\ \hat{\mathbf{e}}_\theta$$

<u>Evaluating wall sheer stress at the two walls</u>

$\tau = 2\mu\underline{\underline{\epsilon}}$

Where $\underline{\underline{\epsilon}} = \dfrac{1}{2}(\nabla u + (\nabla u)^T)$

In a general coordinate system(x_1, x_2, x_3), we have:

$\tau_{ij} = \mu(\dfrac{\partial u_i}{\partial x_j} + \dfrac{\partial u_j}{\partial x_i})$

However, for the give problem: $u_r = 0, u_z = 0$ and $u_\theta = u_\theta(r)$. So, only one term of the tensor $\tau_{r\theta}$ remains. For cylindrical coordinates, we get-

$\tau_{r\theta} = \mu\Big(\dfrac{du_\theta}{dr} - \dfrac{u_\theta}{r} + \dfrac{1}{r}\dfrac{\partial u_r}{\partial \theta}\Big)$

As $u_r = u_r(r) = 0$:

$$\implies \tau_{r\theta} = \mu\Big(\frac{\omega_2 R_2^2 - \omega_1 R_1^2}{R_2^2 - R_1^2} - \frac{1}{r^2}\frac{R_1^2 R_2^2(\omega_1 - \omega_2)}{R_2^2 - R_1^2} - \frac{\omega_2 R_2^2 - \omega_1 R_1^2}{R_2^2 - R_1^2} - \frac{1}{r^2}\frac{R_1^2 R_2^2(\omega_1 - \omega_2)}{R_2^2 - R_1^2}\Big)$$

$$\implies \tau_{r\theta} = \mu\Big(-\frac{2}{r^2}\frac{R_1^2 R_2^2(\omega_1 - \omega_2)}{R_2^2 - R_1^2}\Big)$$

At inner wall$(r = R_1)$:

$$\tau_{\mathbf{w}}\Big|_{\mathbf{r=R_1}} = \boldsymbol{\frac{2\mu R_2^2(\omega_2 - \omega_1)}{R_2^2 - R_1^2}}$$

At outer wall$(r = R_2)$:

$$\tau_{\mathbf{w}}\Big|_{\mathbf{r=R_2}} = \boldsymbol{\frac{2\mu R_1^2(\omega_2 - \omega_1)}{R_2^2 - R_1^2}}$$

AE 225 QUESTION 4

November 1, 2017

KRISHNA WADHWANI - 160010031

```
In [1]: from numpy import *
        import matplotlib.pyplot as plt
        %matplotlib inline
        from scipy.optimize import fsolve
        import random
        import warnings
        warnings.filterwarnings('ignore')
```

1 Plotting Streamlines

1.1 Potential function is given as

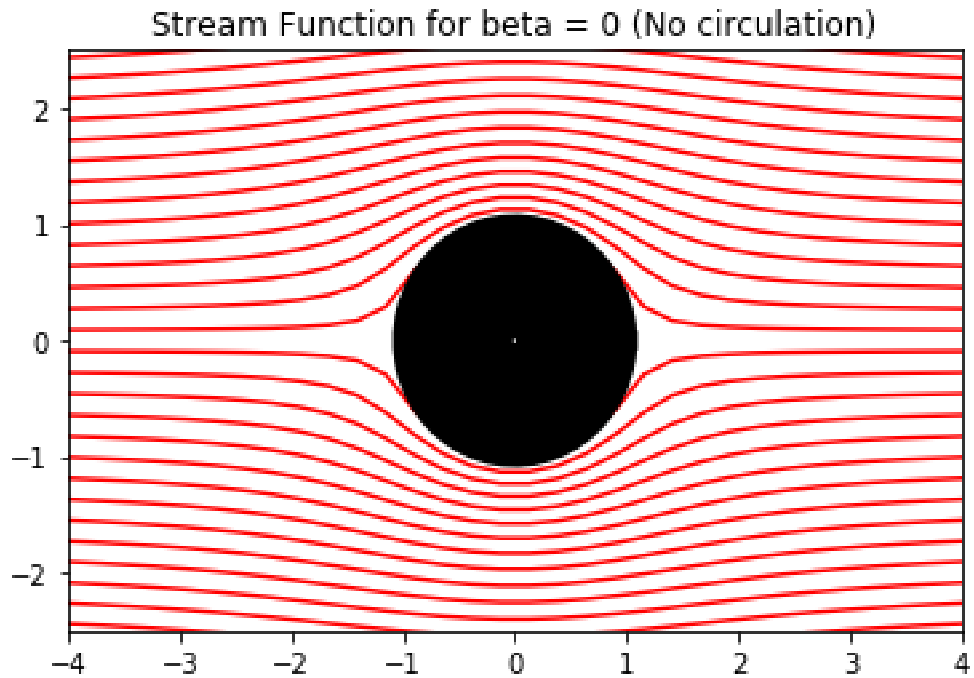
1.1.1 $\psi = Uy - \frac{Ua^2y}{x^2+y^2} - Klog(\sqrt{x^2+y^2})$

```
In [2]: def PSI(y):
                return psi- U*y+(a*a*U *y)/(x*x+y*y)+K*log((x*x+y*y)**0.5)
        U=2
        a=1
        Lambda=a*a*U
        #def singleStreamFunction(K):
            #r=linspace(1+1e-10,10,10000)
            #a=linspace(-5,0,10)
            #b=linspace(0,5,10)
            #psis=concatenate((a,b))
            #if K==0:
             #   psis=linspace(-5,5,30)
           # U=2
            #a=1
           # L=a*a*U
           # for psi in psis:
            #    rr=[]
             #   theta=[]
             #   flag=(psi+K*log(r))/(U*r- L/r)
             #   for i in range(len(flag)):
```

```
    #       if flag[i]<=1 and flag[i]>=-1:
    #           rr.append(r[i])
    #           theta.append(arcsin(flag[i]))
    #   x=rr*cos(theta)
    #   y=rr*sin(theta)
    #    plt.plot(x,y,'r')
   #    plt.plot(-x,y,'r')
   #plt.plot(a*cos(linspace(0,2*pi,50)), a*sin(linspace(0,2*pi,50)) ,'k')
```

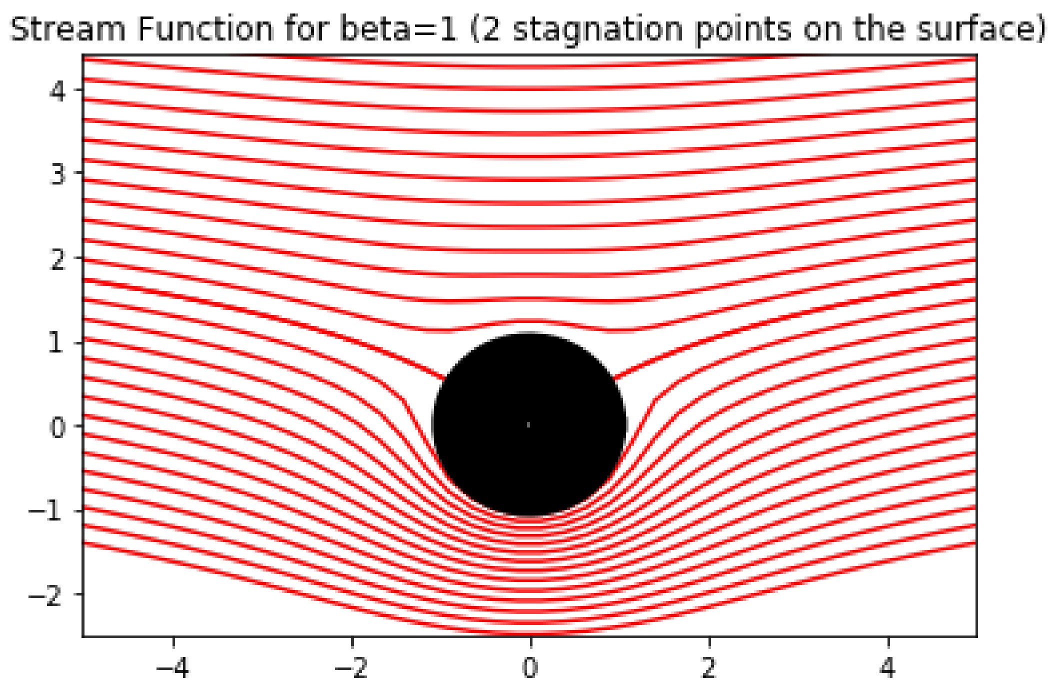
In [3]:
```
#Defining stream function and Variables
psi_array=linspace(-5,5,30)
x_array=linspace(-5,5,40)
K=0
#Plotting streamlines
for psi in psi_array:
    y_list=[]
    x_list=[]
    for x in x_array:
        if len(y_list)==0:
            y=fsolve(PSI,random.randint(0,5))
            y_list.append(y)
            x_list.append(x)
        else:
            y=fsolve(PSI,y_list[-1])
            y_list.append(y)
            x_list.append(x)
    plt.plot(x_list,y_list,'r')
#plotting the circle to show the cylinder
for r in linspace(0,a,10):
    plt.plot(r*cos(linspace(0,2*pi,50)), r*sin(linspace(0,2*pi,50)) ,'k', linewidth=8)
plt.title("Stream Function for beta = 0 (No circulation)")
plt.xlim([-4,4])
plt.ylim([-2.5,2.5])
```

Out[3]: (-2.5, 2.5)



In [10]:
```
q=linspace(-6,0,15)
b=linspace(0,6,15)
psi_array=concatenate((q,b))
x_array=linspace(-5,5,40)
K=2
for psi in psi_array:
    y_list=[]
    x_list=[]
    for x in x_array:
        if len(y_list)==0:
            y=fsolve(PSI,random.randint(0,5))
            y_list.append(y)
            x_list.append(x)
        else:
            y=fsolve(PSI,y_list[-1])
            y_list.append(y)
            x_list.append(x)
    plt.plot(x_list,y_list,'r')
for r in linspace(0,a,10):
    plt.plot(r*cos(linspace(0,2*pi,50)), r*sin(linspace(0,2*pi,50)) ,'k', linewidth=6)
plt.xlim([-5,5])
plt.ylim([-2.5,4.4])
plt.title("Stream Function for beta=1 (2 stagnation points on the surface)")
```

Out[10]: <matplotlib.text.Text at 0x11ac5b2b0>

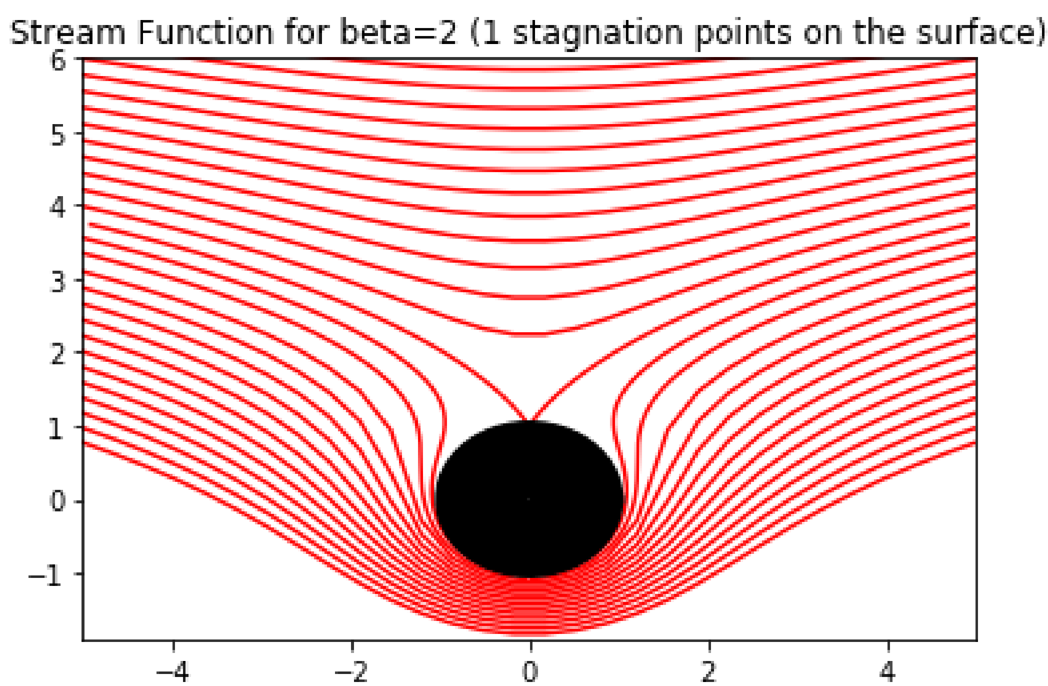



```
In [14]: a=linspace(-5,0,15)
         b=linspace(0,5,15)
         psi_array=concatenate((a,b))

         x_array=linspace(-5,5,30)
         U=2
         a=1
         K=4
         p=[]
         for psi in psi_array:
             y_list=[]
             x_list=[]
             for x in x_array:
                 if psi>-0.9 and psi<=0:
                     p.append(psi)
                     continue
                 elif len(y_list)==0:
                     y=fsolve(PSI,2)
                     y_list.append(y)
                     x_list.append(x)
                 else:
                     y=fsolve(PSI,y_list[-1])
                     y_list.append(y)
                     x_list.append(x)
             plt.plot(x_list,y_list,'r')
```

```
r=linspace(1+1e-10,6.17,10000)
L=a*a*U
#fsolve does not give satisfactory result for the psi values skipped in the above loop
#Plotting streamlines without fsolve for these cases
for psi in [ -0.66666666666666696, -0.33333333333333393,0]:
    rr=[]
    theta=[]
    flag=(psi+K*log(r))/(U*r- L/r)
    for i in range(len(flag)):
        if flag[i]<=1 and flag[i]>=-1:
            rr.append(r[i])
            theta.append(arcsin(flag[i]))
    x=rr*cos(theta)
    y=rr*sin(theta)
    plt.plot(x,y,'r')
    plt.plot(-x,y,'r')
plt.plot(a*cos(linspace(0,2*pi,50)), a*sin(linspace(0,2*pi,50)) ,'k')
for r in linspace(0,a,14):
    plt.plot(r*cos(linspace(0,2*pi,50)), r*sin(linspace(0,2*pi,50)) ,'k', linewidth=4)
plt.title("Stream Function for beta=2 (1 stagnation points on the surface)")
plt.xlim([-5,5])
plt.ylim([-1.9,6])
```

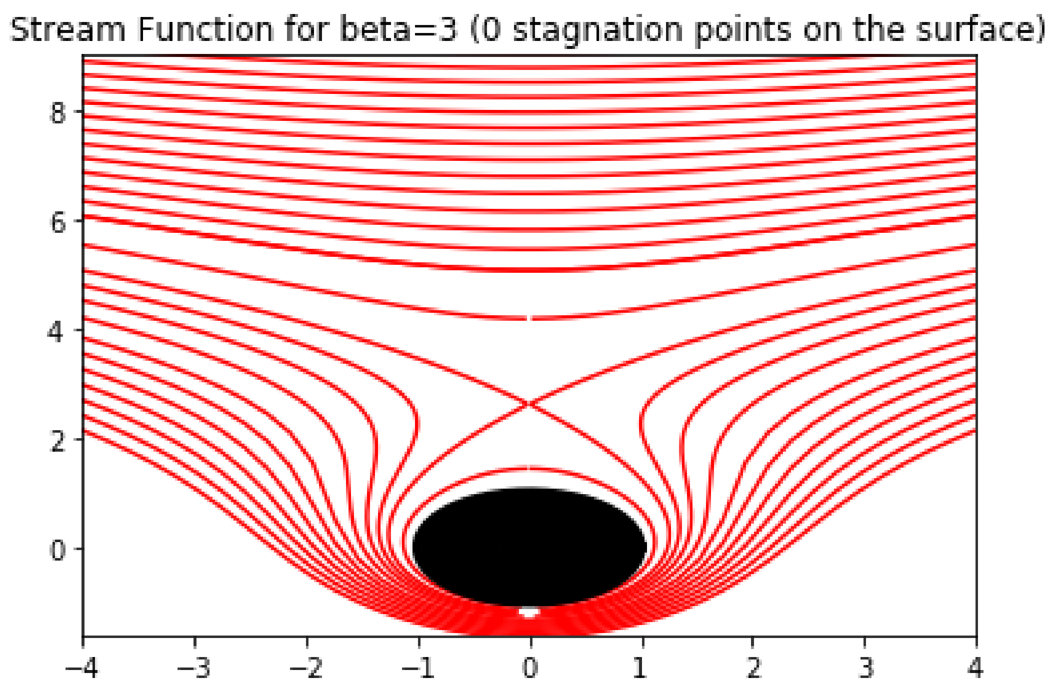
Out[14]: (-1.9, 6)



In [6]:
```
x_array=linspace(-5,5,50)
a=linspace(-5,0,15)
b=linspace(0,5,15)
psi_array=concatenate((a,b))
U=2
a=1
K=6
p=[]
for psi in psi_array:
    y_list=[]
    x_list=[]
    for x in x_array:
        if len(y_list)==0:
            y=fsolve(PSI,1)
            y_list.append(y)
            x_list.append(x)
        elif psi>-2.6 and psi<0:
            p.append(psi)
            continue
        else:
            y=float(fsolve(PSI,y_list[-1]))
            y_list.append(y)
            x_list.append(x)
    plt.plot(x_list,y_list,'r')
#fsolve does not give satisfactory result for the psi values skipped in the above loop
#Plotting streamlines without fsolve for these cases
r=linspace(1+1e-10, 6.9,10000)
for psi in [-2.4,-2,-1.65,-1.3025,-0.7]: #-2.4,-2,-1.65,-1.3025,
    rr=[]
    theta=[]
    flag=(psi+K*log(r))/(U*r- L/r)
    for i in range(len(flag)):
        if flag[i]<=1 and flag[i]>=-1:
            rr.append(r[i])
            theta.append(arcsin(flag[i]))
    x=rr*cos(theta)
    y=rr*sin(theta)
    if psi==-0.7:
        for i in range(len(y)):
            if y[i]>3:
                break_point=i
                break
        plt.plot(x[:break_point],y[:break_point],'r')
        plt.plot(-x[:break_point],y[:break_point],'r')
        plt.plot(x[break_point:],y[break_point:],'r')
        plt.plot(-x[break_point:],y[break_point:],'r')
        break
    plt.plot(x,y,'r')
```

```
        plt.plot(-x,y,'r')
    for r in linspace(0,a,14):
        plt.plot(r*cos(linspace(0,2*pi,50)), r*sin(linspace(0,2*pi,50)) ,'k', linewidth=4)
    plt.title("Stream Function for beta=3 (0 stagnation points on the surface)")
    plt.ylim([-1.6,9])
    plt.xlim([-4.,4.])
```

Out[6]: (-4.0, 4.0)



2 Plotting Equipotential Lines

2.1 Velocity Potential function is defined as:

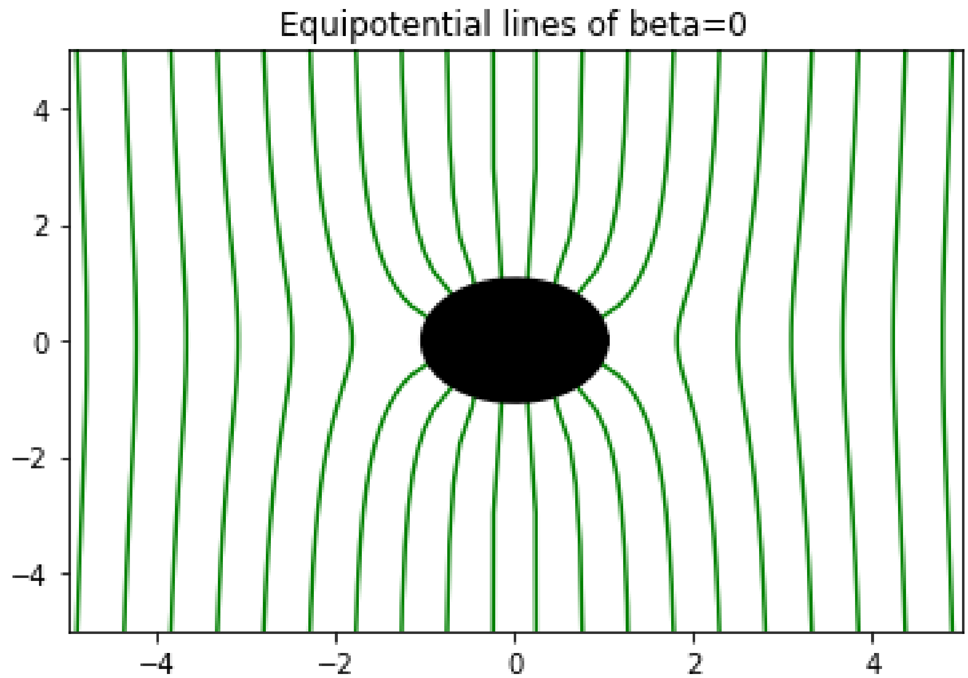
2.1.1 $\phi = Ua(\frac{r}{a} + \frac{a}{r})cos\theta + m\theta$

```
In [15]: def equiPotentialLines(m):
          #Defining variables and velocity potential function
             a=1
             U=1
             def velocity_potential(r):
                 return phi-U*r*cos(theta)-( (U*a*a*cos(theta))/(r) ) - m*theta
             phi_array=linspace(-5,5,20)
             theta_array=linspace(0,2*pi,100)
          #Plotting equipotential lines
             for phi in phi_array:
```

```
            theta_list=[]
            r_list=[]
            for theta in theta_array:
                if len(r_list)==0:
                    r=float(fsolve(velocity_potential, random.randint(0,3)))
                    if r>0:
                        r_list.append(r)
                        theta_list.append(theta)
                else:
                    r=float(fsolve(velocity_potential, r_list[-1]))
                    if r>0:
                        r_list.append(r)
                        theta_list.append(theta)
            x=r_list*cos(theta_list)
            y=r_list*sin(theta_list)
            plt.plot(x[where(x<0)], y[where(x<0)],'g')
            plt.plot(-x[where(x<0)], y[where(x<0)],'g')
        for i in linspace(0,a,15):
            plt.plot(i*cos(linspace(0,2*pi,50)), i*sin(linspace(0,2*pi,50)) ,'k',linewidth=
        if m==3:
            plt.xlim([-13.5,13.5])
            plt.ylim([-7,7])
        elif m==2:
            plt.xlim([-10.5,10.5])
            plt.ylim([-5,5])
        elif m==1:
            plt.xlim([-8,8])
            plt.ylim([-5,5])
        elif m==0:
            plt.xlim([-5,5])
            plt.ylim([-5,5])
```

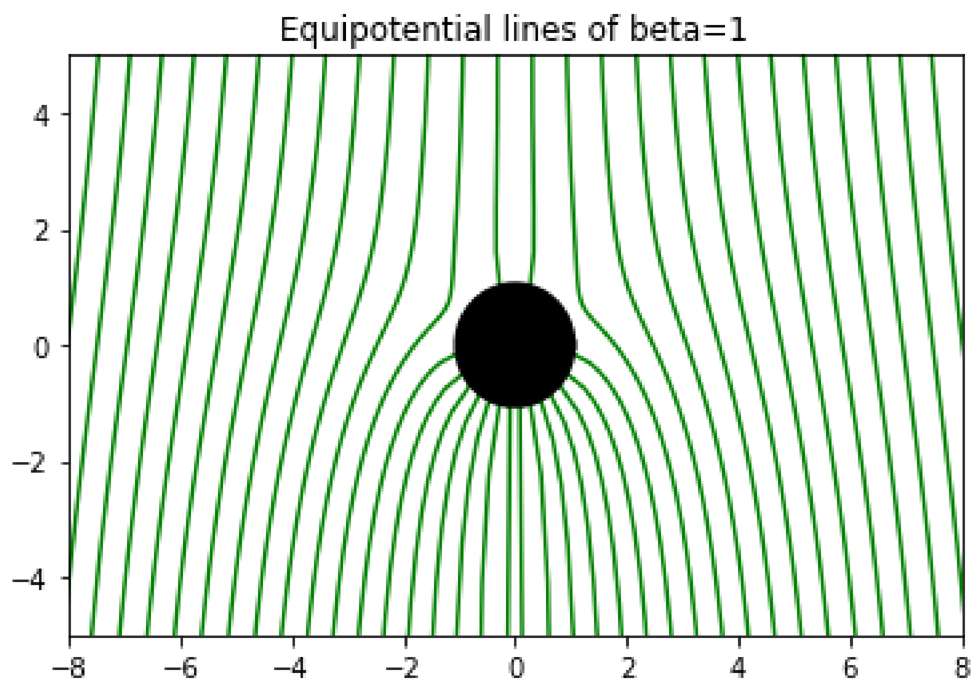
```
In [16]: equiPotentialLines(0)
         plt.title("Equipotential lines of beta=0")

Out[16]: <matplotlib.text.Text at 0x11b1ec550>
```



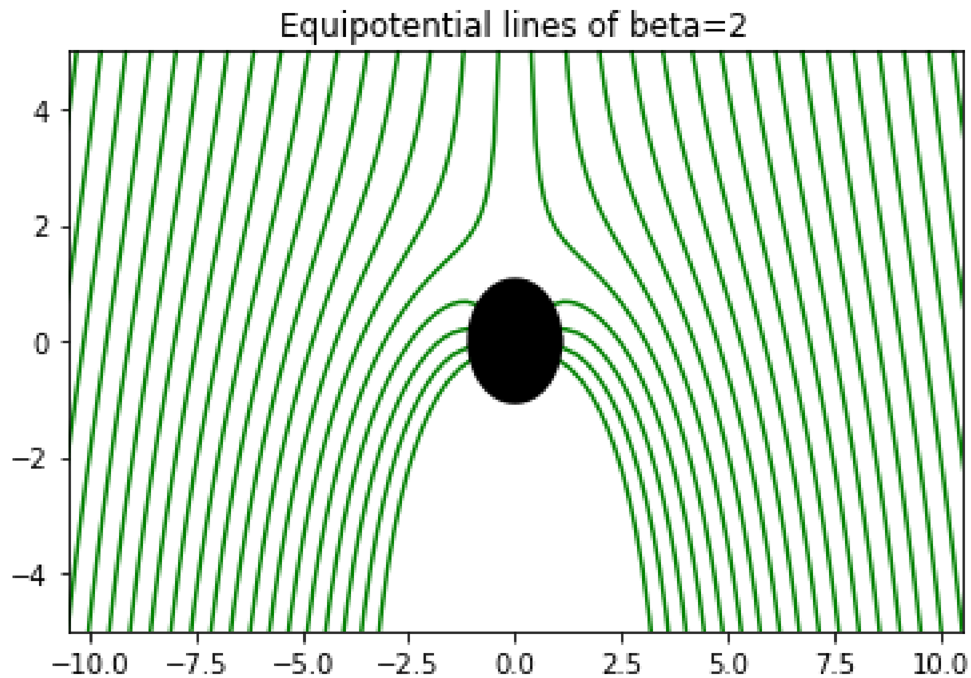

```
In [17]: equiPotentialLines(1)
         plt.title("Equipotential lines of beta=1")

Out[17]: <matplotlib.text.Text at 0x11bb06518>
```



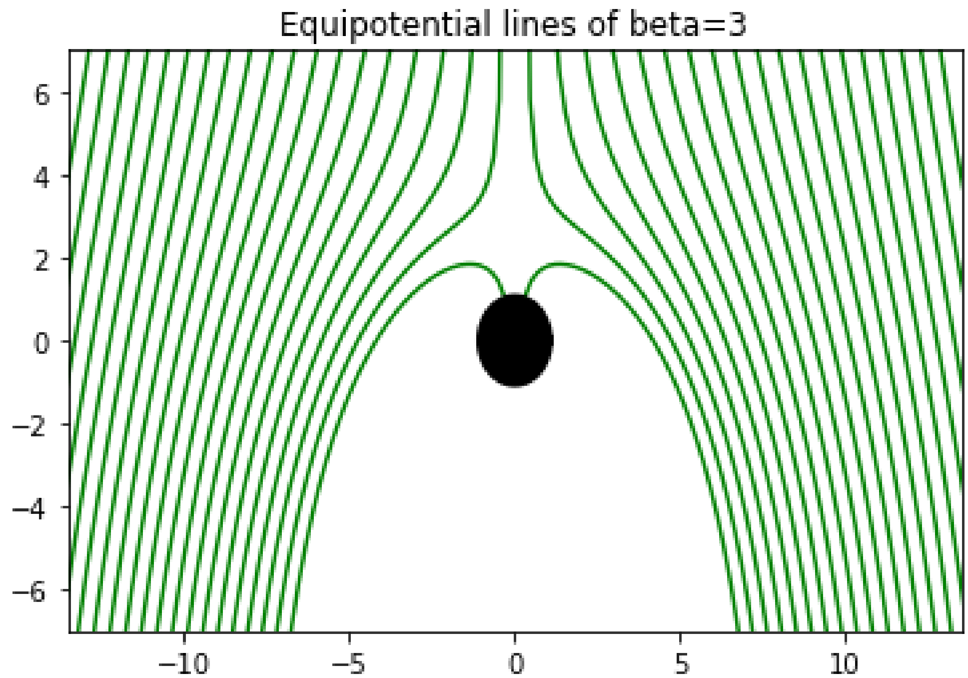
```
In [18]: equiPotentialLines(2)
         plt.title("Equipotential lines of beta=2")
```

Out[18]: <matplotlib.text.Text at 0x11ab753c8>




```
In [19]: equiPotentialLines(3)
         plt.title("Equipotential lines of beta=3")
```

Out[19]: <matplotlib.text.Text at 0x11b54a588>



3 THE END