Python code for streamlines representation

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import special as sp
fig = plt.figure()
ax = fig.add_subplot(111)
a = 4.4934
r = np.linspace(0.01,1,100)
#s = 0.5*(a**(-4))*np.sqrt(r)*sp.jv(1.5,r)
#s = s + (1/6)*(a**(-3))*(r**2 - r**4)*sp.jv(2.5,a)
r0 = np.linspace(0.1,1,10, endpoint = False)
t = np.linspace(0,2*np.pi,100)
x = np.cos(t)
y = np.sin(t)
ax.plot(x,y, color = "black", linewidth = 3)
for r1 in r0 :
    s0 = 0.5*(a**(-4))*np.sqrt(r1)*sp.jv(1.5, a*r1)
    s0 = s0 + (1/6)*(a**(-3))*(r1**2 - r1**4)*sp.jv(2.5,a)
    h = [0,0]
    w = [0,0]
    for i in range(2) :
        h[i] = np.array([])
        w[i] = np.array([])
    for r2 in r :
        s = 0.5*(a**(-4))*np.sqrt(r2)*sp.jv(1.5,a*r2)
        s = s + (1/6)*(a**(-3))*(r2**2 - r2**4)*sp.jv(2.5,a)
        t = np.sqrt(2*s0)/np.sqrt(2*s)
        if t <= 1 :
            h[0] = np.append(h[0], r2)
            h[1] = np.append(h[1], r2)
            w[0] = np.append(w[0], np.arcsin(t))
            w[1] = np.append(w[1], np.pi - np.arcsin(t))
    n = h[0].size
    for i in range(2) :
        ix = np.array([0])
        iy = np.array([r1])
        x = h[i]*np.cos(w[i])
        y = h[i]*np.sin(w[i])
        j = 0
        while j < n :
            x0 = ix[j]
            y0 = iy[j]
            x1 = x
            y1 = y
            l = 0
            while l == 0 and j < n-1 :
                d = (x1 - x0)**2 + (y1 - y0)**2
                index = np.where(d == np.amin(d))
                q = 0
                for p in index :
                    if not(x1[p] in ix) :
                        q = 1
                        k = p
                    if not(y1[p] in iy) :
                        q = 1
                        k = p
                if q == 0 :
                    x1 = np.delete(x1, index)
                    y1 = np.delete(y1, index)
                else :
                    ix = np.append(ix, x1[k])
```

```
                iy = np.append(iy, y1[k])
                l = 1
            j = j+1
        ir = ix**2 + iy**2
        ir = np.sqrt(ir)
        ic = ix/ir
        it = iy/ir
        iw = (a**(-4))*np.sqrt(ir)*sp.jv(1.5, a*ir)
        iw = iw + (1/3)*(a**(-3))*(ir - ir**3)*sp.jv(2.5, a)
        u = (1/ir)*ic*iw
        v = (1/3)*(a**(-3))*(1 - 2*(ir**2))*sp.jv(2.5, a)
        vi = 0.25*(a**(-4))*((ir**(-1.5)))*sp.jv(1.5, a*ir)
        vi = vi - 0.5*(a**(-3))*(ir**(-0.5))*sp.jv(2.5, a*ir)
        v = -it*(v + vi)
        ux = u*ic - v*it
        uy = u*it + v*ic
        nv = np.sqrt(u**2 + v**2)
        mx = np.amax(nv)
        c = 0.1/mx
        ax.plot(iy,ix, color = "black", linewidth = 2)
        ax.plot(-iy,ix, color = "black", linewidth = 2)
        ix = ix[:: 10]
        iy = iy[:: 10]
        u = c*ux[:: 10]
        v = c*uy[:: 10]
        ax.quiver(iy,ix,v,u,units = "width", angles = "xy")
        ax.quiver(-iy,ix,-v,u,units = "width", angles = "xy")
ax.set_title("Fig1.  streamlines", fontdict = ({"fontname" : "DejaVu Sans",
"fontsize" : 12}))
ax.set_xlabel("x axis")
ax.set_ylabel("y axis")
#fig.savefig("/home/victor/Dokumente/str.png")
plt.show()
```