

# Exercício 1

Filomen Incahuanaco Quispe

31 de Agosto de 2017

## 1 Introdução

O presente trabalho tem como entrada uma curva simples fechada e a saída são os bordos de um domínio retangular. Foram usado dois arquivos para teste `naca012.txt` e `mrc020.txt`(obtida de [airfoiltools.com/airfoil](http://airfoiltools.com/airfoil) )

As principais dificuldades foram a orientação do preenchimento das bordas externas da curva, e obter os deltas para a discretização. Que em nosso caso é um fator obtido com o número de pontos da curva dividido por 6, além de fazer um trato especial para casos de número impar. Finalmente ao gerar o arquivo como dos casas decimais o curva interna se distorçona, o cual se resolve só trocando a 3 casas decimais ou maior. Como isso pode-se tratar qualquer entrada, como se pode mostrar nas Figuras 1 e 2.

## 2 Malhas obtidas

Figura 1: Resultado para entrada naca012, como 63 pontos

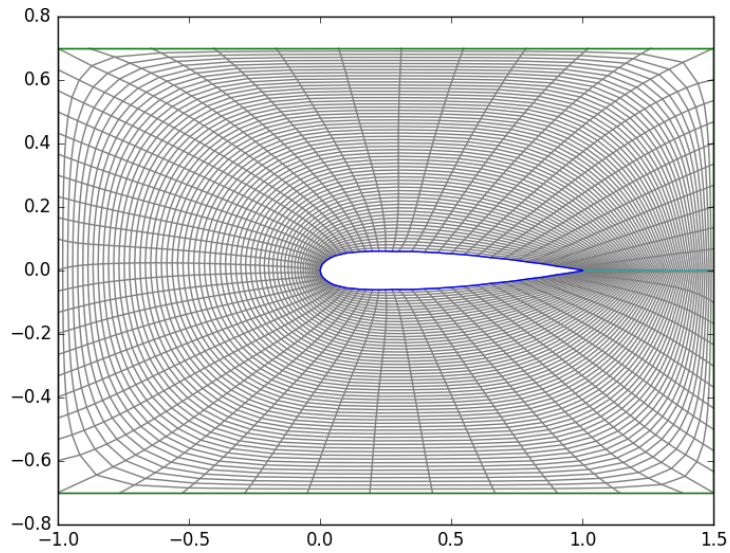
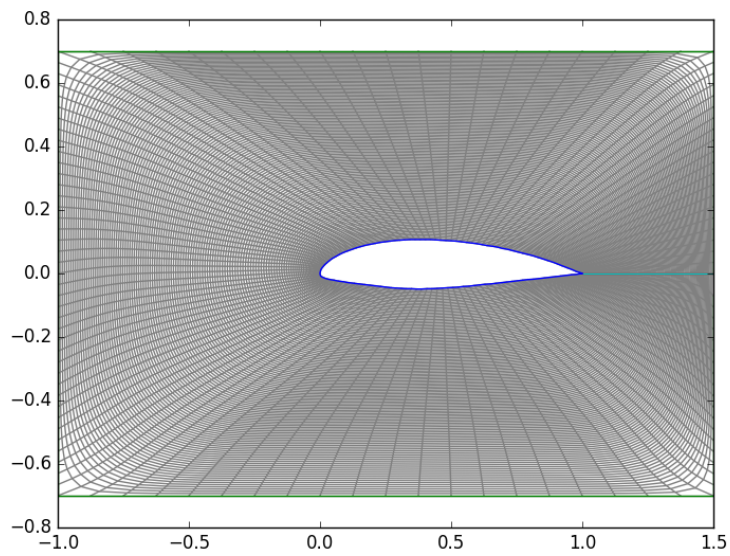


Figura 2: Resultado para entrada mrc020, com 120 pontos



### 3 Execução

```
python excer1.py naca012.txt naca012_out.txt
python elliptic.py naca012_out.txt
```

### 4 Código fonte

---

```
import sys
import numpy as np
import matplotlib.pyplot as plt

f = open(sys.argv[1], 'rt')
ah_x=[]
ah_y=[]
for l in f.readlines():
    numbers_float = map(float, l.split())
    ah_x.append(numbers_float[0])
    ah_y.append(numbers_float[1])

ah_size=len(ah_x)
#consider grid from bottom to top
#
#
#
#
#AH
rt = np.zeros((ah_size,2))
for i in range(ah_size):
    rt[i,0]=ah_x[i]
    rt[i,1]=ah_y[i]

#AB & HG
deltaAB=(1.5 - ah_x[0] )/(ah_size) # ==deltaHG
rl = np.zeros((ah_size,2))
rr = np.zeros((ah_size,2))
for i in range(ah_size):
    rl[i,0]=1.5 - i*deltaAB #ah_x[0]+i*deltaAB
    rl[i,1]=ah_y[0]

    rr[i,0]=1.5 - i*deltaAB #ah_x[ah_size-1]+i*deltaAB
    rr[i,1]=ah_y[ah_size-1]
```

```

#BC,FG, DE & CD, EF
rb = np.zeros((ah_size,2))
b=int(ah_size/6) #assuming a=b, 2a+b+2a+b=ah_size
a=int(ah_size/6)
print (ah_size % 2)
if( ah_size % 2==1):
    b=b+1

deltaBC= (0.7 - -0.7) / ah_size
deltaFG= (0.7 - -0.7) / ah_size
deltaDE= (0.7 - -0.7) / ah_size
deltaCD= (1.5 - -1.0) / ah_size
deltaEF= (1.5 - -1.0) / ah_size

for i in range(a): #for a
    #BC
    rb[i,0]=1.5
    rb[i,1]=ah_y[0] + i*(deltaBC*3)
    #FG
    rb[3*a+2*b+i,0]=1.5
    rb[3*a+2*b+i,1]=-0.7 + i*(deltaFG*3)
    #DE/2
    rb[a+b+i,0]=-1.0
    rb[a+b+i,1]=0.7 - i*(deltaDE*3)
    #DE/2
    rb[2*a+b+i,0]=-1.0
    rb[2*a+b+i,1]=ah_y[0] - i*(deltaDE*3)

for i in range(b): #for b
    rb[a+i,0]=1.5-i*(deltaCD*6)
    rb[a+i,1]=0.7
    rb[3*a+b+i,0]=-1.0+i*(deltaEF*6)
    rb[3*a+b+i,1]=-0.7

rb[ah_size-1]=rb[ah_size-2]

f = open(sys.argv[2], 'wt')
f.write(str(ah_size))
f.write('\n')

```

```

for i in range(ah_size):
    f.write("{:.3f}".format(rt[i,0]))
    f.write(' ')
    f.write("{:.3f}".format(rt[i,1]))
    f.write('\n')
f.write(str(ah_size))
f.write('\n')
for i in range(ah_size):
    f.write("{:.3f}".format(rb[i,0]))
    f.write(' ')
    f.write("{:.3f}".format(rb[i,1]))
    f.write('\n')

f.write(str(ah_size))
f.write('\n')
for i in range(ah_size):
    f.write("{:.3f}".format(rl[i,0]))
    f.write(' ')
    f.write("{:.3f}".format(rl[i,1]))
    f.write('\n')

f.write(str(ah_size))
f.write('\n')
for i in range(ah_size):
    f.write("{:.3f}".format(rr[i,0]))
    f.write(' ')
    f.write("{:.3f}".format(rr[i,1]))
    f.write('\n')

f.close()

```

---