

```
In [1]: from functions_bending_schreyer_adaptive89 import *
import matplotlib.pyplot as plt
from mpmath import mpmathify
import mpmath as mp
import numpy as np
import math
import time
from scipy.interpolate import CubicSpline
from rungekuttacoefficients import *
```

```
In [2]: print("Main!")
L = 0.1
wh = 0.001
def hsc(s):
    return 0.01 # + (s - L/2)**2 * 4

def h(s):
    return wh * hsc(s)
s_eval = mp.matrix(np.linspace(0,L,int(200),endpoint = True))
ret = bend_theta_y(s_eval, h, thickness=mpmathify(0.001), E=mpmathify(10**10), Fweight=mpmathify(0.1), y0 = mpmathify(
```

Main!

STARTING!(IC): [1.0e-1200]

[0.0]

[0.0]

[0.0]

LOOP 89

DONE 89

STARTING!(IC): [0.0]

[0.0]

[1.0e-1200]

[0.0]

LOOP 89

DONE 89

[[mpf('5.5187653698417795e+164'), mpf('1.3495896834808172e+168')], [mpf('2.1374086369023423e+168'), mpf('5.2269383683345427e+171')]]

mat 3872.98334620742 3872.98334620742 DET: 0

0.0

STARTING!(IC): [1.0e-1200]

[0.0]

[0.0]

[0.0]

```
LOOP 89
1000 0.0101705581095843
2000 0.0115826174598864
3000 0.0132204502397856
4000 0.01350956692616
5000 0.0152993910872126
6000 0.016053015532474
7000 0.0175825965779707
8000 0.0183284013791165
9000 0.0194045527460875
10000 0.0204229134296535
11000 0.0213927648512422
12000 0.0218299018800797
13000 0.0230038996994384
14000 0.0235960580704904
15000 0.0247772579376312
16000 0.0260845296544157
17000 0.0263132639245164
18000 0.0275158696648022
19000 0.0284911302686964
20000 0.0292434709066509
21000 0.0298971113447795
22000 0.0316021824516168
23000 0.0319565901301478
24000 0.0334201524031739
25000 0.034650477084435
26000 0.0352877548446452
27000 0.0362285753814015
28000 0.0371957008991907
29000 0.0382945622247444
30000 0.039287480545215
31000 0.0416636534744637
32000 0.0423857970476705
33000 0.0432344207397452
34000 0.0440085503806851
35000 0.0448598302376718
36000 0.0456566876333236
37000 0.0465354301362947
38000 0.0475156082415157
39000 0.0483987915142532
40000 0.0492163547059143
41000 0.0502519456545544
42000 0.050680771323256
43000 0.0517366928407151
44000 0.0529289345762111
45000 0.0539478523457587
```

```
46000 0.054392303942482
47000 0.0556267007907198
48000 0.056178233078078
49000 0.0585732693571138
50000 0.0597615371294779
51000 0.0602653823500702
52000 0.0615302508309029
53000 0.0623895432718901
54000 0.0637121958492574
55000 0.0644468966020144
56000 0.0663515129003041
57000 0.0670003259405938
58000 0.0680862194629167
59000 0.0693988604521526
60000 0.0702378337261796
61000 0.0714898932972628
62000 0.0719249676132467
63000 0.0726075189284147
64000 0.073848128577513
65000 0.0746571315354182
66000 0.0760226064208084
67000 0.0769621472414525
68000 0.0777706929407818
69000 0.0786333324119754
70000 0.079744695573595
71000 0.0806633593706683
72000 0.0821336508465454
73000 0.0825335464380948
74000 0.0846753142734293
75000 0.0862161982123014
76000 0.0875289175887235
77000 0.0886949735004499
78000 0.0896297878526313
79000 0.091474710803931
80000 0.0916499172938994
81000 0.0929790217662317
82000 0.0936241931422168
83000 0.0943193543769542
84000 0.0951539245503069
85000 0.0961281038701362
86000 0.0967205433161089
87000 0.0976619886433726
88000 0.0985833683800604
DONE 89
STARTING!(IC): [ 0.0]
[ 0.0]
```

```
[      0.0]
[1.0e-1200]
LOOP 89
1000 0.0110066757249416
2000 0.0118569731823533
3000 0.0130821356762307
4000 0.0140224894450921
5000 0.0155026310975227
6000 0.0167403973048977
7000 0.0178860341070484
8000 0.0189784645232656
9000 0.0199069775464758
10000 0.0212895031303411
11000 0.0221780648630669
12000 0.0224916575961388
13000 0.0249746100873967
14000 0.0266614857069136
15000 0.0284616795805438
16000 0.0294216772327387
17000 0.030712278089056
18000 0.0314172497401267
19000 0.0334603153239966
20000 0.0341297385232811
21000 0.0345470818084622
22000 0.0358819936781552
23000 0.0362650093467763
24000 0.0371951064254386
25000 0.0379059856087398
26000 0.0387842742287715
27000 0.0399582477539977
28000 0.0406946052314963
29000 0.0421343797099569
30000 0.0428303324382158
31000 0.0434663362559619
32000 0.0446610413661745
33000 0.0456147059552586
34000 0.0468533314574653
35000 0.0482415301332612
36000 0.0490238551404131
37000 0.0502748493134759
38000 0.0511636798900411
39000 0.0517944769914397
40000 0.052362642718413
41000 0.0542207199302845
42000 0.0552445986555065
43000 0.056474434834319
```

```
44000 0.0572876063188929
45000 0.0578934415899602
46000 0.0588883692456495
47000 0.0613612100117496
48000 0.0626265772630731
49000 0.0639611984458117
50000 0.064603170855555
51000 0.0655068553905063
52000 0.0668085636511652
53000 0.0677364186631099
54000 0.0686469988149916
55000 0.069239291754798
56000 0.0701450635654706
57000 0.0711358294987686
58000 0.073006102243771
59000 0.0745309178480492
60000 0.0754126122025172
61000 0.0763567919733368
62000 0.0772571612150509
63000 0.0780044864065926
64000 0.0788553122182906
65000 0.0794191452949136
66000 0.080525673127945
67000 0.0815930036500777
68000 0.0826761145322936
69000 0.0835572268746196
70000 0.0845480927061144
71000 0.0859546605856005
72000 0.0872698275576622
73000 0.0885362423775736
74000 0.0894336309319902
75000 0.0909561902820269
76000 0.0916031500428807
77000 0.0927862973593715
78000 0.0939707977598055
79000 0.095002635052592
80000 0.0960501416995315
81000 0.0964252912062826
82000 0.0981871754081074
DONE 89
[[mpf('5.5187653698417795e+164'), mpf('1.3495896834808172e+168')], [mpf('2.1374086369023423e+168'), mpf('5.22693836833
45427e+171')]]
[[mpf('1.4013164907985023e+167'), mpf('7.4656349071842888e+170')], [mpf('-6.6405484303458889e+151'), mpf('-2.421184712
3695839e+156')]]
```

```
In [3]: ret[0]#3.8e3
```

```
Out[3]: [[mpf('1.4013164907985023e+167'), mpf('7.4656349071842888e+170')],  
         [mpf('-6.6405484303458889e+151'), mpf('-2.4211847123695839e+156')]]
```

```
In [4]: (ret[-1])
```

```
Out[4]: [[mpf('5.5187653698417795e+164'), mpf('1.3495896834808172e+168')],  
         [mpf('2.1374086369023423e+168'), mpf('5.2269383683345427e+171')]]
```

```
In [5]: mp.det(mp.matrix(ret[0]))
```

```
Out[5]: mpf('-2.8970869630683457e+323')
```

```
In [6]: M = ret[0]  
        M[0][0] * M[1][1] - M[1][0] * M[0][1]
```

```
Out[6]: mpf('-2.8970869630683453e+323')
```

```
In [8]: M = ret[1]  
        M[0][0] * M[1][1] - M[1][0] * M[0][1]
```

```
Out[8]: mpf('-2.8970869630683453e+323')
```

```
In [ ]:
```