

~\Documents\documents_general\structured_courses\math564\evaluations\projects
p05\solve6.py

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  """
4  script for solving the Minimum Time Path Problem
5  Version: November 9, 2023
6  Author: Tom Asaki
7  """
8
9  import numpy as np
10 import optimize as opt
11 import pandas as pd
12 import matplotlib.pyplot as plt
13
14 from objective import pathtime as obj
15
16 N=12 # This is the order of the fit (2N decision variables)
17
18 # read in the velocity data array defined on
19 # [0,1]x[0,1] and set the path end points
20 v=pd.read_csv('SpeedData.csv',header=None).to_numpy()
21 my,mx=v.shape
22 A=(.05,.05)
23 B=(.95,.95)
24
25 alg=dict(obj      = obj,
26          x0        = 0.2*np.random.randn(2*N,1),
27          params     = (v,A,B),
28          method     = 'BFGS',
29          maxiter    = 999,
30          progress   = 10,
31          ngtol      = 1E-8,
32          dftol      = 1E-8,
33          dxtol      = 1E-8,
34          Lambda     = 1.,
35          Lambdamax   = 100.,
36          linesearch = 'StrongWolfe',
37          c1         = 0.001,
38          c2         = 0.9,
39          m          = 10,
40          maxcond    = 1000,
41          )
42
43 res=opt.minimize(alg)
44
45 #####
46 # plot the optimal path superimposed on the velocity image
47 smp      = 1000 # number of points defining the path
48 FigDPI   = 256  # figure dpi (effects scale)
49 FigSize  = (8,6) # fiugure size
50 ColorMap = 'jet' # velocity colormap
51 LineColor = 'white' # path plot color
```

```
52 LineWidth    = 1          # path line width
53 PointSize    = 16         # size of path endpoints
54
55 r=np.linspace(0,1,smp)
56 xx=(1-r)*A[0]+r*B[0]
57 yy=(1-r)*A[1]+r*B[1]
58 for k in range(N):
59     s=np.sin((k+1)*np.pi*r)
60     xx+=res['x'][k,-1]*s
61     yy+=res['x'][k+N,-1]*s
62 xxr=xx*(mx-1)
63 yyr=yy*(my-1)
64
65 fig = plt.figure(dpi=FigDPI,figsize=FigSize)
66 ax=fig.add_subplot()
67 vim=ax.imshow(v,cmap=ColorMap)
68 plt.colorbar(vim,orientation='vertical')
69 ax.plot(yyr,xxr,color=LineColor,linewidth=LineWidth)
70 ax.scatter(A[1]*mx,A[0]*my,PointSize,LineColor)
71 ax.scatter(B[1]*mx,B[0]*my,PointSize,LineColor)
72 plt.xticks([])
73 plt.yticks([])
74 plt.show()
75
76
77
78
```