“Optimization”

Shervin Halat

98131018

Homework #2

(Implementation Problems)

9.

Considering the LP form of the mentioned exercise of the problem:

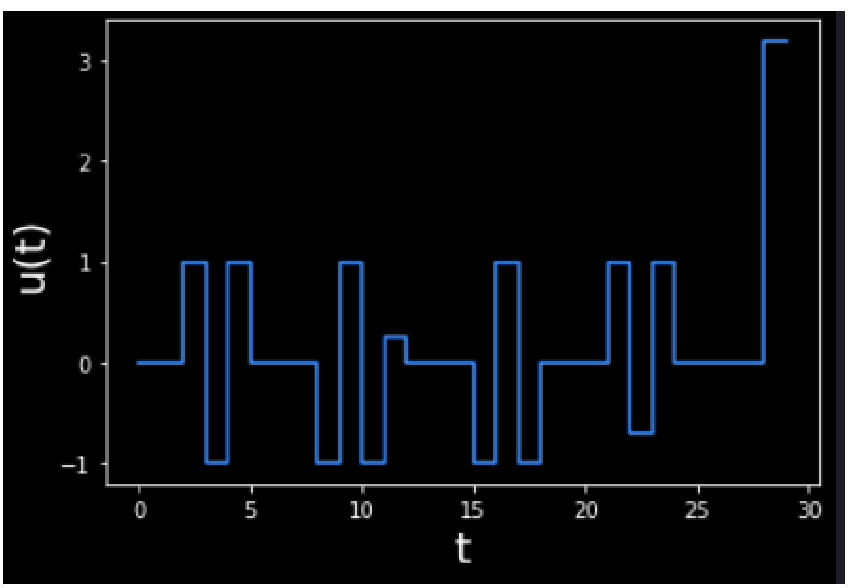
$$\begin{array}{ll}\text{minimize} & \mathbf{1}^T t \\ \text{subject to} & -t \preceq u \preceq t \\ & -\frac{t+\mathbf{1}}{2} \preceq u \preceq \frac{t+\mathbf{1}}{2} \\ & Cu = x_{\text{des}}\end{array}$$

Optimal value and optimal points are as follows:

```
print(f'optimal value is as follows:\n{prob.value}\n'\
      ,f'optimal point is as follows:\n{t.value}')
```

```
optimal value is as follows:
17.323567853890694
 optimal point is as follows:
[-3.58589859e-11 -3.06073695e-11  1.25789129e-09  1.00000000e+00
  1.00000000e+00  1.00000000e+00  6.17798481e-12 -4.16078206e-11
 -2.62306116e-12  1.00000000e+00  1.00000000e+00  1.00000000e+00
  2.46624155e-01 -3.14831421e-11 -3.90765007e-11  1.01410960e-10
  1.00000000e+00  1.00000000e+00  1.00000000e+00  8.25374829e-11
 -4.16082163e-11 -3.53429556e-11  1.00000000e+00  6.98881473e-01
  1.00000000e+00 -8.80602338e-12  1.36527074e-09  1.56960048e-11
 -4.57250850e-11  5.37806222e+00]
```

The obtained plot of u(t) vs t is as follows:



10.

a.

```
Part 'A':

i = 1 & n = 100:
|Ax - b| l2-norm = 0.00000
execution time = 0.0030

i = 1 & n = 400:
|Ax - b| l2-norm = 0.00000
execution time = 0.0020

i = 1 & n = 1600:
|Ax - b| l2-norm = 0.00000
execution time = 0.0239

i = 2 & n = 100:
|Ax - b| l2-norm = 0.11533
execution time = 0.0090

i = 2 & n = 400:
|Ax - b| l2-norm = 0.00185
execution time = 0.0229

i = 2 & n = 1600:
|Ax - b| l2-norm = 0.06461
execution time = 0.7321
```

b.

```
Part 'B':

i = 1 & n = 100:
|Ax - b| l2-norm = 0.00000
|A - A_hat| l2-norm = 0.0001
|xs - xs_hat| l2-norm = 0.00025
execution time = 0.0040

i = 1 & n = 400:
|Ax - b| l2-norm = 0.00000
|A - A_hat| l2-norm = 0.0004
|xs - xs_hat| l2-norm = 0.00199
execution time = 0.0020

i = 1 & n = 1600:
|Ax - b| l2-norm = 0.00000
|A - A_hat| l2-norm = 0.0016
|xs - xs_hat| l2-norm = 0.01597
execution time = 0.0244

i = 2 & n = 100:
|Ax - b| l2-norm = 0.05469
|A - A_hat| l2-norm = 0.0001
|xs - xs_hat| l2-norm = 8922.29148
execution time = 0.0098

i = 2 & n = 400:
|Ax - b| l2-norm = 0.00408
|A - A_hat| l2-norm = 0.0004
|xs - xs_hat| l2-norm = 97108.95613
execution time = 0.0229

i = 2 & n = 1600:
|Ax - b| l2-norm = 0.09942
|A - A_hat| l2-norm = 0.0016
|xs - xs_hat| l2-norm = 12624251.49666
execution time = 0.7162
```

c.

As it can be figured out from the comparison of the results shown above, the $|x^* - \hat{x}^*|_2$ remarkably increases as the matrix 'A' switches from 'tridiagonal' matrix to 'Hilbert' matrix as against of that for 'tridiagonal' matrix. Besides, $|x^* - \hat{x}^*|_2$ increases exponentially as the dimension of the 'Hilbert' matrix increases from 100 to 1600. The main reason is that 'Hilbert' matrix is an ill-conditioned type matrix; therefore, it is a difficult type of matrices to use in numerical computations. Also, 'Hilbert' matrix determinant converges to zero as its dimension increases which is consistent with the results shown above since for conjugate gradient algorithm, 'A' matrix is assumed to be positive definite and consequently should has positive determinant.

Approximation of 'Hilbert' matrix determinant in terms of its dimension is shown below:

$$\det(H) \sim a_n \, n^{-1/4} (2\pi)^n \, 4^{-n^2}$$

Moreover, it can be concluded that the 'Hilbert' matrix is remarkably unstable to even very small amount of noise and this increases by increase in its dimension. This fact results in a huge gap between obtained optimal 'x*' points in part 'a' and 'b' for 'Hilbert' matrix as against of that for 'tridiagonal' matrix.