

CMP 755 Robotics

HW5

1. Attached python script is written to draw cspace of the manipulator. The resulting cspace is given in Figure 1. Lower half is restricted (purple) because of Theta_1 limits defined by question. The purple in upper half is where one of the links intersect with obstacle segment AB.

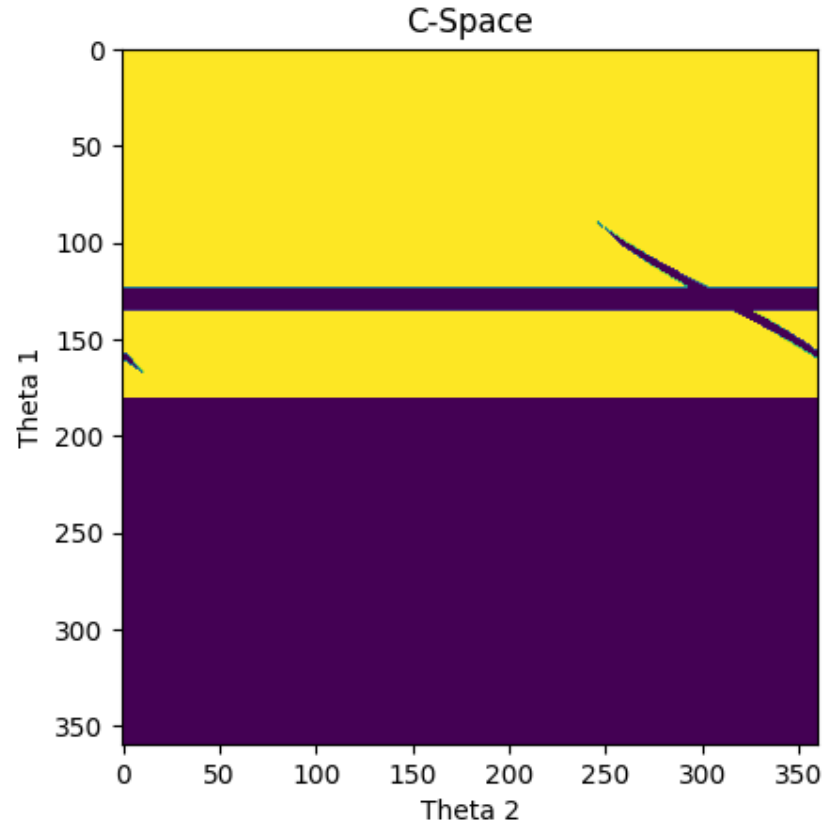


Figure 1: cspace of 2-link manipulator (purple: Restricted, yellow: Free)

2. A* algorithm is applied both with python code and by hand. The results are as in Figure 2 and Figure 3. Python code is attached to report.

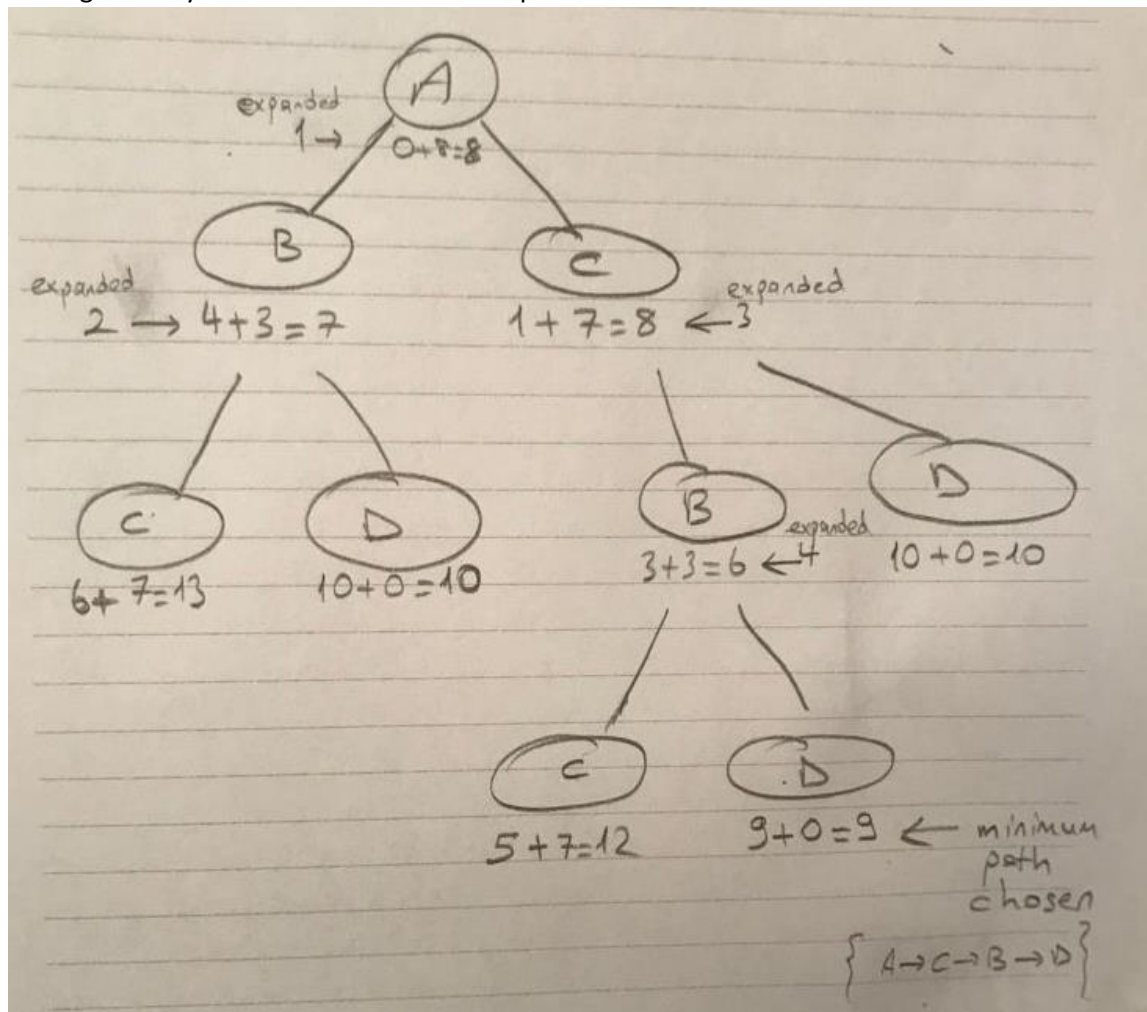


Figure 2: A* algorithm to find shortest path A-D

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_64\python.exe
Path found: ['A', 'C', 'B', 'D']
Press any key to continue . . .
```

Figure 3: Results of A* algorithm implemented on Python

3. Voronoi diagram is drawn for points $A=[-1,0]$, $B=[0,-2]$, $C=[1,0]$, $D=[0,2]$ as in Figure 4.

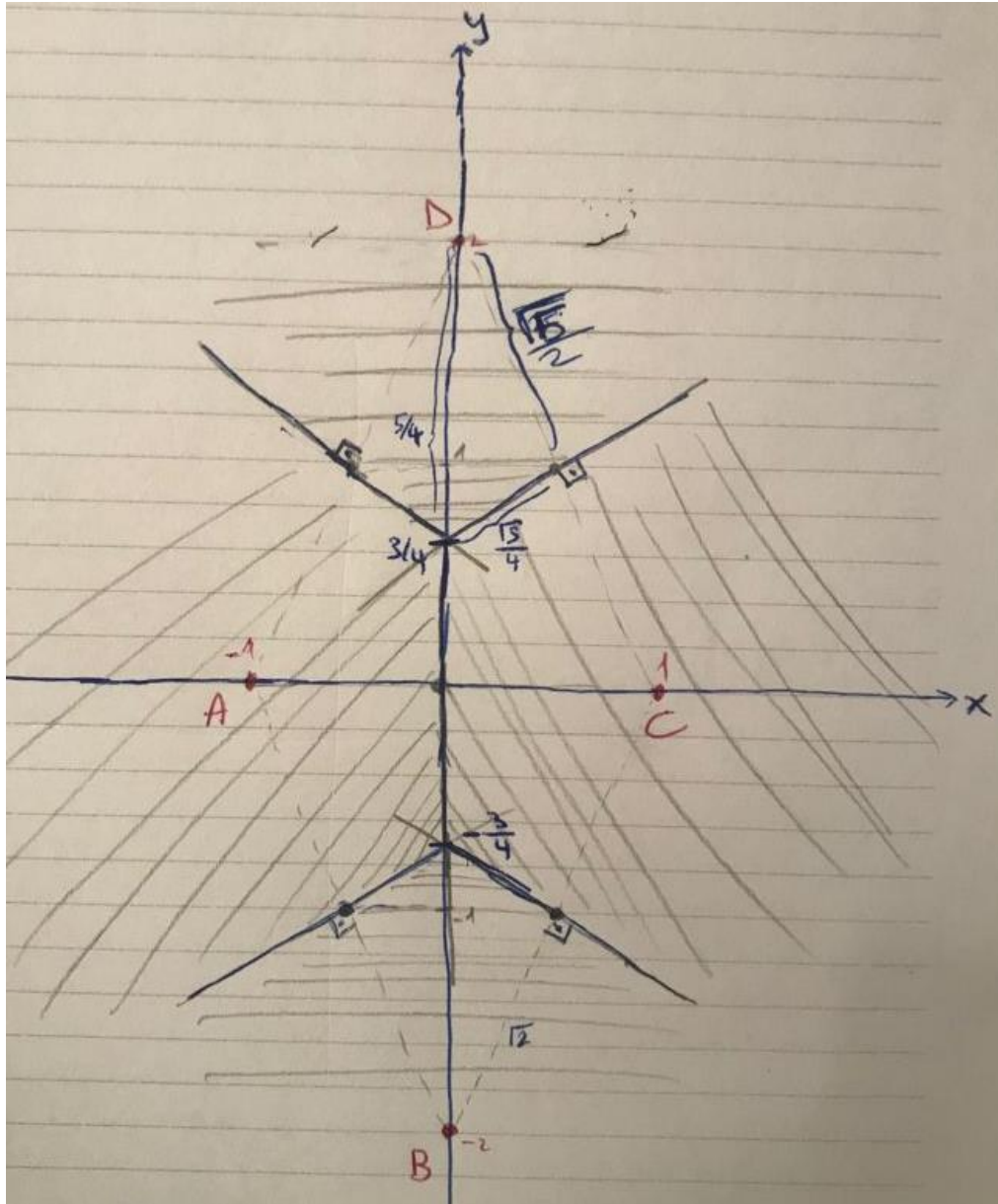


Figure 4: Voronoi diagram for points $A=[-1,0]$, $B=[0,-2]$, $C=[1,0]$, $D=[0,2]$

4. Optimal values for the grid are found as in Figure 5. Since discount factor is 0.5, the values = $0.5 * \max(\text{neighbours values})$.

And policy is found as action that gives optimum value. In this case, optimal policy is not unique because several grids have the same value.

The same result is also found in 3 iterations by python script (Figure 7).

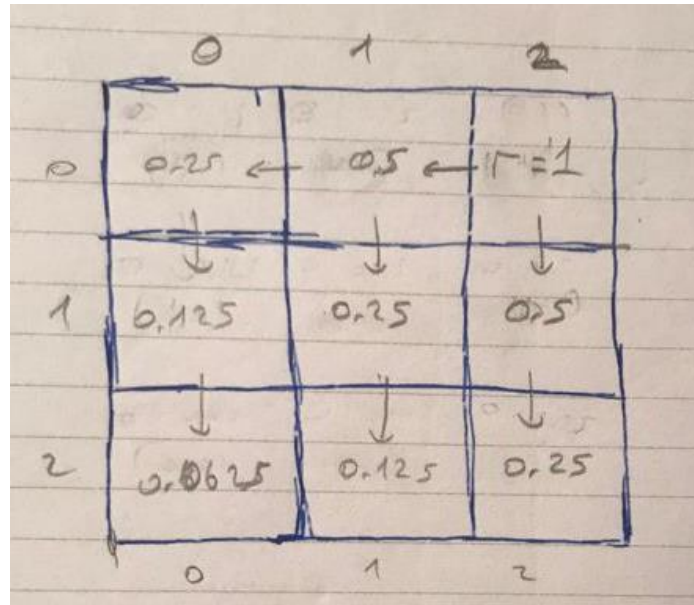


Figure 5: Optimal values

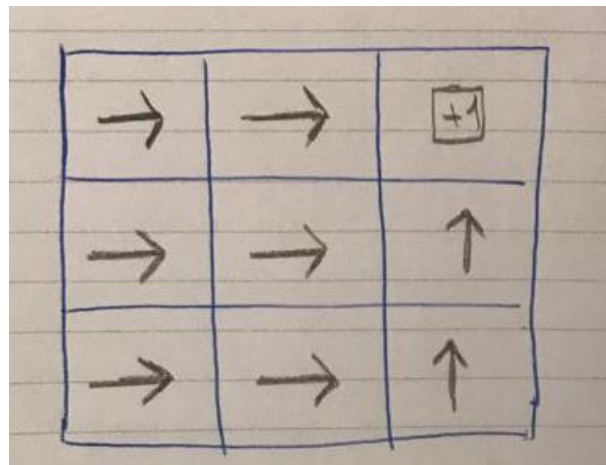


Figure 6: Optimal policy (actions)

```
iteration number 2
optimal policy is
{(0, 0): 'R', (0, 1): 'R', (1, 0): 'R', (1, 1): 'R', (1, 2): 'U', (2, 0): 'R', (2, 1): 'R', (2, 2): 'U'}

Value of state (0, 0) is 0.25
Value of state (0, 1) is 0.5
Value of state (0, 2) is 1
Value of state (1, 0) is 0.125
Value of state (1, 1) is 0.25
Value of state (1, 2) is 0.5
Value of state (2, 0) is 0.0625
Value of state (2, 1) is 0.125
Value of state (2, 2) is 0.25
```

Figure 7: Optimal values and policy found by Python script

5. The results are shown in Figure 8 and Figure 9. Matlab script is attached.

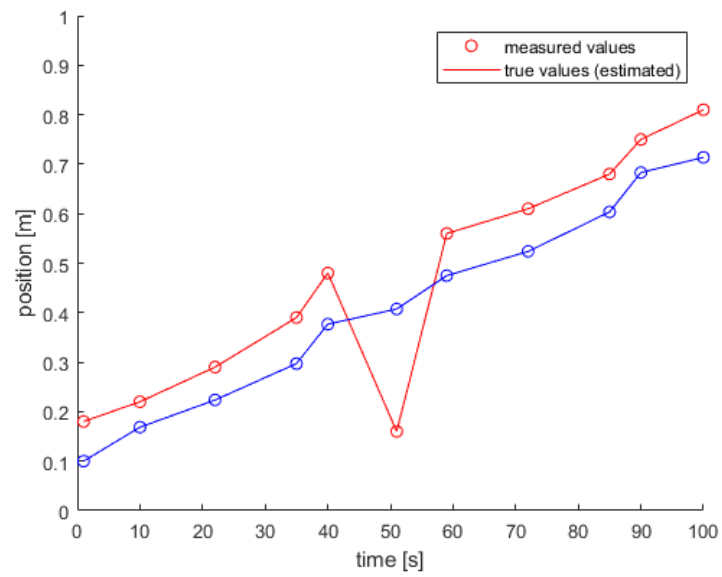


Figure 8: Measurements and Estimations of Position

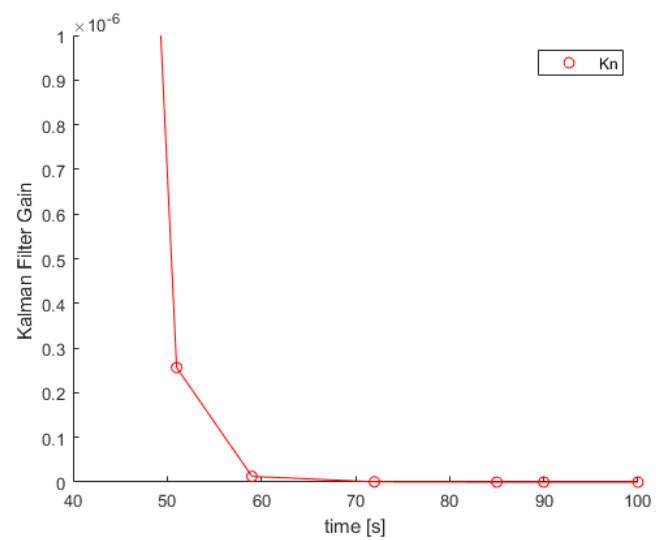


Figure 9: Kalman Filter Gain Values