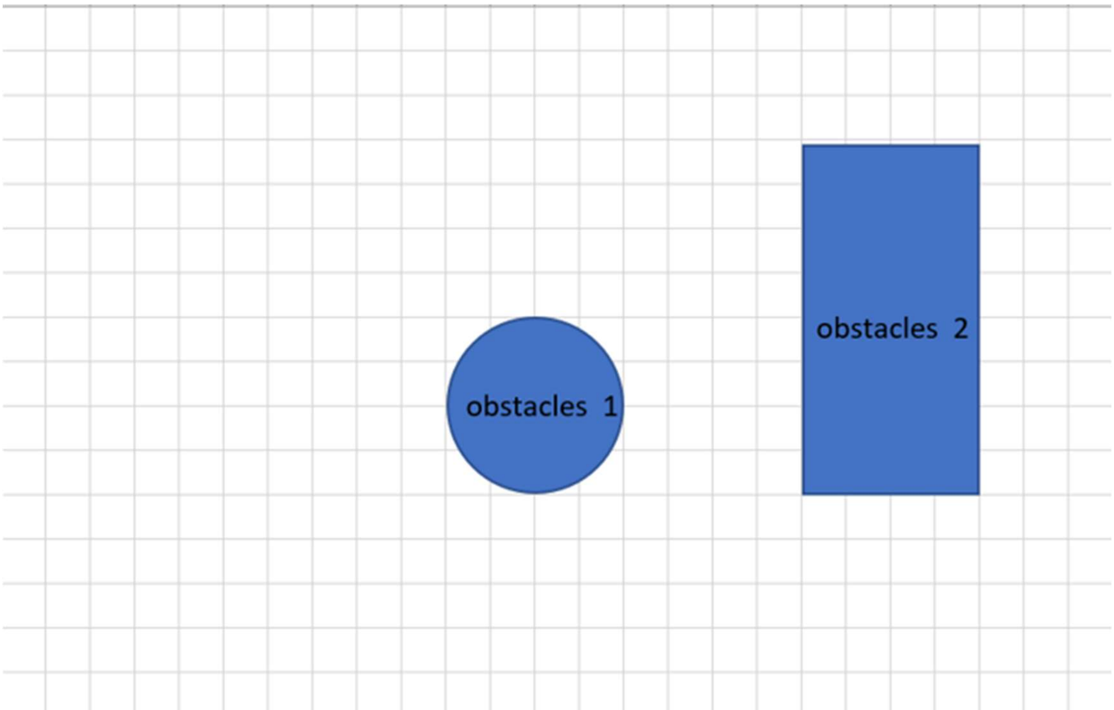
EE5110

Special Topics in Automation and Control

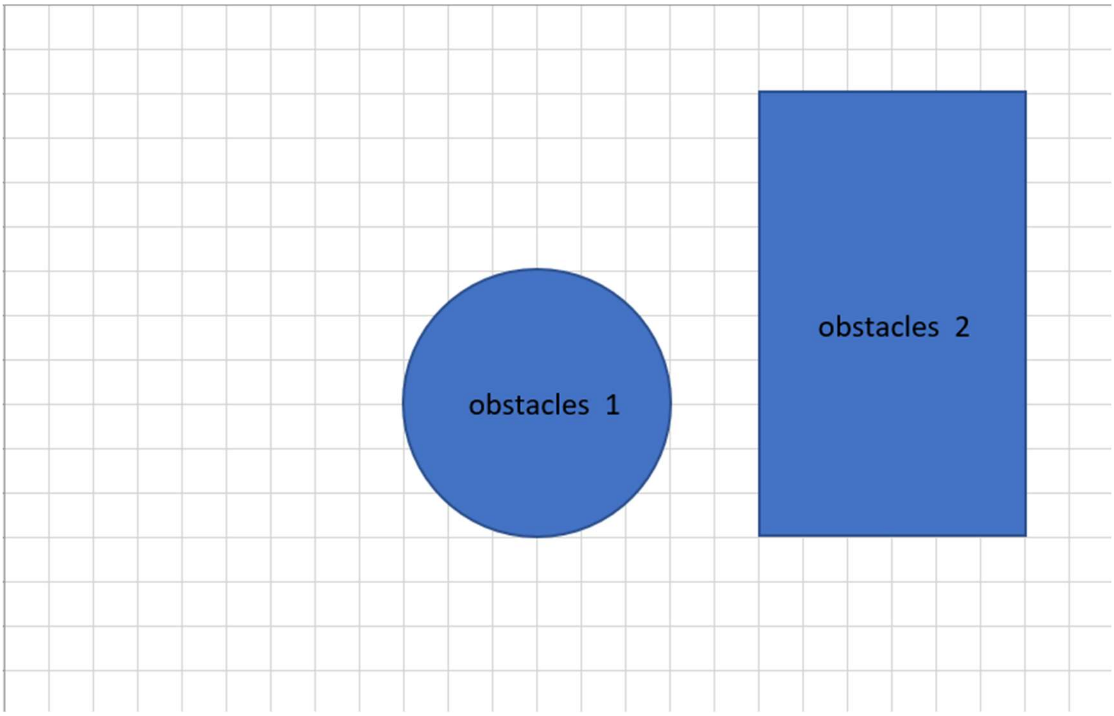
Autonomous Systems

Student ID: A0232935A
Name: Liu Weihao

1. C-Space:



This is the work space, then we need to expand the obstacles with robot's radius=1 unit. The C-space is as follows:



Obstacles are expressed in C-space:

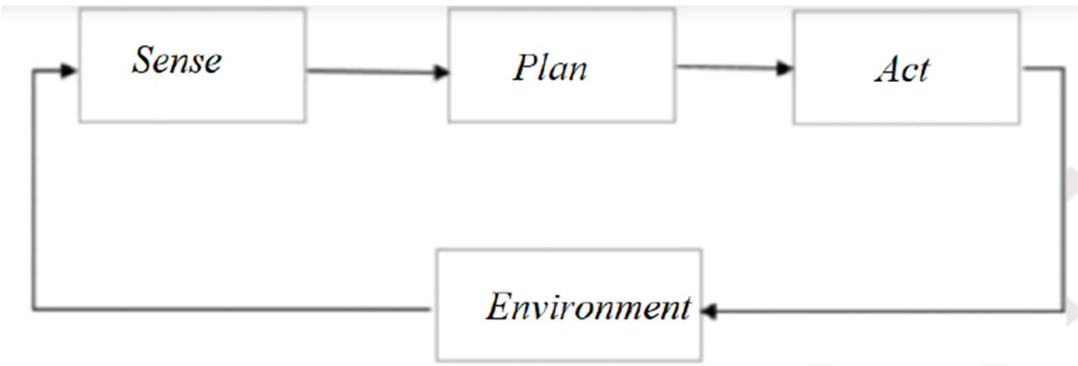
$$
\begin{aligned}
C_{obstacle1} = \{&M(10,5), M(11,5), M(12,5), M(13,5), M(14,5), M(15,5)\\
&M(10,6), M(11,6), M(12,6), M(13,6), M(14,6), M(15,6)\\
&M(10,7), M(11,7), M(12,7), M(13,7), M(14,7), M(15,7)\\
&M(10,8), M(11,8), M(12,8), M(13,8), M(14,8), M(15,8)\\
&M(10,9), M(11,9), M(12,9), M(13,9), M(14,9), M(15,9)\\
&M(10,10), M(11,10), M(12,10), M(13,10), M(14,10), M(15,10)\}
\end{aligned}
$$

$$
\begin{aligned}
C_{obstacles2} = \{&M(18,5), M(19,5), M(20,5), M(21,5), M(22,5), M(23,5),\\
&M(18,6), M(19,6), M(20,6), M(21,6), M(22,6), M(23,6),\\
&M(18,7), M(19,7), M(20,7), M(21,7), M(22,7), M(23,7),\\
&M(18,8), M(19,8), M(20,8), M(21,8), M(22,8), M(23,8),\\
&M(18,9), M(19,9), M(20,9), M(21,9), M(22,9), M(23,9),\\
&M(18,10), M(19,10), M(20,10), M(21,10), M(22,10), M(23,10),\\
&M(18,11), M(19,11), M(20,11), M(21,11), M(22,11), M(23,11),\\
&M(18,12), M(19,12), M(20,12), M(21,12), M(22,12), M(23,12),\\
&M(18,13), M(19,13), M(20,13), M(21,13), M(22,13), M(23,13),\\
&M(18,14), M(19,14), M(20,14), M(21,14), M(22,14), M(23,14)\}
\end{aligned}
$$

2. How many the classes in the motion planning architecture have we learnt from this module? Please draw block diagrams to illustrate each class and give your explanations.

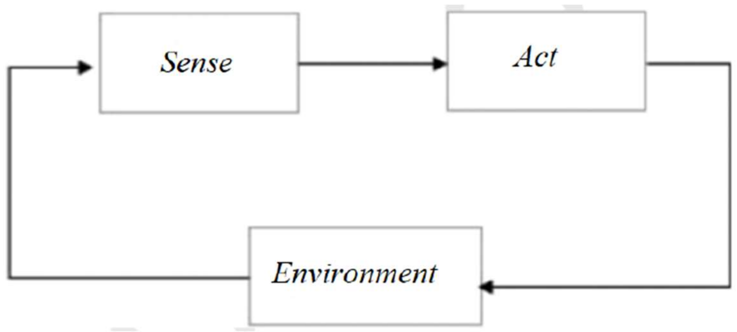
The motion planning can be grouped into three classes: deliberative planning architecture, reactive architecture, and hybrid architecture.

Deliberative:



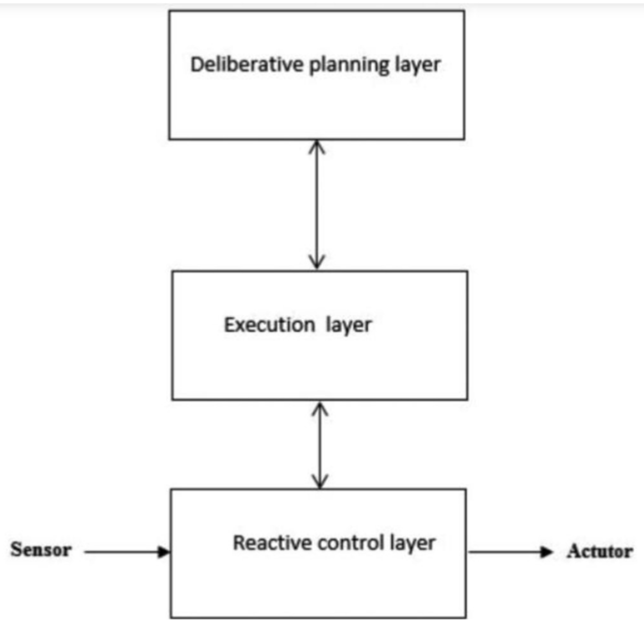
Deliberative control relies on reasoning on state information. First it needs the as much map information as possible. After that, we can compute the optimal path and follow the planned path.

Reactive architecture:



Different with deliberative, reactive architecture doesn't plan the path, it direct coupling of perception and control action.

Hybrid architecture:



Deliberative planning and reactive execution are involved with different activities, time scales, and spatial scope.
Planning configures and sets parameters for the reactive control system.
Execution occurs only due to reactive system.
Planning occurs prior to and concurrent to reactive system.

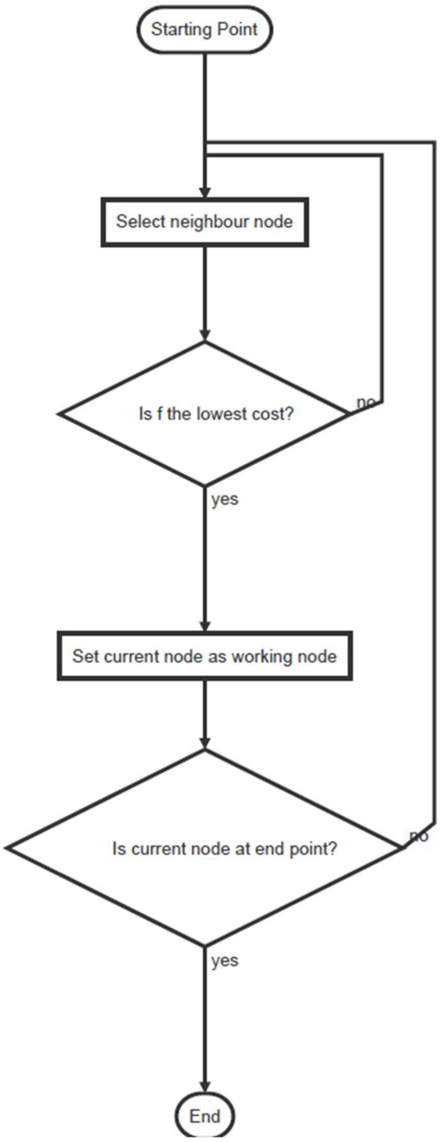
3. What is the major difference between Dijkstra and A* search algorithms? According to your understanding, please explain it.

Both Dijkstra and A* can find the optimal path.
Dijkstra don't know where the end point is. It starts at the starting point, all directions are complete equivalent. At each step, it will choose the closest route to the starting point. This algorithm will search the whole map, so it's very slow.

A* contain two parts, one is the cost of moving from the starting point to the current node (same as Dijkstra), and another is the estimated path between current node to the destination node. It can be expressed as: $F = G + H$. So, A* can find the optimal path because it has G part, and the whole progress is very fast, because it has H part.

4.

- Draw a flow chart of A* algorithm for a general case and explain it briefly (not only for this question),



- Write down a detailed process for finding the shortest path (Don't use the A* codes to find the shortest path and you may follow the lecture's steps).

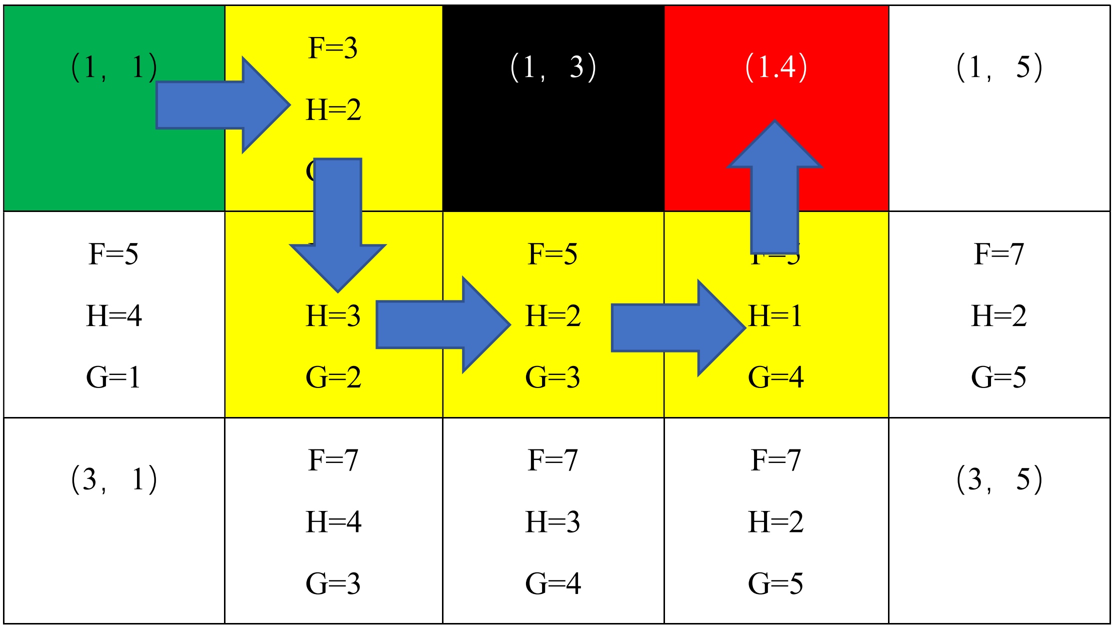
(1, 1)	F=3 H=2 G=1	(1, 3)	(1.4)	(1, 5)
F=5 H=4 G=1	(2, 2)	(2, 3)	(2, 4)	(2, 5)
(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)

(1, 1)	F=3 H=2 G=1	(1, 3)	(1.4)	(1, 5)
F=5 H=4 G=1	F=5 H=3 G=2	(2, 3)	(2, 4)	(2, 5)
(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)

（1， 1）	F=3 H=2 G=1	（1， 3）	（1.4）	（1， 5）
F=5 H=4 G=1	F=5 H=3 G=2	F=5 H=2 G=3	（2， 4）	（2， 5）
（3， 1）	F=7 H=4 G=3	（3， 3）	（3， 4）	（3， 5）

（1， 1）	F=3 H=2 G=1	（1， 3）	（1.4）	（1， 5）
F=5 H=4 G=1	F=5 H=3 G=2	F=5 H=2 G=3	F=5 H=1 G=4	（2， 5）
（3， 1）	F=7 H=4 G=3	F=7 H=3 G=4	（3， 4）	（3， 5）

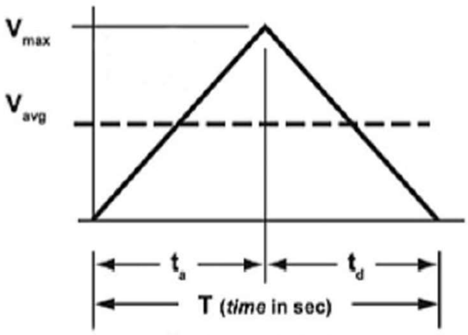
(1， 1)	F=3 H=2 G=1	(1， 3)	(1.4)	(1， 5)
F=5 H=4 G=1	F=5 H=3 G=2	F=5 H=2 G=3	F=5 H=1 G=4	(2， 5)
(3， 1)	F=7 H=4 G=3	F=7 H=3 G=4	(3， 4)	(3， 5)

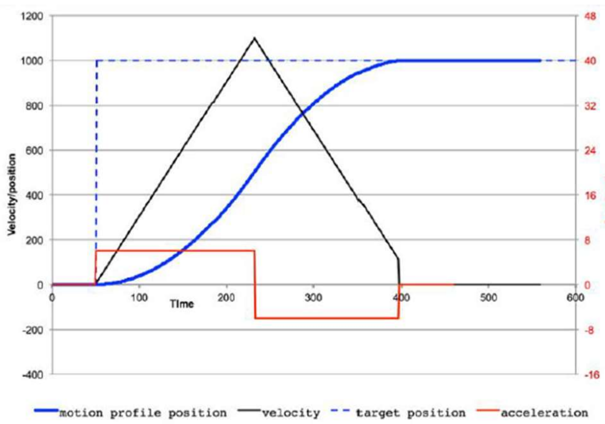


5. Literature review regarding trajectory generation in autonomous system. In the literature review, please present existing several methods in trajectory generation including typical references (it is better to limit body texts under 1200 words).

1) Triangular motion profile

The basic premise of a triangular move profile is to accelerate to a maximum velocity, then immediately decelerate, with acceleration and deceleration being equal in terms of both time and distance. In other words, if you want to move from here to there in the fastest time, you would use a triangular move profile. Determining any of the move variables time, velocity, acceleration, or distance is based on the geometry of the triangle. Below are some common equations for triangular move profiles.





Not very smooth but allows for easy incorporation of joint velocity and acceleration limits.

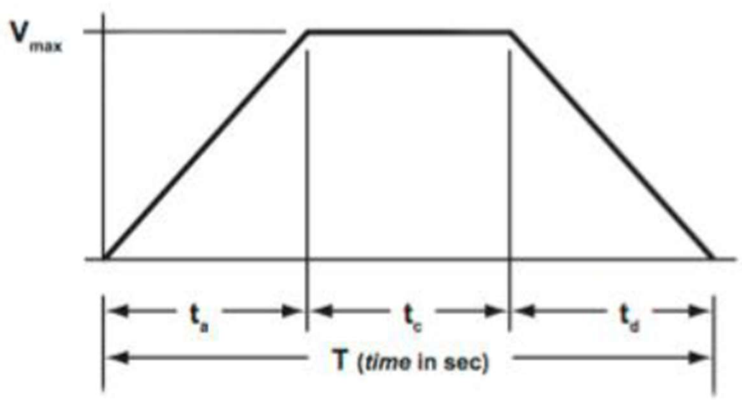
Limitations:

(a) Overall fastest option for getting from one point to the other,

(b) Not smooth

2) Trapezoidal motion profiles

A trapezoidal move profile is used when the application needs to accelerate to a maximum velocity and then travel at that velocity for a specified time or distance. Some common processes that use trapezoidal move profiles are machining, dispensing, and painting.



The simplest form of trapezoidal move profile, and the one used in the examples below, is the 1/3, 1/3, 1/3 profile. In this case, 1/3 of the time is used for accelerating, 1/3 is used for constant velocity, and 1/3 is used for decelerating. But if you understand the geometry of the move profile, which is essentially two triangles and a rectangle, you can calculate the necessary parameters regardless of whether the time segments are equal or not.

Advantages:

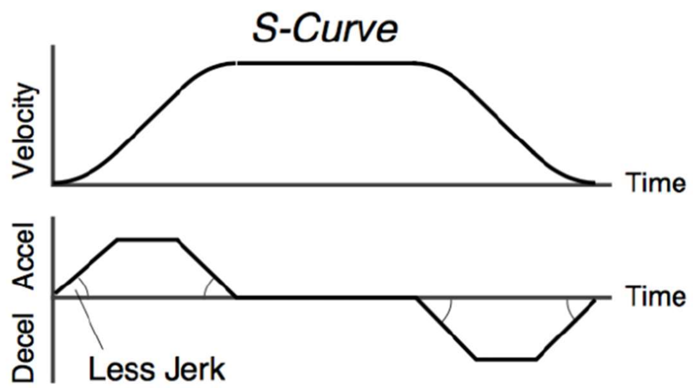
For applications that benefit from a period of constant velocity, a trapezoidal move profile offers the advantage of having a lower maximum velocity than a triangular profile.

Limitations:

Trapezoidal profile causes discontinuities in acceleration at t=0, ta, ta+tc and T.

3) S-cure

Despite underpinning the majority of motion control applications, neither triangular nor trapezoidal move profiles are ideal for motion systems due to a phenomenon known as “jerk.” Jerk is the rate of change of acceleration, and for trapezoidal and triangular move profiles, the initial acceleration and final deceleration occur instantly, meaning jerk is (theoretically) infinite.



6.

1) give all six coefficients

$$x(t) = p_0 + p_1 t + p_2 t^2 + p_3 t^3 + p_4 t^4 + p_5 t^5$$
$$\dot{x}(t) = p_1 + 2p_2 t + 3p_3 t^2 + 4p_4 t^3 + 5p_5 t^4$$
$$\ddot{x}(t) = 2p_2 + 6p_3 t + 12p_4 t^2 + 20p_5 t^3$$

Then we can get:

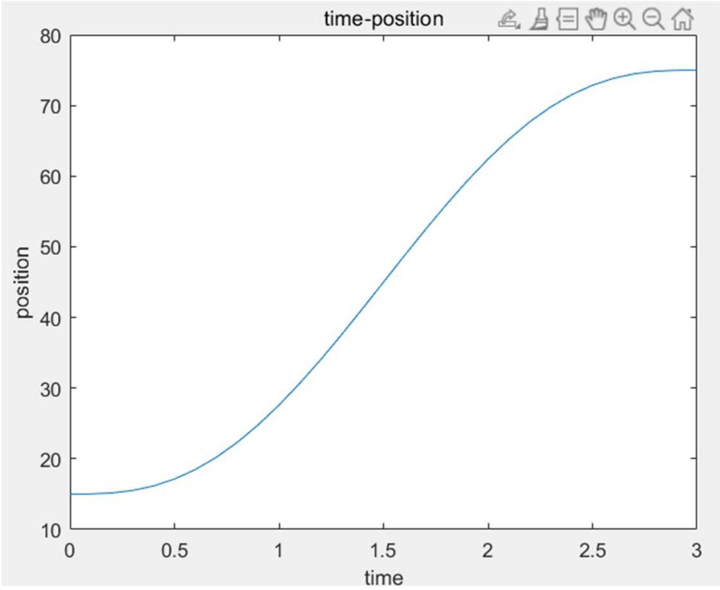
$$\begin{bmatrix} 15 \\ 75 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix}$$

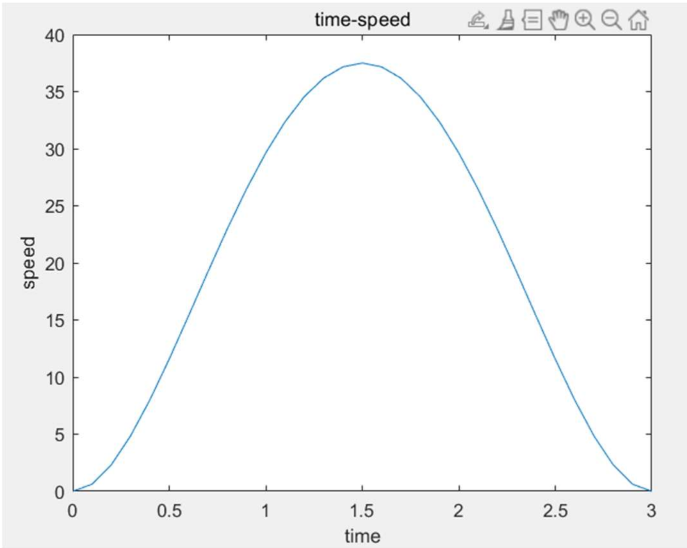
$$\begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} \frac{360}{T^5} \\ -\frac{900}{T^4} \\ \frac{600}{T^3} \\ 0 \\ 0 \\ 15 \end{bmatrix}$$

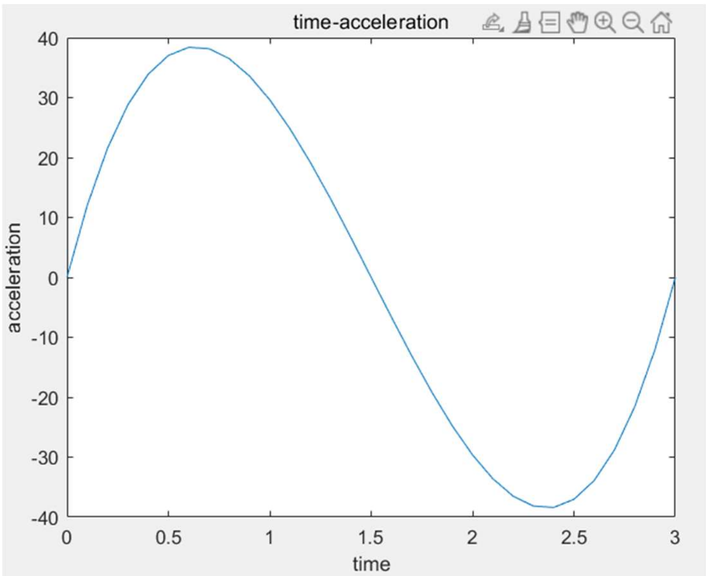
Because $T = 3s$, so we can get:

$$\begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} 1.4815 \\ -11.11 \\ 22.22 \\ 0 \\ 0 \\ 15 \end{bmatrix}$$

2) plot position, speed, and acceleration profiles


time-position
position
time
80
70
60
50
40
30
20
10
0
0.5
1
1.5
2
2.5
3


time-speed
speed
time
40
35
30
25
20
15
10
5
0
0
0.5
1
1.5
2
2.5
3


time-acceleration
acceleration
time
40
30
20
10
0
-10
-20
-30
-40
0
0.5
1
1.5
2
2.5
3