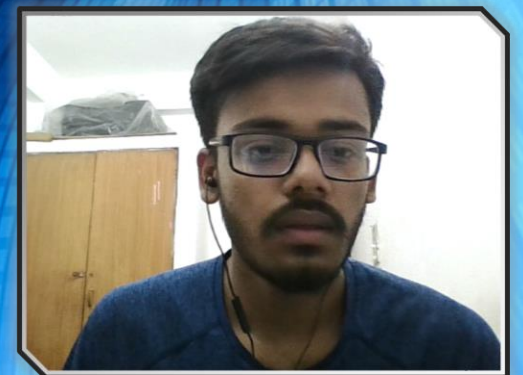


Inverse Kinematics of Three DOF Planar Robot

TO: ROSHAN KUMAR HOTA SIR, ASSISTANT PROFESSOR,
DEPARTMENT OF MECHANICAL ENGINEERING

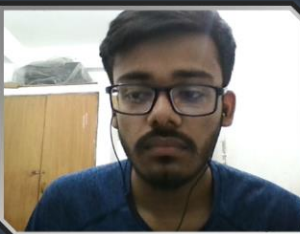
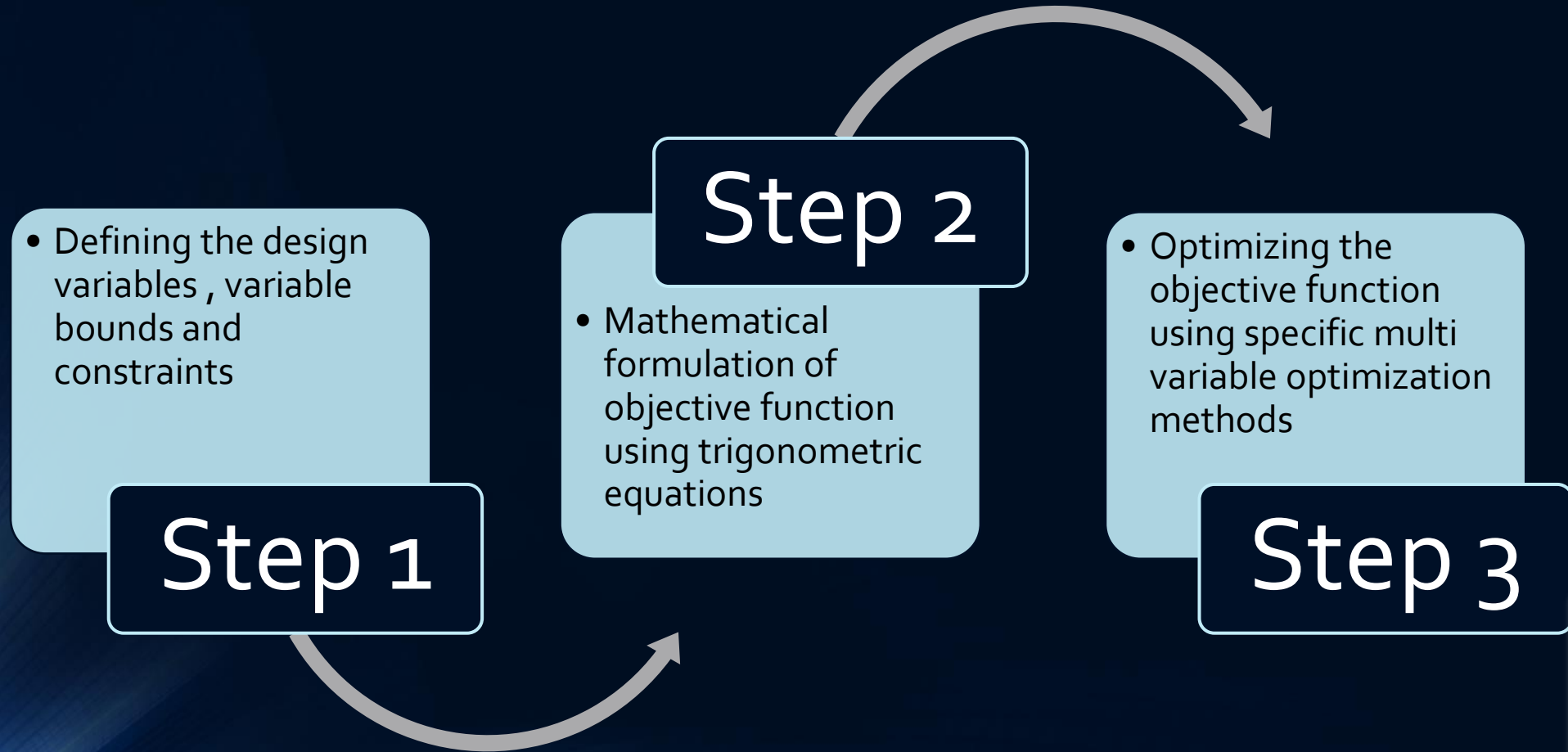


TEAM MEMBERS:

- TANMAY TRIPATHI (20226152)
- KESHAV KRISHNA SHARMA (20226075)
- MANUDEV VERMA (20226082)



APPROACH OF THE PROBLEM :

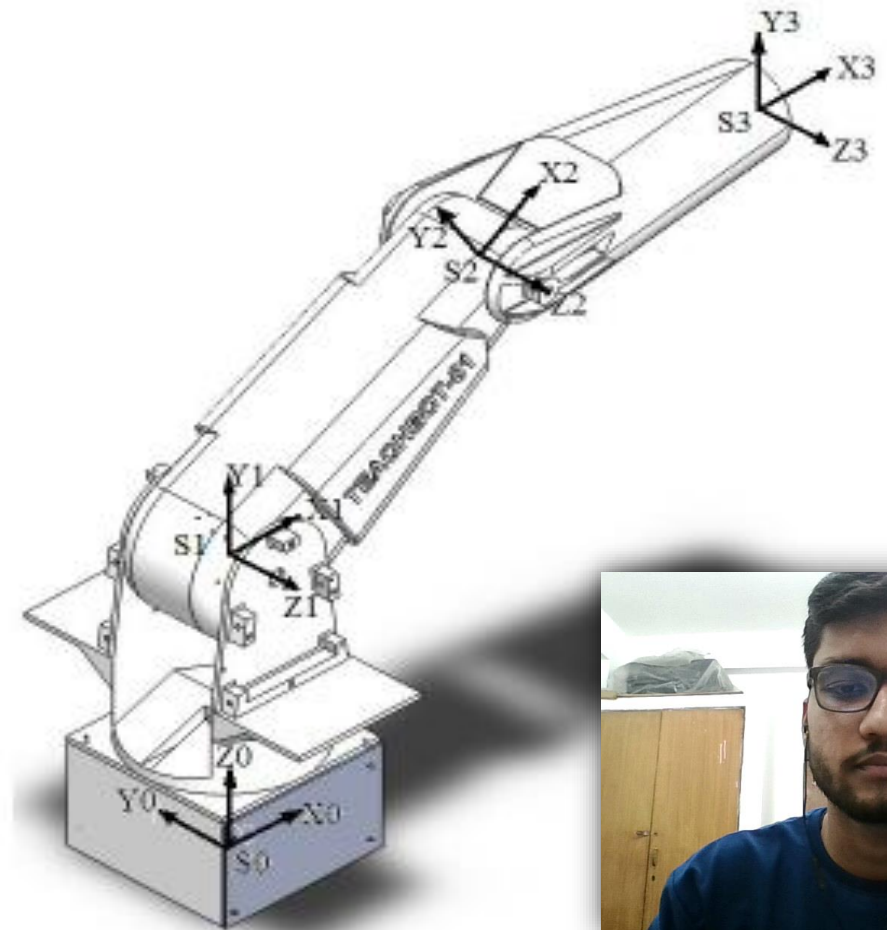
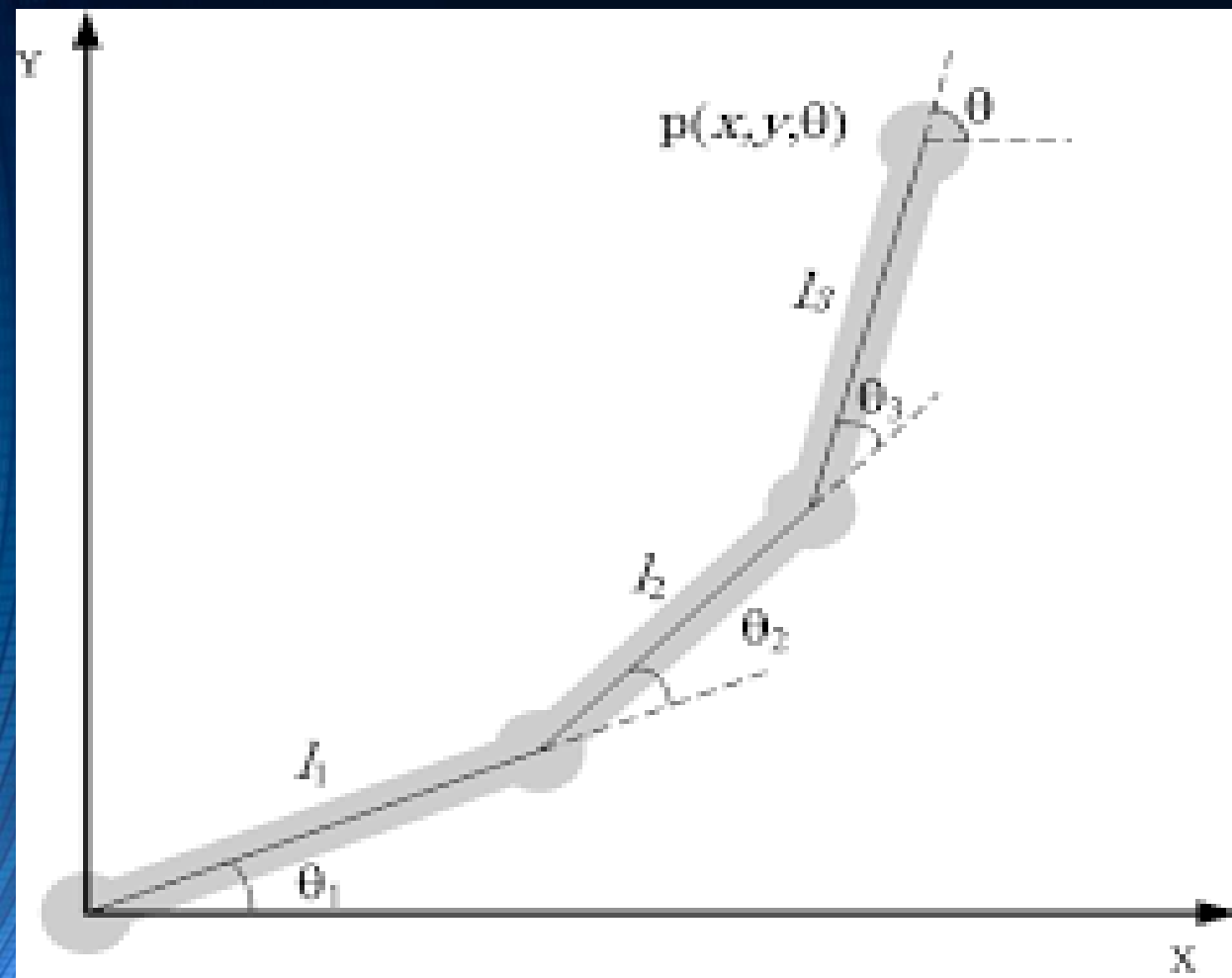


INTRODUCTION:



Inverse kinematics (IK) is a method of solving the joint variables when the end-effector position and orientation (relative to the base frame) of a 3R planar Robot and all the geometric link parameters are known. Optimizing the inverse kinematics of a 3R planar robot involves enhancing computational efficiency while ensuring accurate positioning of end-effectors. One approach is to leverage geometric techniques, such as the law of cosines and the geometric interpretation of the robot's kinematic equations, to simplify the calculation process. Additionally, utilizing numerical methods like iterative algorithms (e.g., Newton-Raphson) can refine the solutions iteratively, enhancing precision. Employing optimization algorithms such as genetic algorithms or particle swarm optimization can further fine-tune parameters for optimal performance, considering constraints such as joint limits and workspace boundaries. Moreover, incorporating efficient data structures and parallel processing techniques can expedite computations, crucial for real-time applications. By integrating these methodologies, the inverse kinematics of a 3R planar robot can be optimized to achieve a balance between computational efficiency and accuracy, facilitating smoother and more precise motion control in various applications.

DIAGRAMS



In above planar figure , let:

$$x_1 = l_1 \cos \Theta_1 + l_2 \cos(\Theta_1 + \Theta_2) + l_3 \cos(\Theta_1 + \Theta_2 + \Theta_3)$$

$$y_1 = l_1 \sin \Theta_1 + l_2 \sin(\Theta_1 + \Theta_2) + l_3 \sin(\Theta_1 + \Theta_2 + \Theta_3)$$

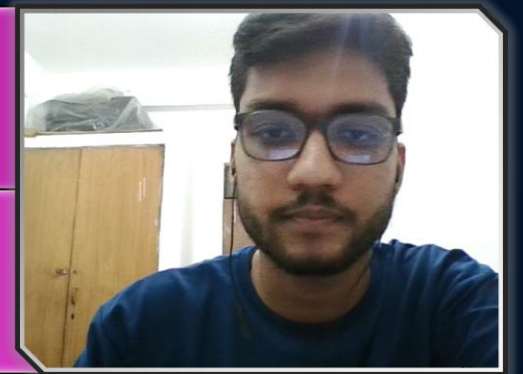
Hence objective function will be;

$$F(\Theta_1, \Theta_2, \Theta_3) = [x - x_1]^2 + [y - y_1]^2$$

Where the design variables $\Theta_1, \Theta_2, \Theta_3$ are angles as shown in figure and l_1, l_2 and l_3 are lengths of robotic arms and (x_1, y_1) is the end effector position

Constraints :

$$0 < \Theta_1, \Theta_2, \Theta_3 < 360^\circ$$



Explanation

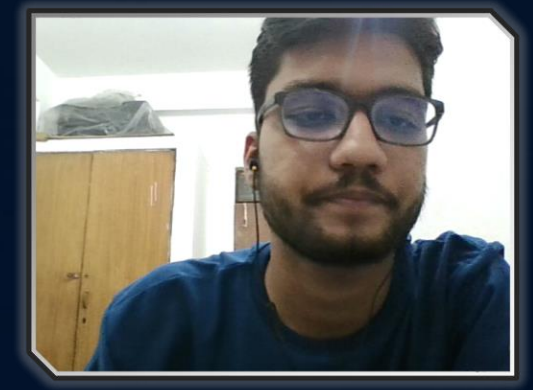
After formulating we get our objective function as:

$$F(\Theta_1, \Theta_2, \Theta_3) = [x - x_1]^2 + [y - y_1]^2$$

Where ,

$$x_1 = l_1 \cos \Theta_1 + l_2 \cos(\Theta_1 + \Theta_2) + l_3 \cos(\Theta_1 + \Theta_2 + \Theta_3)$$

$$y_1 = l_1 \sin \Theta_1 + l_2 \sin(\Theta_1 + \Theta_2) + l_3 \sin(\Theta_1 + \Theta_2 + \Theta_3)$$



The concept behind formulating the above objective function is that the $F(\Theta_1, \Theta_2, \Theta_3)$ will be minimum when both the square quantities (that are $[x - x_1]^2$ and $[y - y_1]^2$) are zero (i.e. when $x = x_1$ and $y = y_1$, which are the desired end effector points). The idea behind squaring the difference of desired point and end effector point is that the function formed will be the square of Euclidean distance and minimizing it will reduce the error between the two quantities. Therefore we can use the multivariable optimization algorithm known as Cauchy steepest descent along with analytical univariable search method to find the minima of our objective function.

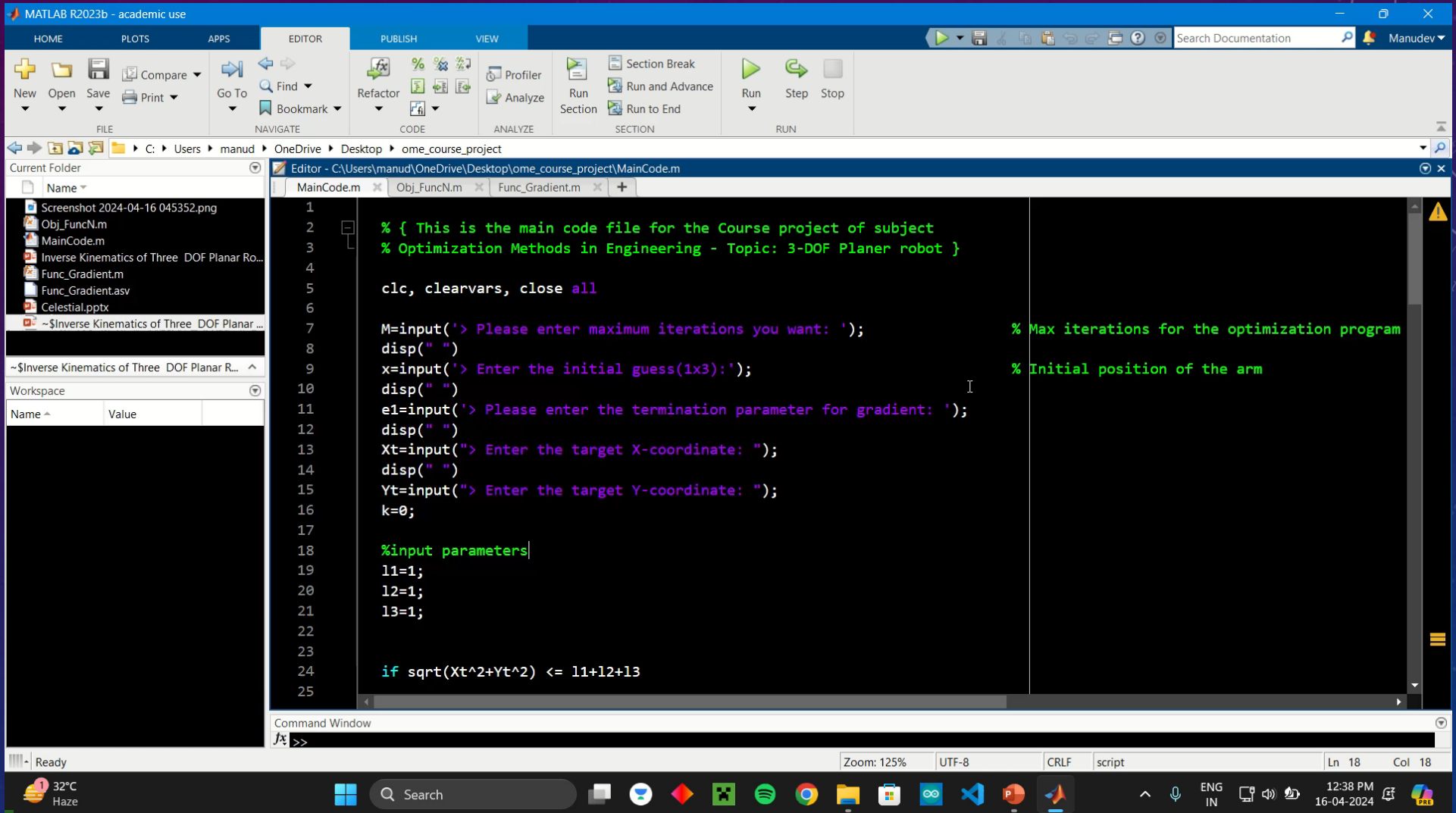
--- [MAIN CODE] ---

[THIS SECTION CONSISTS MAINLY OF THE CODE AND PROCESSING]

[WORK FOR THE OPTIMIZATION]



---[CODE EXPLANATION]---

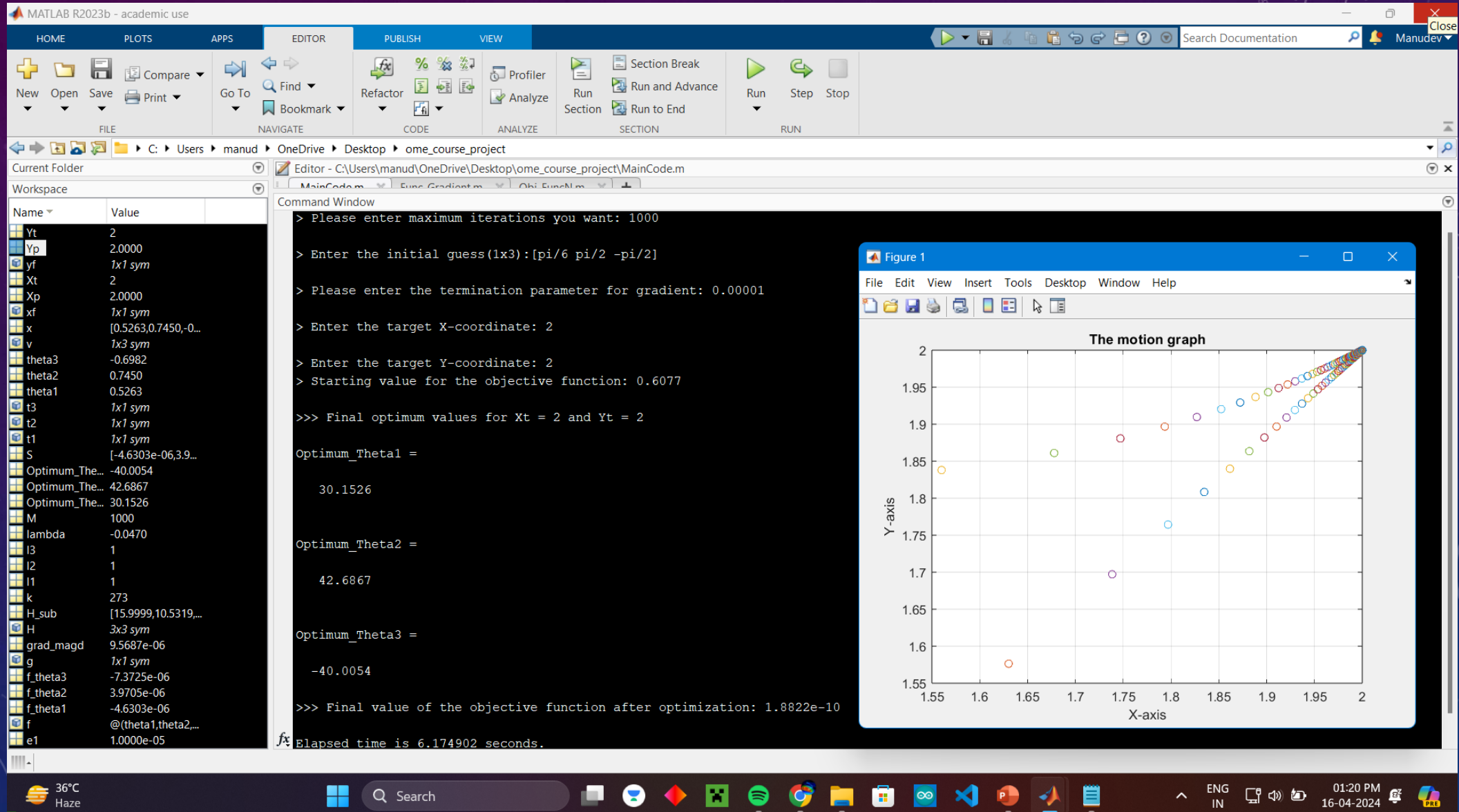


The image shows the MATLAB R2023b - academic use interface. The main window displays the Editor with the file `MainCode.m` open. The code is written in MATLAB and includes comments in green and function calls in purple. The code is as follows:

```
1
2  %{ This is the main code file for the Course project of subject
3  % Optimization Methods in Engineering - Topic: 3-DOF Planer robot }
4
5  clc, clearvars, close all
6
7  M=input('> Please enter maximum iterations you want: '); % Max iterations for the optimization program
8  disp(" ")
9  x=input('> Enter the initial guess(1x3):'); % Initial position of the arm
10 disp(" ")
11 e1=input('> Please enter the termination parameter for gradient: ');
12 disp(" ")
13 Xt=input("> Enter the target X-coordinate: ");
14 disp(" ")
15 Yt=input("> Enter the target Y-coordinate: ");
16 k=0;
17
18 %input parameters
19 l1=1;
20 l2=1;
21 l3=1;
22
23
24 if sqrt(Xt^2+Yt^2) <= l1+l2+l3
25
```


The interface also shows the Command Window at the bottom with the prompt `>>`. The status bar at the bottom indicates the system is ready, with a temperature of 32°C and a haze condition. The taskbar shows various application icons, including the Start button, Search, and several open applications like Chrome, File Explorer, and MATLAB.

[SCREENSHOT OF A SAMPLE RUN]



REFERENCES:

- www.mathworks.com
- <https://in.mathworks.com/help/matlab/>
- <https://www.youtube.com/>
- Book: Optimization Methods in Engineering
- Kalyanmoy Deb

- Lambda value via analytical method : 
- www.google.com

Analytical Method:

How does one minimize a function in a search direction using an analytical method? It means S is fixed and you want to pick λ , the step length to minimize $f(x)$. Note $\Delta X_i = \lambda S_i$.

$$\begin{aligned} f(X_{i+1}) &= f(X_i + \lambda S_i) \\ &\cong f(X_i) + \nabla^T f(X_i)(\Delta X_i) + \frac{1}{2}(\Delta X_i)^T H(X_i)(\Delta X_i) \end{aligned}$$

Thus,

$$\frac{df(X_i + \lambda S_i)}{d\lambda} = 0$$

$$\Rightarrow \nabla f_i S_i + \lambda (S_i)^T H_i S_i = 0$$

$$\Rightarrow \lambda = -\frac{\nabla f_i S_i}{S_i^T H_i S_i} = \frac{S_i^T S_i}{S_i^T H_i S_i}$$

Expand by Taylor
series formula

$$\Rightarrow y = -\frac{z_1^T H_1 z_1}{z_1^T z_1} = \frac{z_1^T H_1 z_1}{z_1^T z_1}$$

The background is a gradient of deep blue and purple, speckled with white dots resembling a starry sky. Overlaid on this are several faint, light-colored geometric patterns. In the top right, there is a large circular scale with degree markings from 0 to 210 and concentric circles. In the bottom right, there are concentric circles with dashed lines and arrows indicating a clockwise direction. In the bottom left, there are also concentric circles with a dashed line and an arrow. In the top left, there is a small circular arc with an arrow.

THANK YOU!