TEXAS TECH UNIVERSITY.

CS 5331: Special problems in Computer Science: Aerial Computing

Summer II 2023

PROJECT: 3

Name: SubhashChandra Bose Goud Muchchunoor

R Number: R11850821

Project #3: Path planning strategies

Project requirements:

1. Simulate path planning strategies: (i) Random Planning (RAND), (ii) Distance-based Planning (NNF), and

(iii) Density-based Planning (DF). This project is to implement two major operations: Clustering and Path Planning. Suppose there are 100 points of interest (POIs) that are non-uniformly distributed in a 1,000 x 1,000 (m2) network. The locations of POIs follow a scale-free property, where a group of POIs is located within a small area. A drone equipped with a sensor flies over the network and scans all POIs, in which a coverage radius of the sensor (r) is set to 100 (m), coverage range. In this project, a source code of scale-free will be provided to generate POIs.

First, cluster POIs using a neighbor density-based clustering algorithm to efficiently cover all POIs. The basic idea is to group densely populated POIs in an early stage and remove the number of candidate POIs to cover quickly. Please see the technical paper.

Second, the drone launches from the base, located left and bottom in the network (0, 0), and flies toward the center locations (scan points) of each coverage range to scan all POIs. In this project, path planning is to determine the sequence of visits on all scan points. (i) In RAND, the drone first flies toward any randomly selected scan point. Upon arrival, the drone flies toward another randomly selected scan point. The drone repeats the same procedure until visits all scan points. (ii) In NNF, the drone selects a scan point that is the closest located to the drone. Upon arrival, the drone selects another closest located scan point from the current scan point. The drone repeats the same procedure until visits all scan points. (iii) in DF, the drone selects the scan point that covers the most POIs. Upon arrival, the drone selects the scan point that covers the second most POIs. The drone repeats the same procedure until visits all scan points.

For visualization, use a triangle and a solid line for the scan point and the drone path, respectively. For example,

2. Input parameters

- Path planning strategies: 0 for RAND, 1 for NNF, and 2 for DF

3. Run the clustering algorithm and path planning strategies.

>> path(0) % Clustering and RAND

>> path(1) % Clustering and NNF

>> path(2) % Clustering and DF

>> path(3) % The graph after performing the Neighbor density based clustering on the scale free points.

4. Measure the performance of each path planning strategy.

- Total flying distance from the base, left and bottom of the network (0, 0)

My analysis and work on the project:

Project goal: is to perform the **Neighbor Density based Clustering** on the scale free points generated from the given source code by the professor and write the code to apply different path planning strategies on the clustered scan points.

There are 3 different path planning strategies mentioned in the above given requirement are:

0. Random Planning (RAND)
1. Distance-based Planning (NNF)
2. Density-based Planning (DF)

and we have to implement the same using MATLAB code and find Total flying distance in meters from the base, left and bottom of the network (0,0).

Steps to run the program:

- Open the MATLAB application
- Save the code into a MATLAB function file with a .m extension.
- Make sure your code file is in the MATLAB working present directory
- Call the pro3_muchchunoor_subhashchandra function with the desired **path_planning_strategy_type** as an argument.
- The function takes an integer value as input, representing the path planning strategy type:

  - ✓ 0: Random Planning (RAND)
  - ✓ 1: Nearest Neighbor First Planning (NNF)
  - ✓ 2: Density-based First Planning (DF)
  - ✓ 3: Clustered Graph (Only plots the result after clustering and terminates)

Project execution & Result graphs:

Use case 0:

Code is run for the strategy=0 which is Random planning.

>>pro3_muchchunoor_subhashchandra(0);

Output & its explanation:

It has done the Neighbor Density based clustering algorithm on the scale free points which are indicated by the symbol ‘x’ in green color and it marked the coverage areas of radius 100m with doted circles as per the requirement and also marked the center of the coverage area as scan points with triangle symbol in the graph.

After the clustering is done, the drone starts from the network base(0,0) and flew through the randomly chosen clustered scan point. Then it finds another scan point based on the Random path planning strategy and drone flies to that. Same procedure repeats until drone reaches the last scan point.

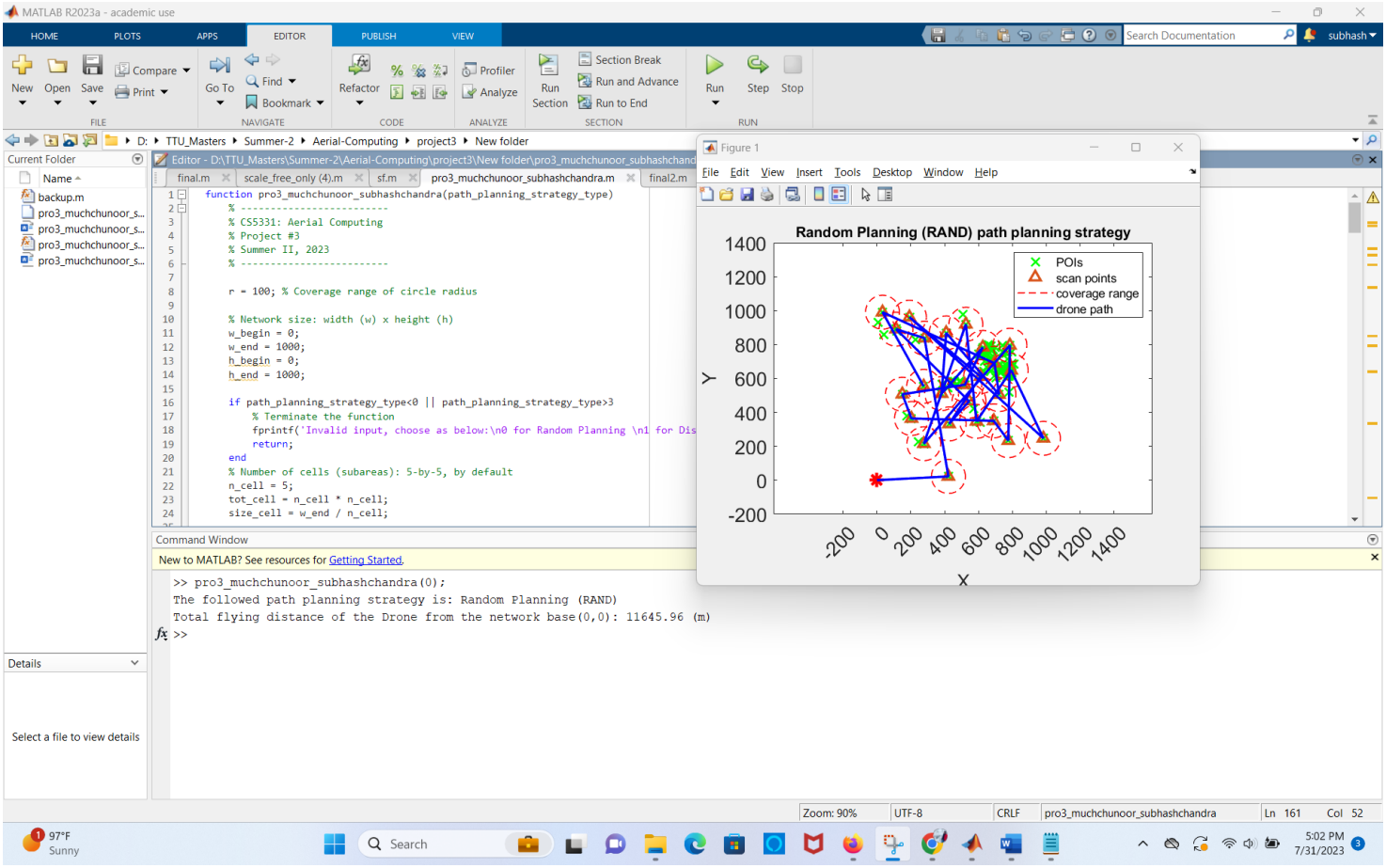
Drone path is indicated by the blue solid line in the graph.

In the final step, it prints the results as below:

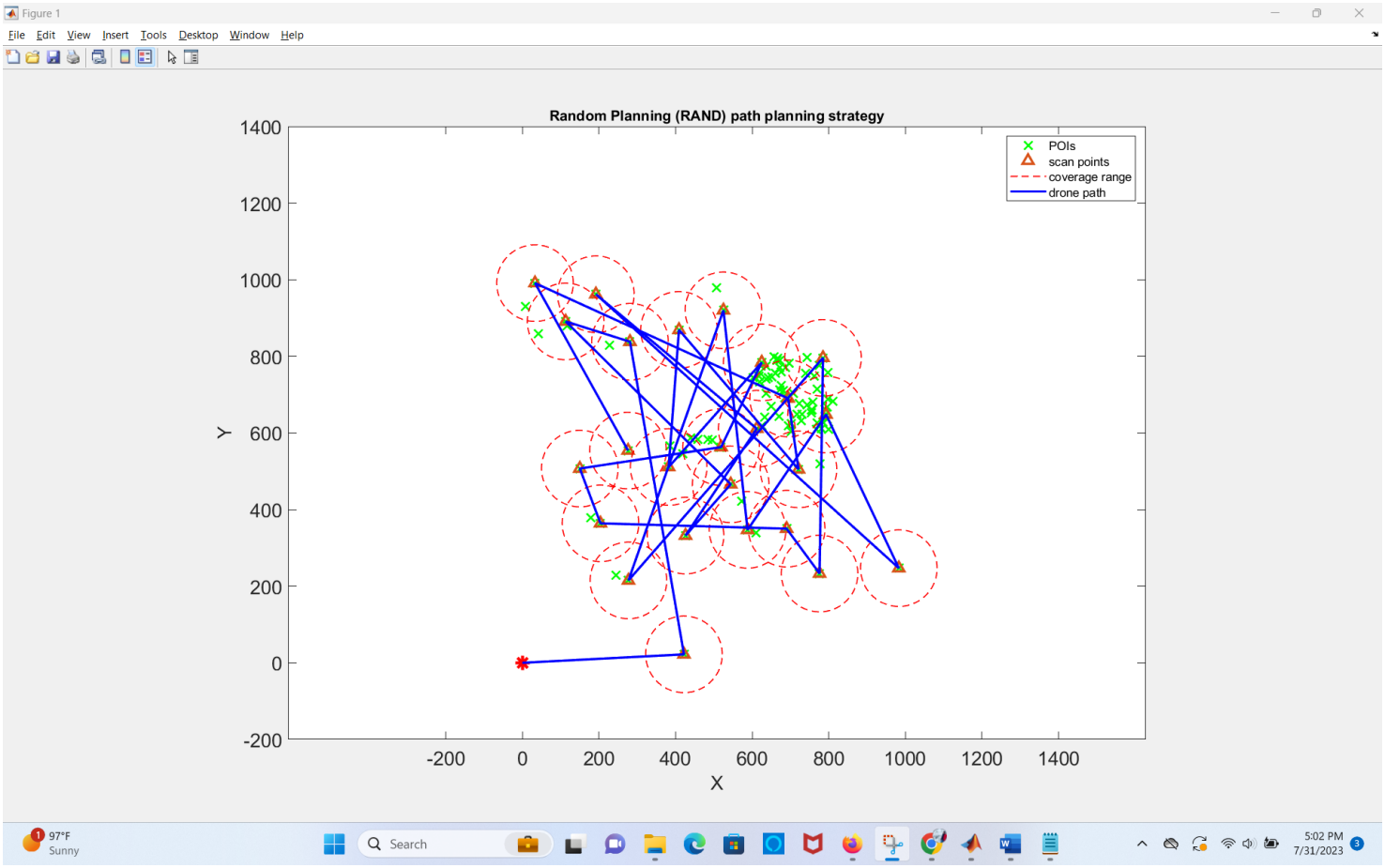
Displays the graph plotted based on the Random path (RAND) planning strategy as in the below screenshot.

The followed path planning strategy is: Random Planning (RAND)

The total flying distance of the drone from the network base(0,0) to last scan point is calculated and it is =11645.96 meters.



Full screen screenshot of the result for better view.



Use case 1:

Code is run for the strategy=1 which is Nearest Neighbor First Planning (NNF).

>>pro3_muchchunoor_subhashchandra(1);

Output & its explanation:

It has done the Neighbor Density based clustering algorithm on the scale free points which are indicated by the symbol ‘x’ in green color, and it marked the coverage areas of radius 100m with red color doted circles as per the requirement and also marked the center of the coverage area as scan points with triangle symbol in the graph.

After that drone started from the network base(0,0) and flew through the nearest clustered scan point first (*It uses the Nearest Neighbor First path planning to find the next nearest neighbor from the drone’s current position*), then it will search for the next nearest scan point from its current location and fly to that scan point. It repeats the same procedure until the drone retches the last scan point of triangle symbols.

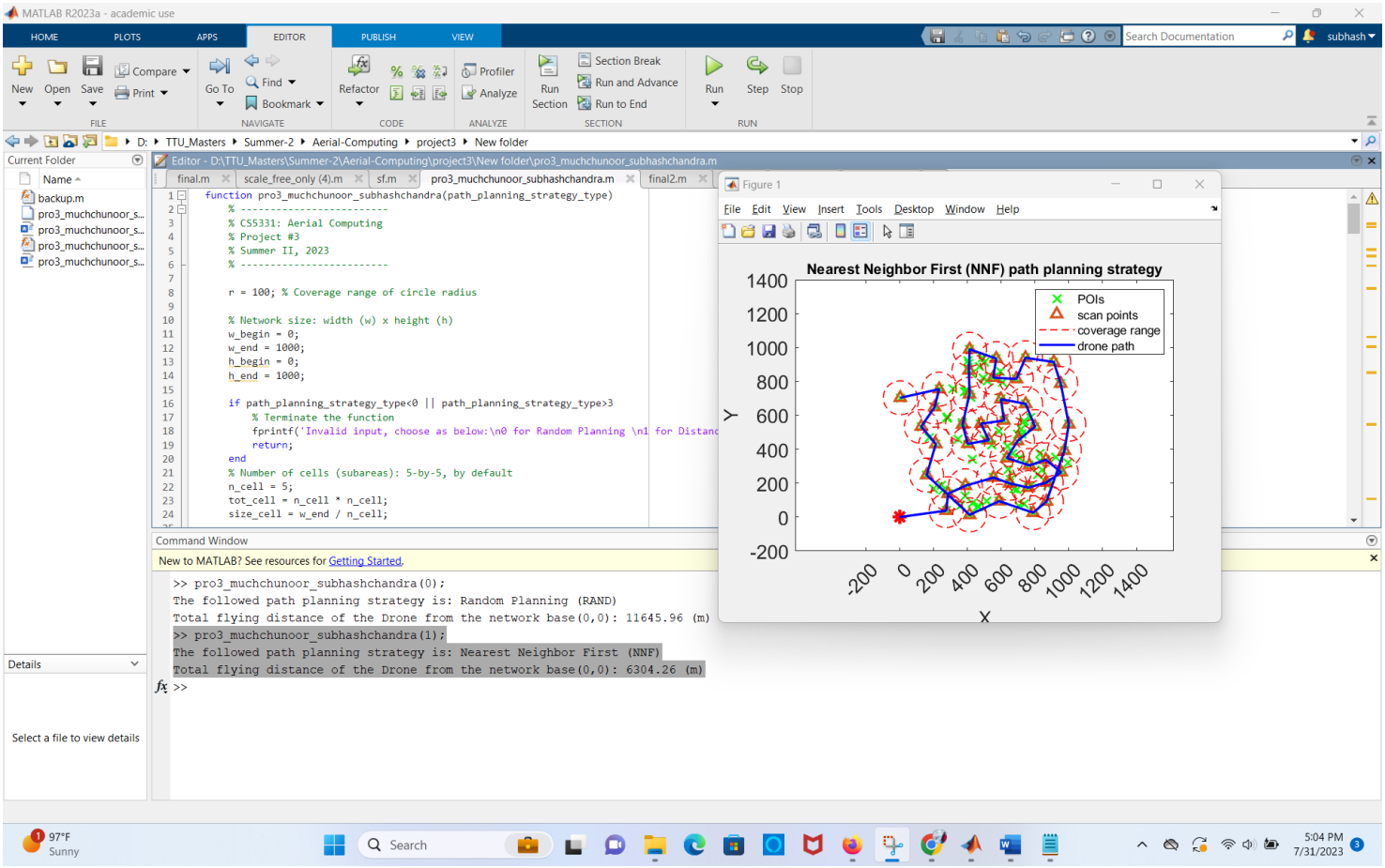
The drone path is indicated by the blue solid line in the graph.

In the final step, it prints the results as below:

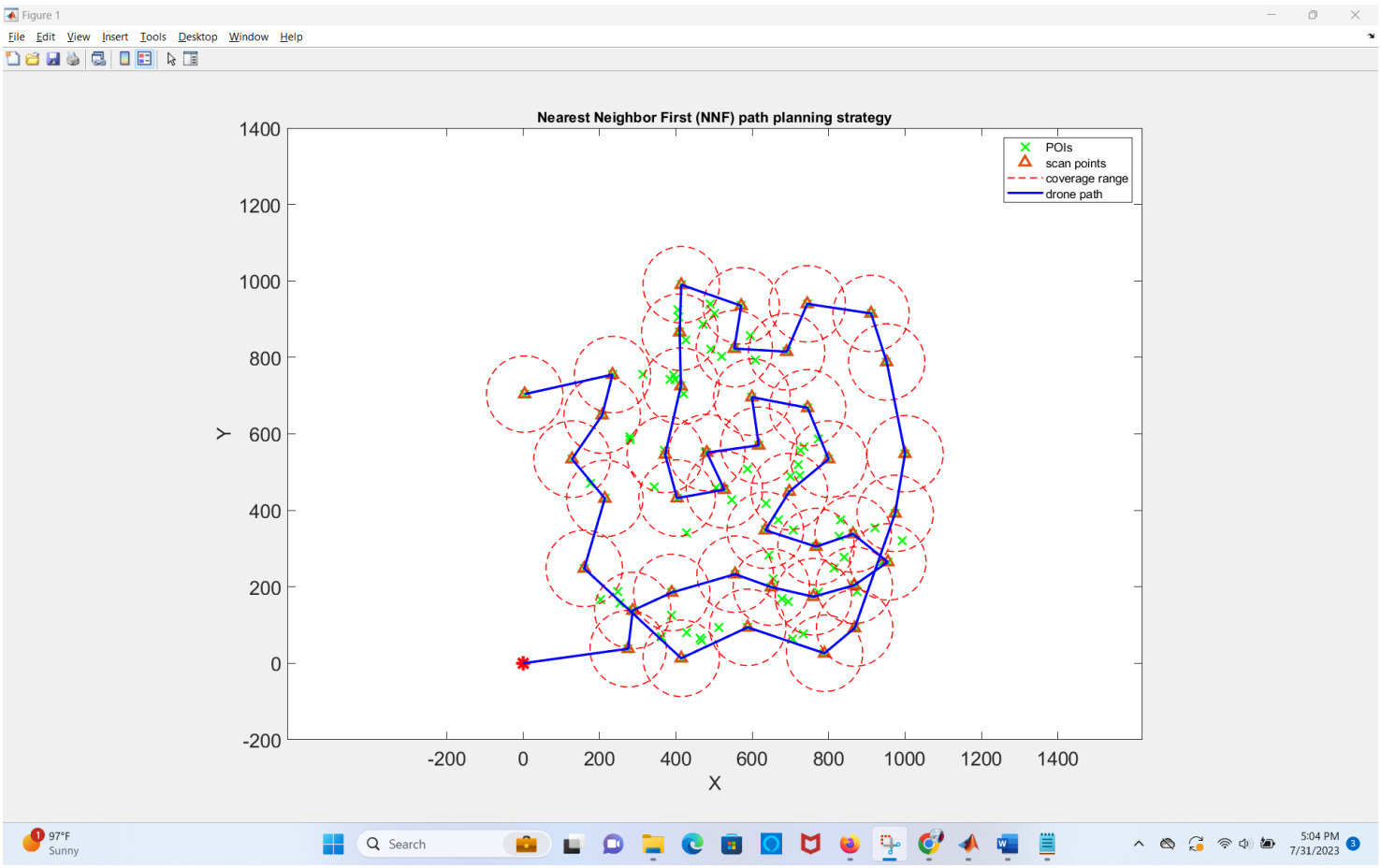
Displays the graph plotted based on the Nearest Neighbor First (NNF) path planning strategy as in the below screenshot.

The followed path planning strategy is: Nearest Neighbor First (NNF)

The total flying distance of the drone from the network base(0,0) to last scan point is calculated and it is =6304.26 meters.



Full screen screenshot of the result for better view.



Use case 2:

Code is run for the strategy=2 which is Density-based First (DF) path planning strategy.

>>pro3_muchchunoor_subhashchandra(2);

Output & its explanation:

It has done the Neighbor Density based clustering algorithm on the scale free points which are indicated by the symbol ‘x’ in green color, and it marked the coverage areas of radius 100m with red color doted circles as per the requirement and also marked the center of the coverage area as scan points with triangle symbol in the graph.

After clustering is done, drone starts flying from the network base(0,0) and flew to the scan point based on Density based First path planning (*it will search for the scan point of coverage area circle with more number of POIs in it.),* then it finds the next scan point based on DF path planning and fly to that scan point. It repeats the same procedure until the drone retches the last scan point of triangle symbols.

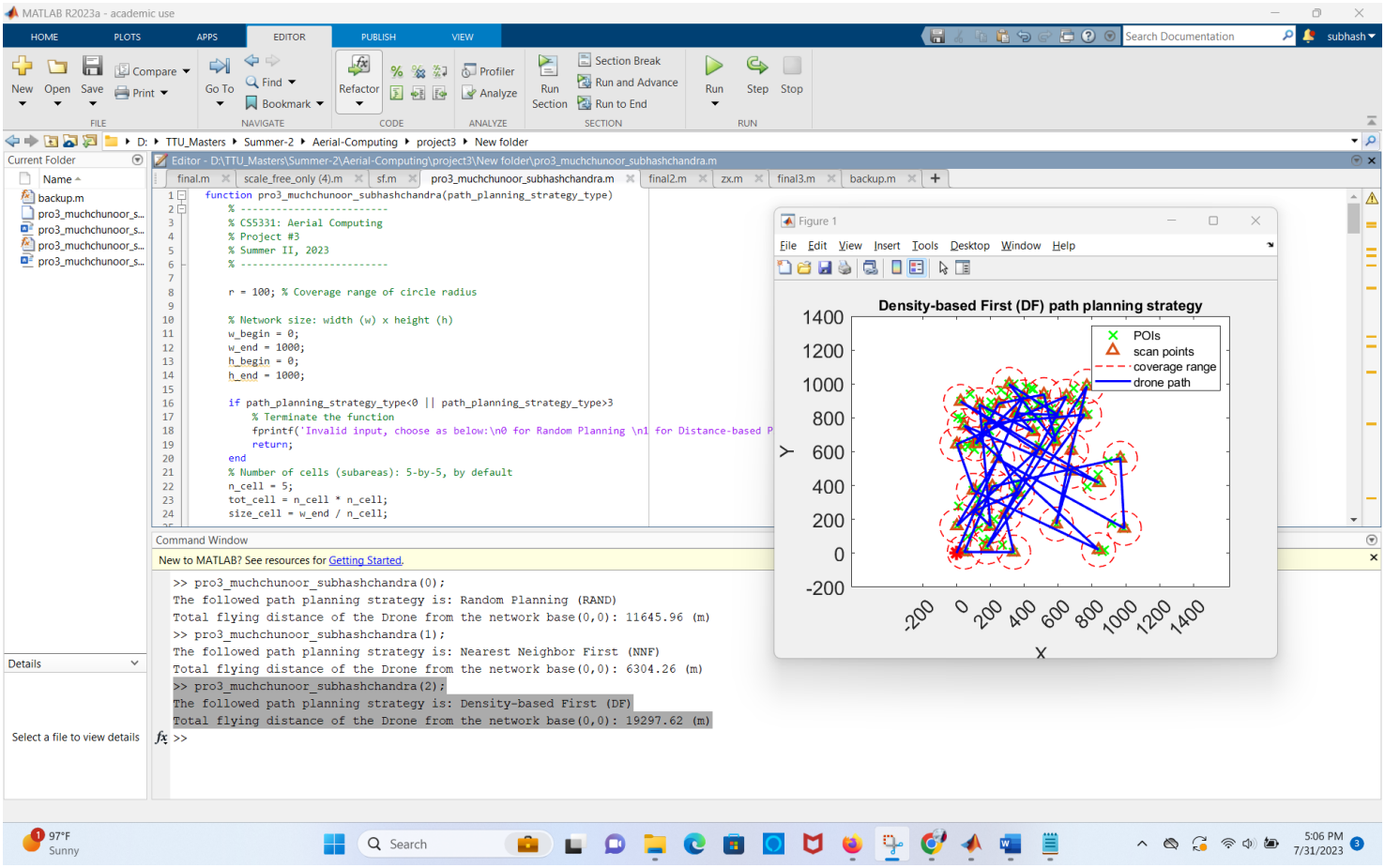
The drone path is indicated by the blue solid line in the graph.

In the final step, it prints the results as below:

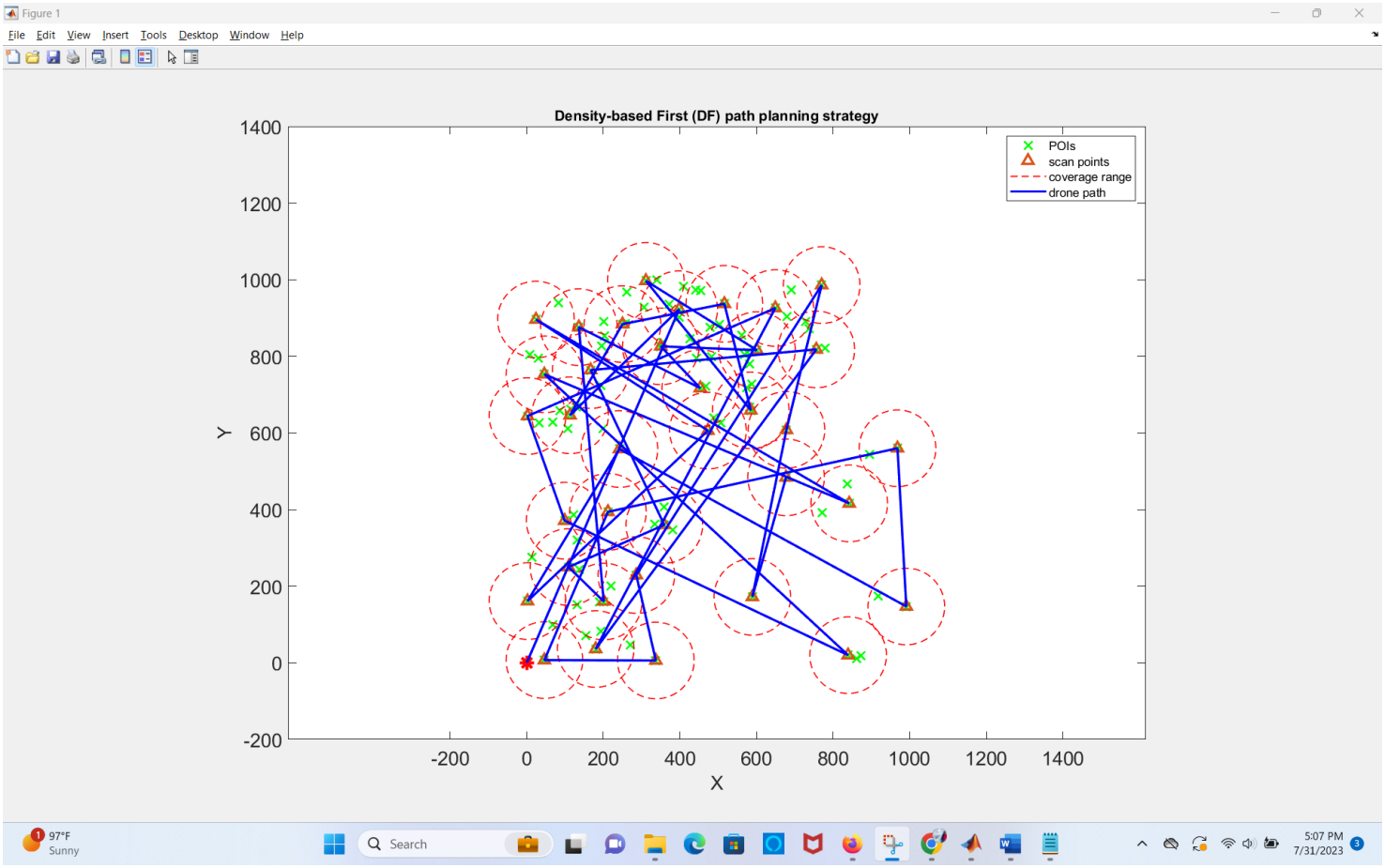
Displays the graph plotted based on the Density-based First (DF) path planning strategy as in the below screenshot.

The followed path planning strategy is: Density-based First (DF)

The total flying distance of the drone from the network base(0,0) to last scan point is calculated and it is =19297.62 meters.



Full screen screenshot of the result for better view.



Use case 3:

Code is run for the strategy=3 which is to Display the Clustered Graph results.

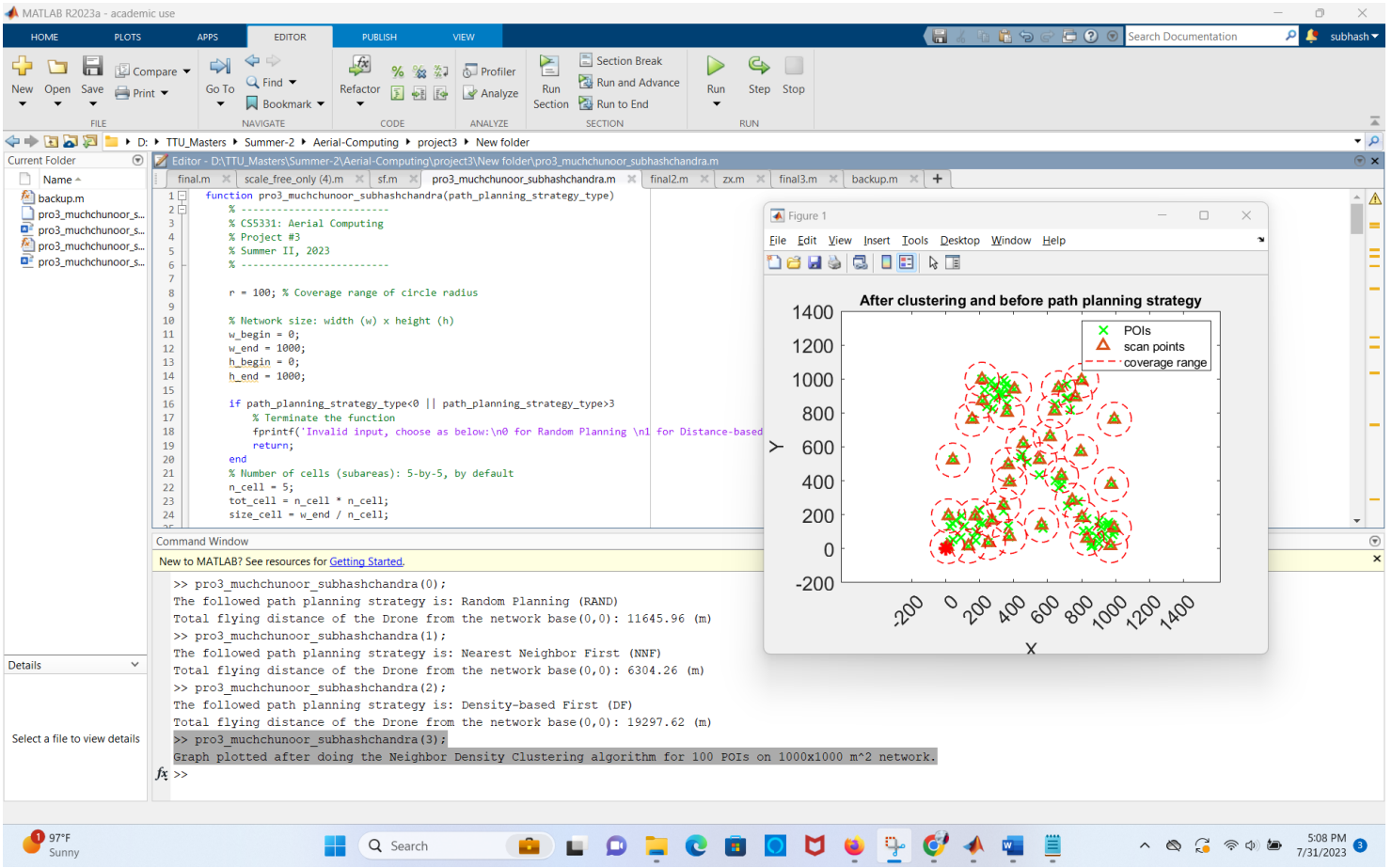
>>pro3_muchchunoor_subhashchandra(3);

Output & its explanation:

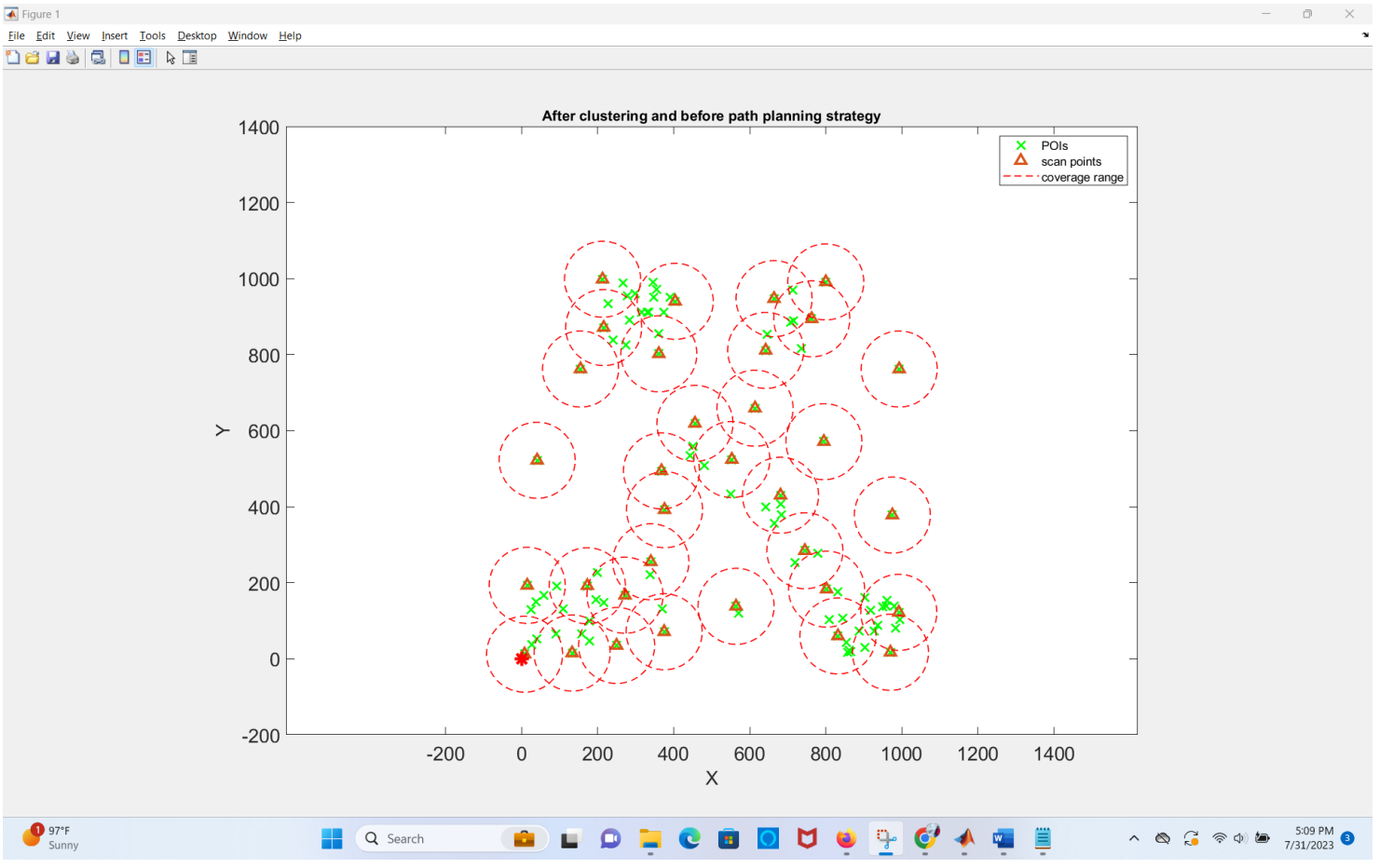
It has done the Neighbor Density based clustering algorithm on the scale free points which are indicated by the symbol ‘x’ in green color, and it marked the coverage areas of radius 100m with red color doted circles as per the requirement and also marked the center of the coverage area as scan points with triangle symbol in the graph.

In the final step, it prints the results as below:

It displays the graph plotted after the clustering is done as in the screenshot below.



Complete screen screenshot of the result for better view.



Invalid cases:

When we call the method with invalid inputs other than 0 or 1 or 2 or 3, it will give the results as in below screenshot.

>> pro3_muchchunoor_subhashchandra(-1);

Output & its explanation:

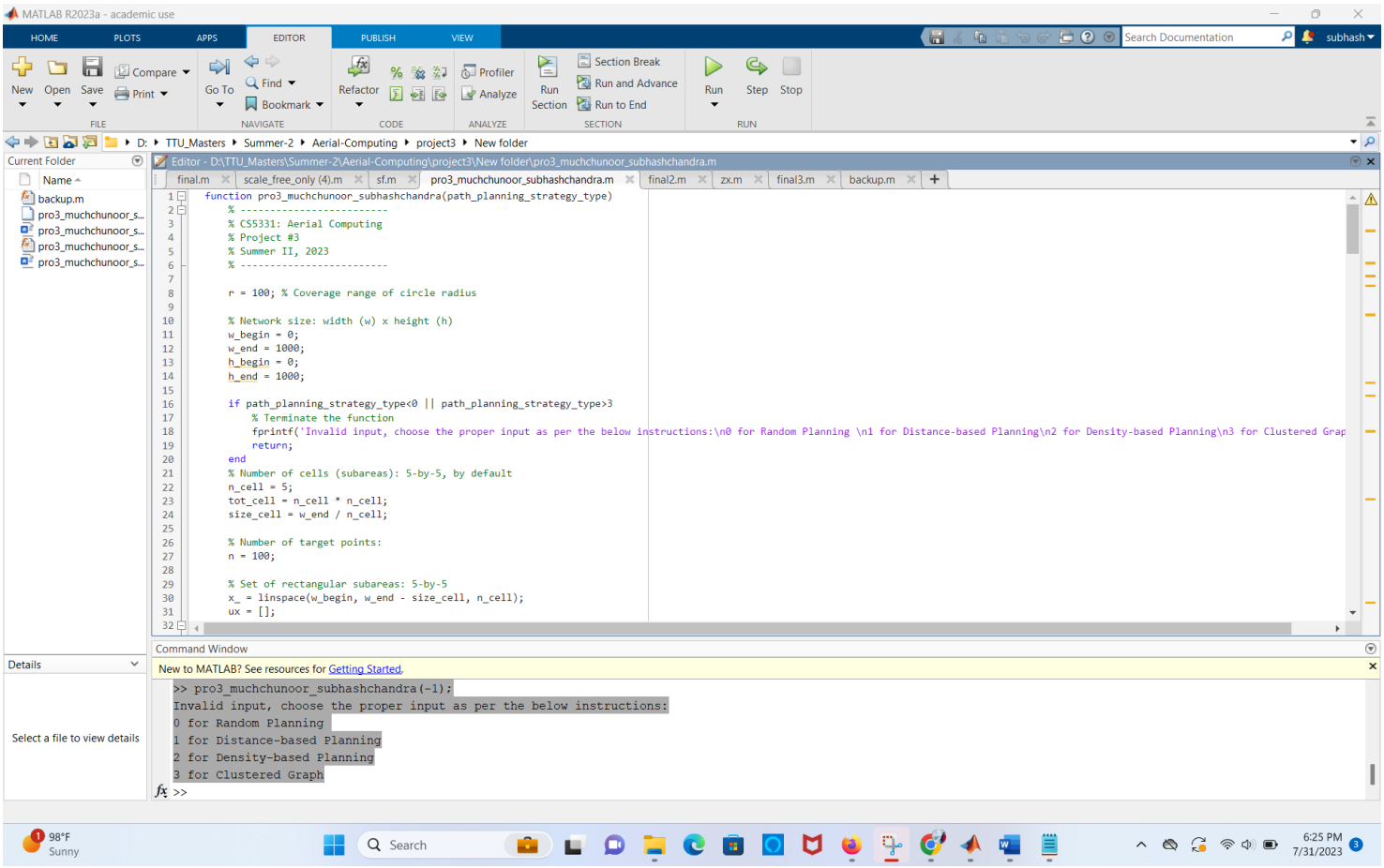
Invalid input, choose the proper input as per the below instructions:

0 for Random Planning

1 for Distance-based Planning

2 for Density-based Planning

3 for Clustered Graph



>> pro3_muchchunoor_subhashchandra(4);

Output & its explanation:

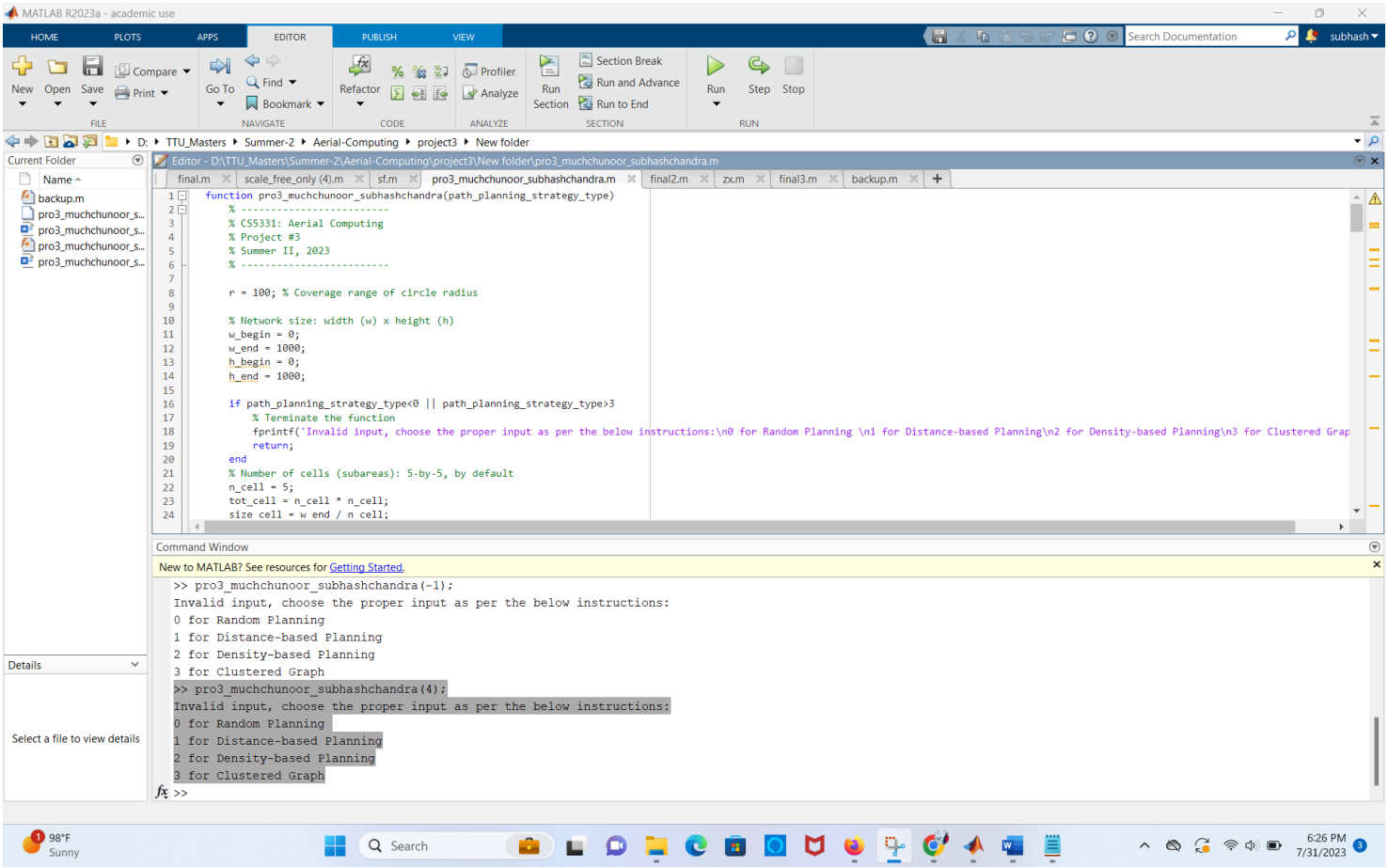
Invalid input, choose the proper input as per the below instructions:

0 for Random Planning

1 for Distance-based Planning

2 for Density-based Planning

3 for Clustered Graph



Workflow of the code in brief:

- This MATLAB code defines a function named pro3_muchchunoor_subhashchandra that performs the neighbor density clustering & the path planning for the drone on the given scale free points.
- The function accepts a single input which is path_planning_strategy_type and it determines the path planning strategy to be used.
- The valid strategy types are 0 (Random Planning), 1 (Nearest Neighbor First), 2 (Density-based First), and 3 (Clustered Graph).
- The function starts by initializing the parameters such as coverage radius of the circle, the size of the network area, and the number of POIs. It then generates a set of POIs within rectangular subareas using a scale-free distribution.
- Next, the function applies the Neighbor Density-based Clustering algorithm to group the target points into clusters within the coverage range of circles.
- It plots the POIs, scan points, and coverage circles for visualization.
- If the path_planning_strategy_type is 3 (Clustered Graph), the function perform clustering and plot the clustered graph. And then it gets terminate here.
- For the other strategy types (0, 1, or 2), the function proceeds with path planning on the clustered scan points.

- It initializes the drone's starting position at (0, 0) and fly based on the coresponding planning strategy.
- For strategy 0 (Random Planning), the drone randomly selects scan points from the available points.
- For strategy 1 (Nearest Neighbor First), the drone prioritizes points closest to the current position.
- For strategy 2 (Density-based First), the drone selects scan points that cover the most POIs in their coverage area circle.
- Once all scan points are visited, the function plots the drone's path on the same graph.
- It also displays the total flying distance of the drone from the base (0, 0) to the last scan point.
- Few other functions such as neighbor_density_clustering, distance_clustering, and count_poi_in_coverage_radius are called in this function, to facilitate the clustering and path planning processes. They are used to calculate neighbor densities, perform distance-based clustering, and count the number of POIs in coverage areas circles.