

Tania Jaswal
CS 455

Project 1: MSN Flocking Formation Control

Project Overview:

This project aims to explore the dynamics of Multi-Sensor Networks (MSN) in achieving flocking formation control. Through the implementation of various algorithms, we investigate how MSN nodes interact to exhibit different flocking behaviors in different scenarios.

Project parameters:

Number of sensor nodes: $n = 100$.

Space dimensions: $m = 2$.

Desired distance among sensor node: $d = 15$.

Scaling factor: $k = 1.2$ and interaction range $r = k \cdot d$.

Epsilon = 0.1 and Delta_t = 0.009

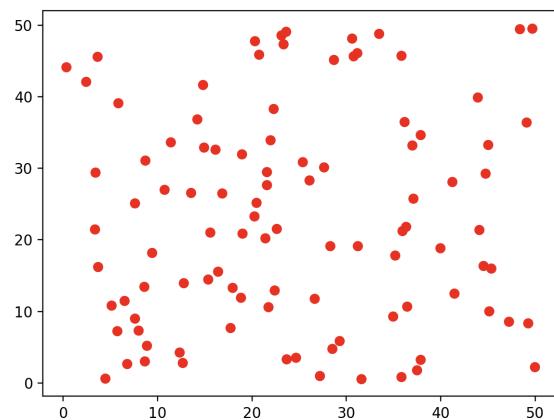
Case 1: MSN Fragmentation

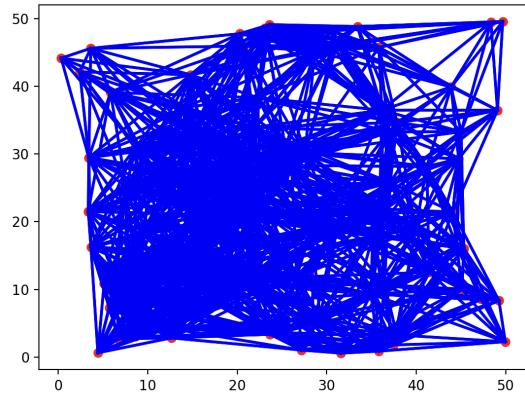
Implementation Details:

In this case, we begin by randomly generating a connected network comprising 100 sensor nodes within a 50x50 area. Subsequently, Algorithm 1 is implemented to observe the fragmentation behaviour within the network. Algorithm 1 aims to simulate scenarios where nodes start to disconnect from the main network due to certain conditions or constraints.

Results:

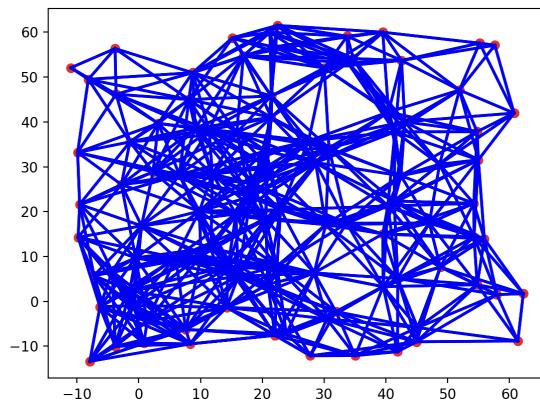
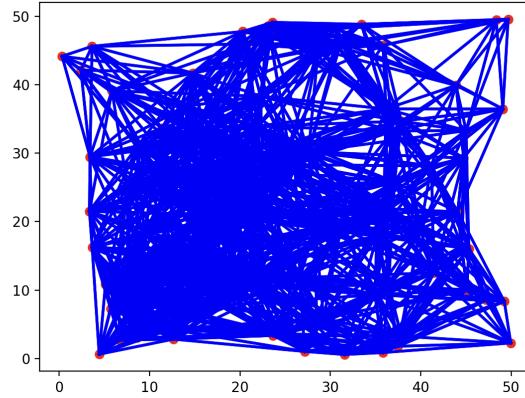
1. **Initial Deployment:**

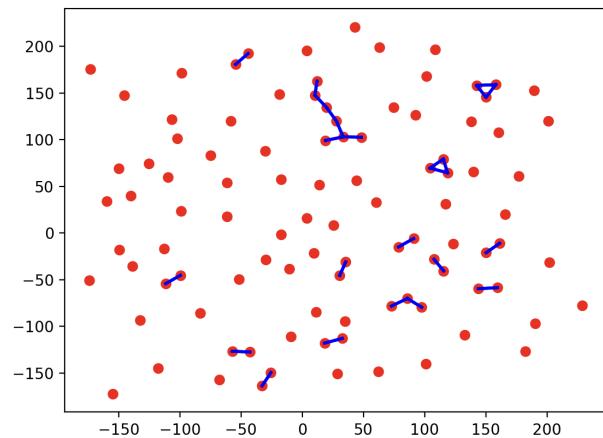
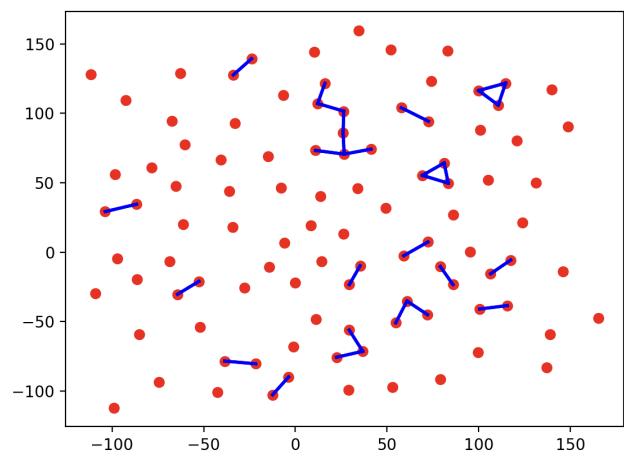
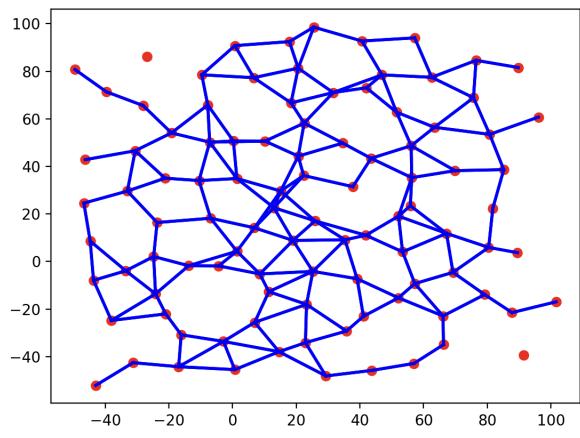


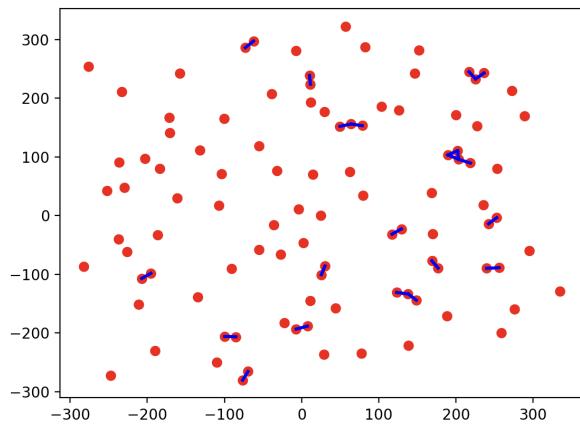


- Provided is a visual representation of the initial deployment, where sensor nodes are distributed within the specified area, with neighbouring nodes connected by blue lines.

2. **Fragmentation 6 Snapshots:**

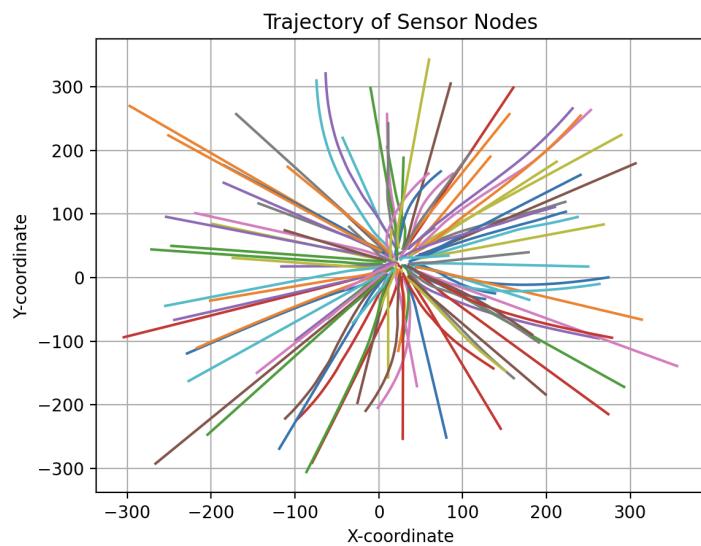






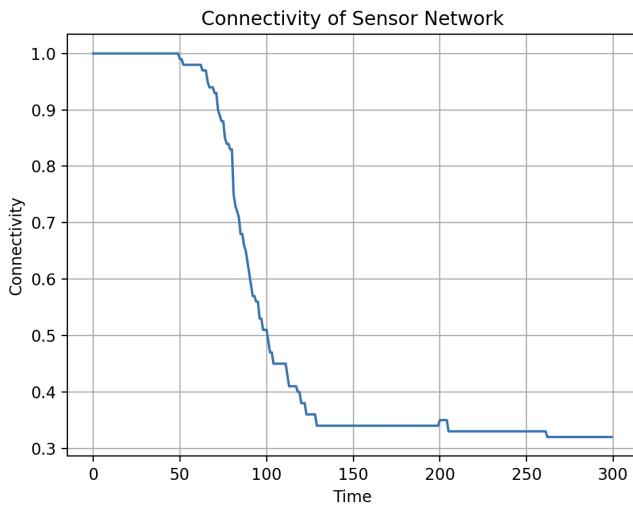
- To visualise the fragmentation process, we present a series of snapshots showcasing how the network fragments over time as Algorithm 1 progresses.

3. **Trajectory and Velocity:**



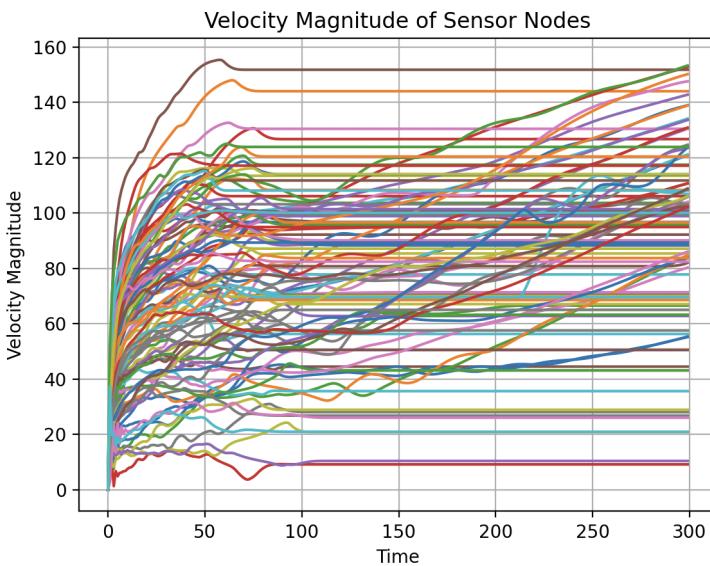
- Trajectory plots are included to illustrate the movement patterns of sensor nodes in the xy coordinate space throughout the fragmentation process.

4. **Connectivity Check:**



- We assess the connectivity of the network using a provided formula and present a plot to visualise the connectivity status over time.

5. **Velocity**



- Velocity plots are included to illustrate the movement patterns of sensor nodes in the xy coordinate space throughout the fragmentation process.

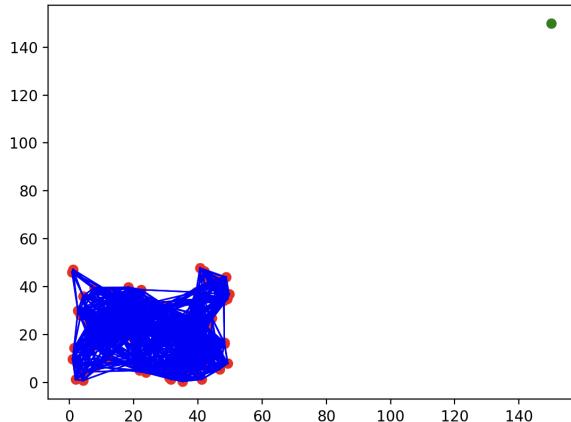
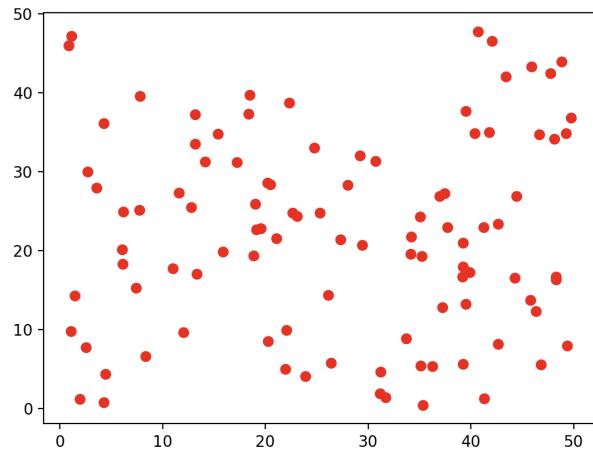
Case 2: MSN Quasi-Lattice Formation with Static Target

Implementation Details:

In this case, we generate a connected network of 100 sensor nodes within a 50x50 area, accompanied by a static target positioned at coordinates (150, 150). Algorithm 2 is employed to achieve quasi-lattice formation around the static target while ensuring collision avoidance among the sensor nodes.

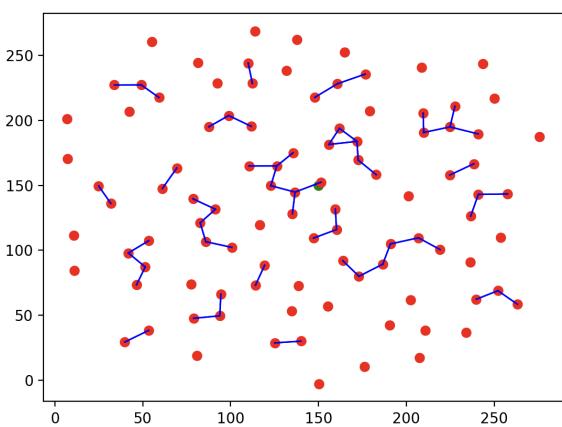
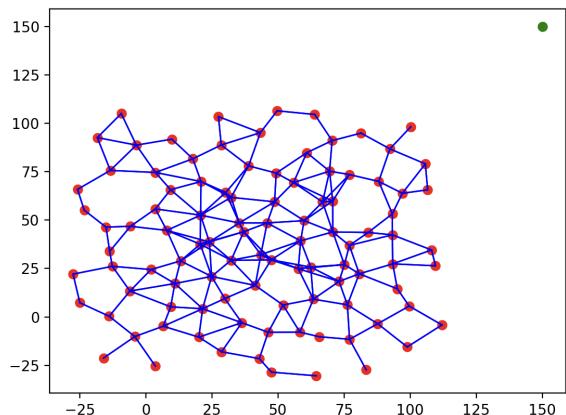
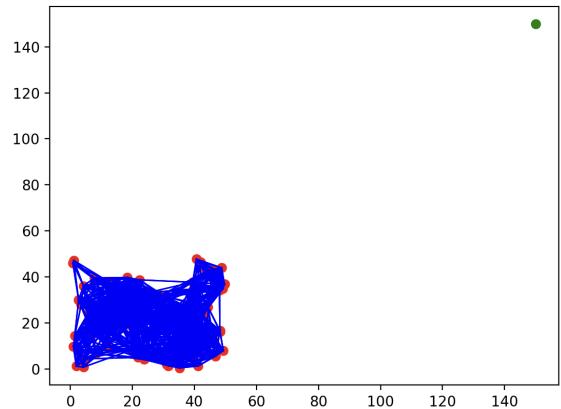
Results:

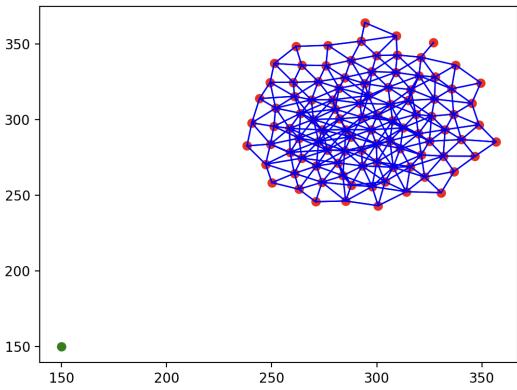
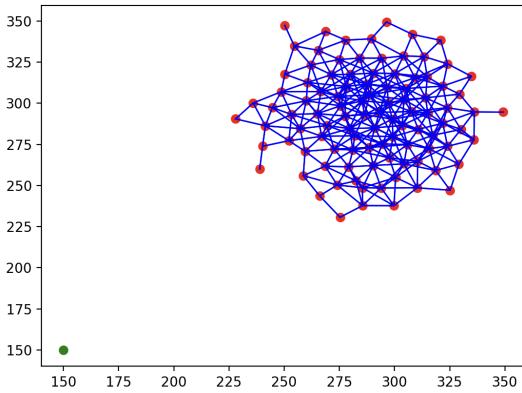
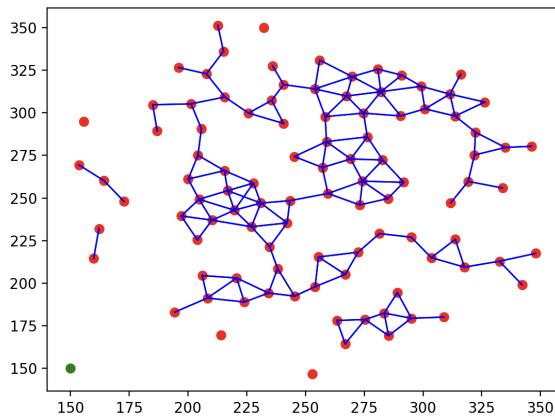
1. **Initial Deployment:**



- We visualise the initial deployment of sensor nodes and the static target, highlighting the connections between neighbouring nodes.

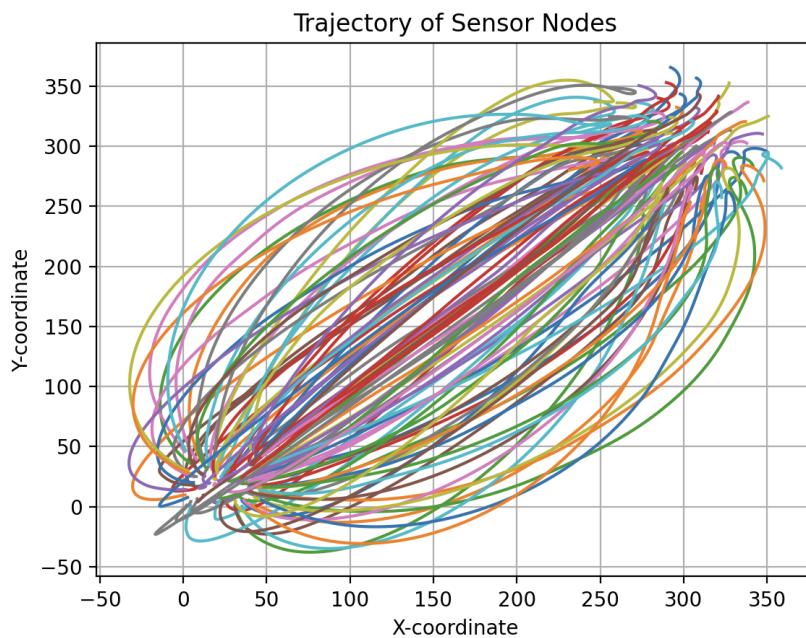
2. **Flocking Snapshots:**





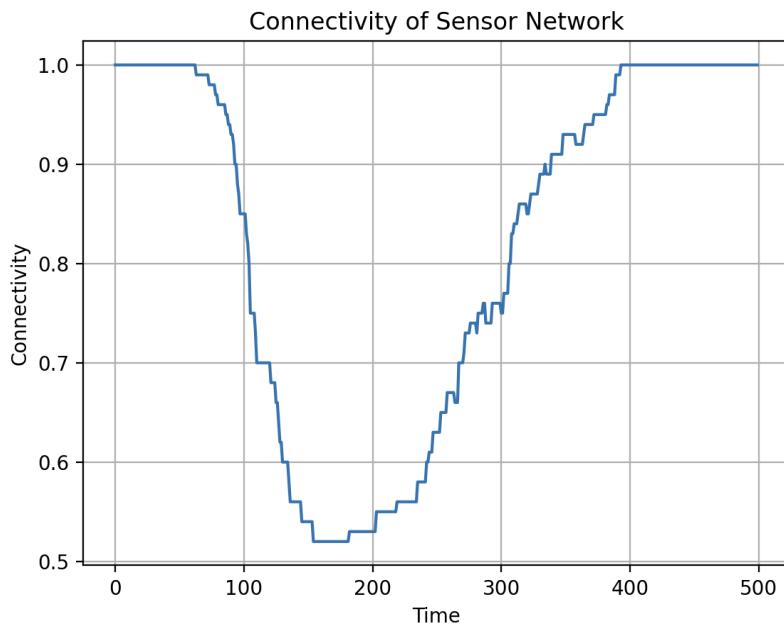
- A series of snapshots are presented to illustrate how sensor nodes flock towards the static target while avoiding collisions with each other.

3. **Trajectory**



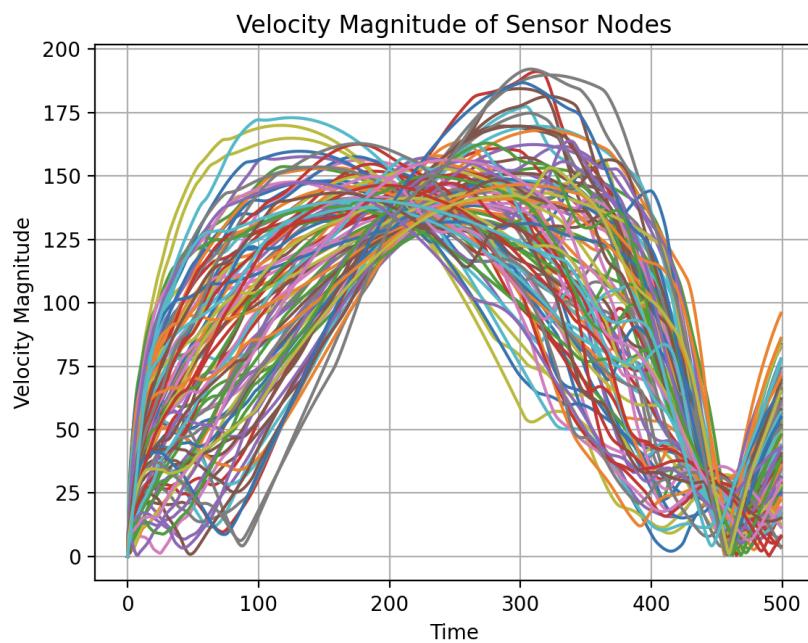
- Trajectory plots provide insights into the movement characteristics of sensor nodes as they navigate towards the target.

4. **Connectivity Check:**



- We assess the connectivity of the network and present a plot to demonstrate the network's connectivity status throughout the simulation.

5. **Velocity**



- Velocity plots provide insights into the movement characteristics of sensor nodes as they navigate towards the target.

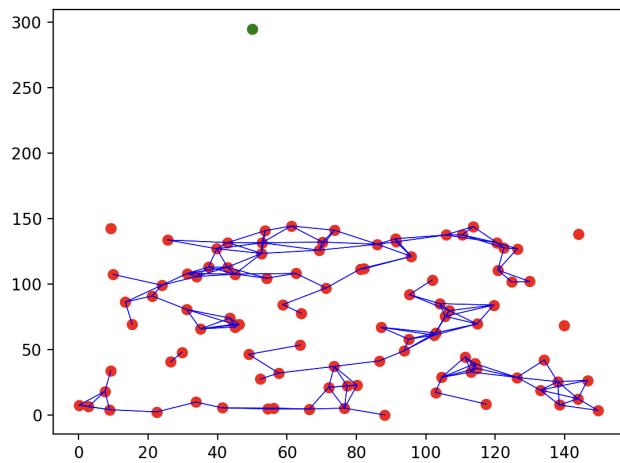
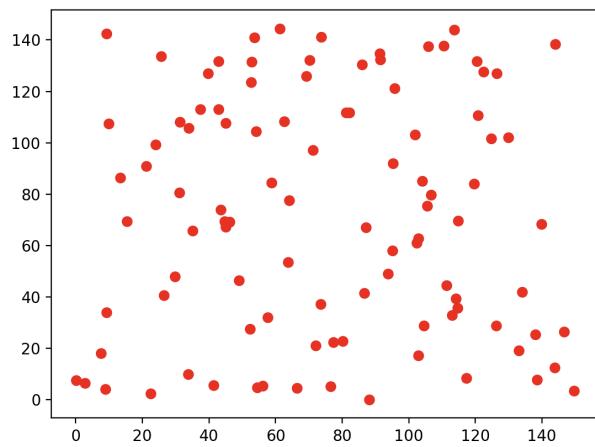
Case 3: MSN Quasi-Lattice Formation with Dynamic Target (Sine Wave)

Implementation Details:

In this case, we create a connected network of 100 sensor nodes within a larger area (150x150), with a target moving along a sine wave trajectory. Algorithm 2 is utilised to achieve quasi-lattice formation around the moving target while ensuring collision avoidance among the sensor nodes.

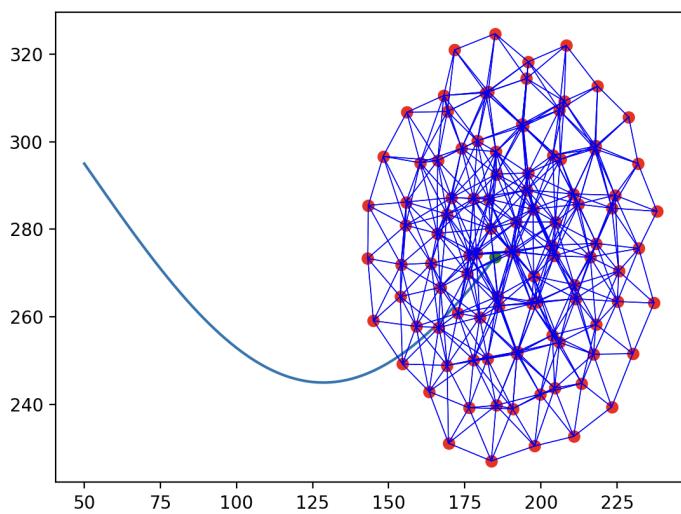
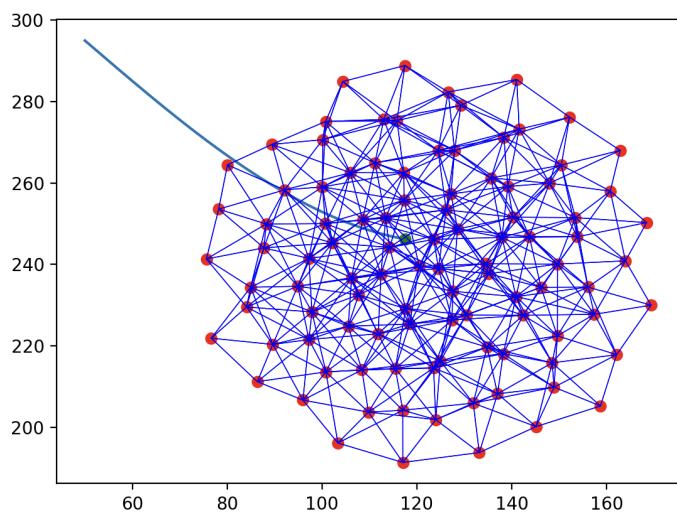
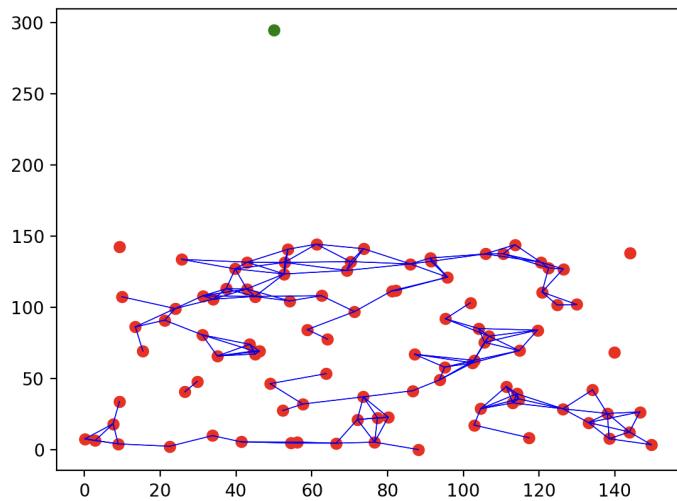
Results:

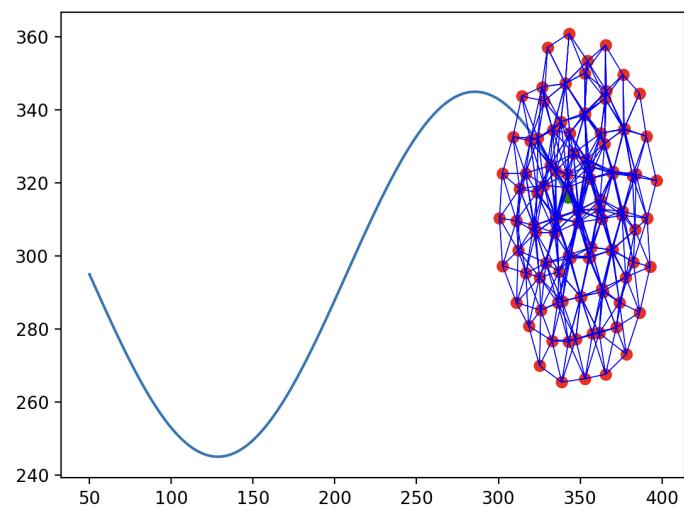
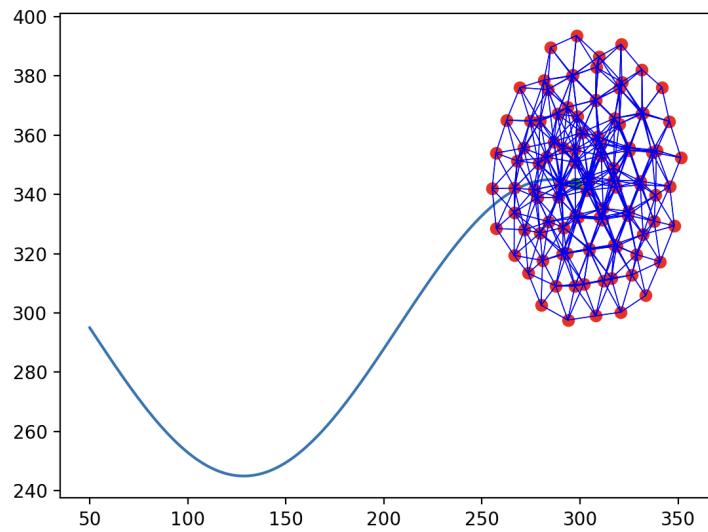
1. **Initial Deployment:**

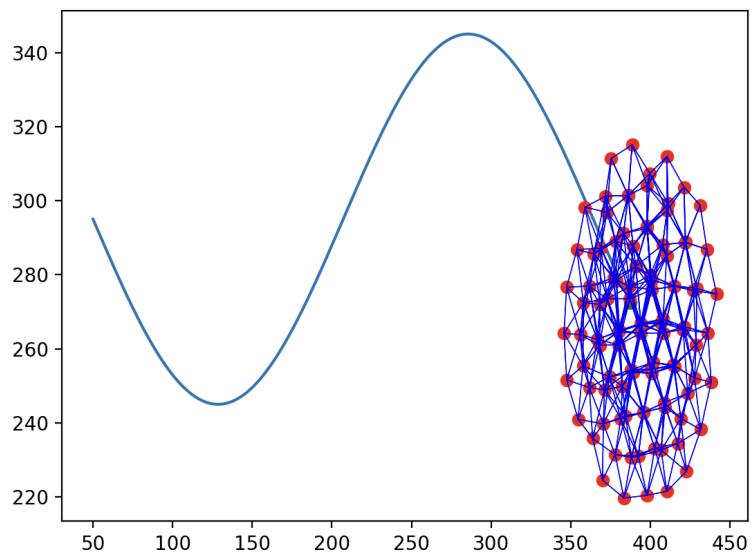


- We visualize the initial deployment of sensor nodes and the moving target, highlighting the connections between neighboring nodes.

2. **Flocking Snapshots:**

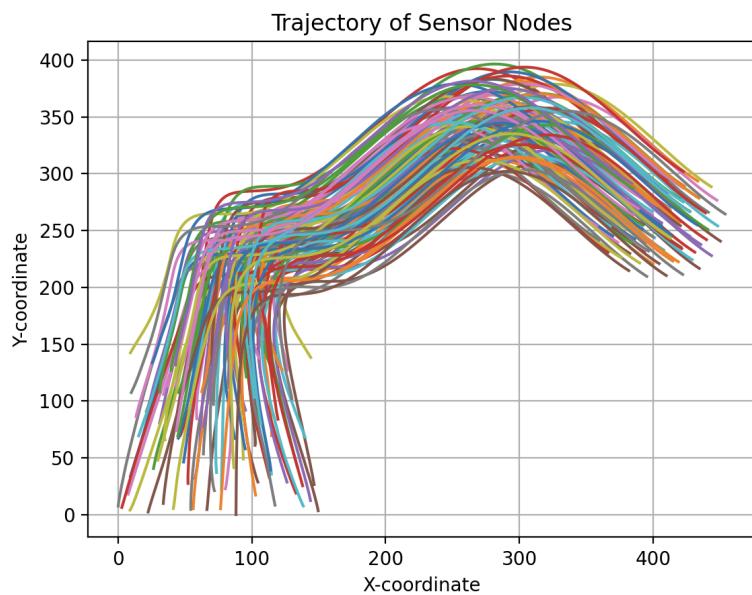


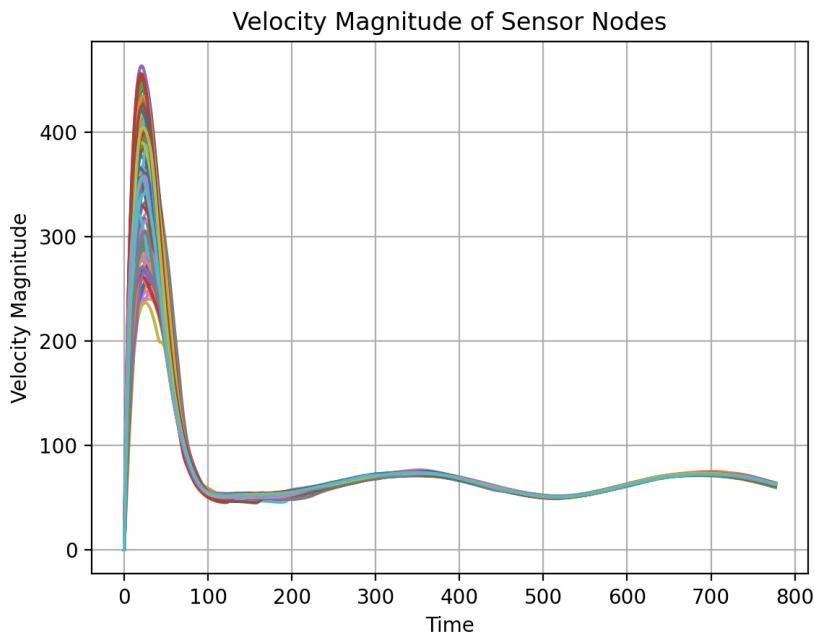




- A series of snapshots are provided to illustrate how sensor nodes dynamically adjust their positions to track the moving target following a sine wave trajectory.

3. **Trajectory and Velocity:**

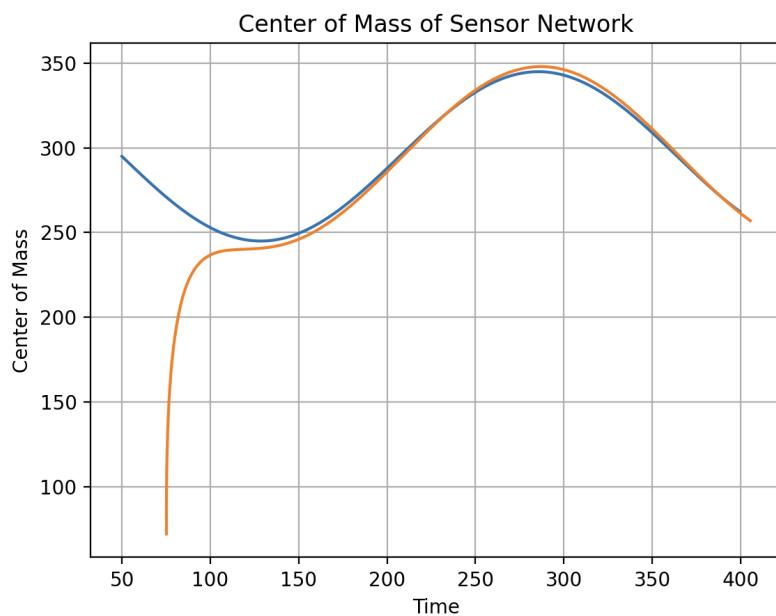




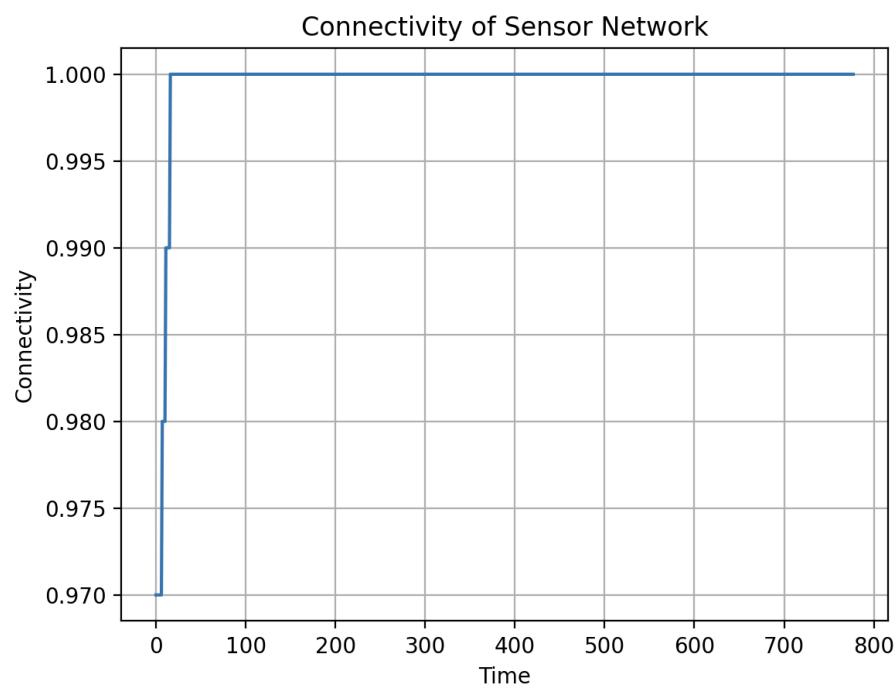
- Trajectory and velocity plots depict the movement patterns of sensor nodes and the target in the xy coordinate space.

4. **Center of Mass Tracking:**

- We track the trajectory of the Center of Mass (COM) of the sensor nodes and overlay it with the trajectory of the moving target to showcase the tracking behavior of the MSN.



5. **Connectivity Check:**



- Connectivity analysis is conducted to evaluate the network's connectivity status over time, with results presented in a plot.

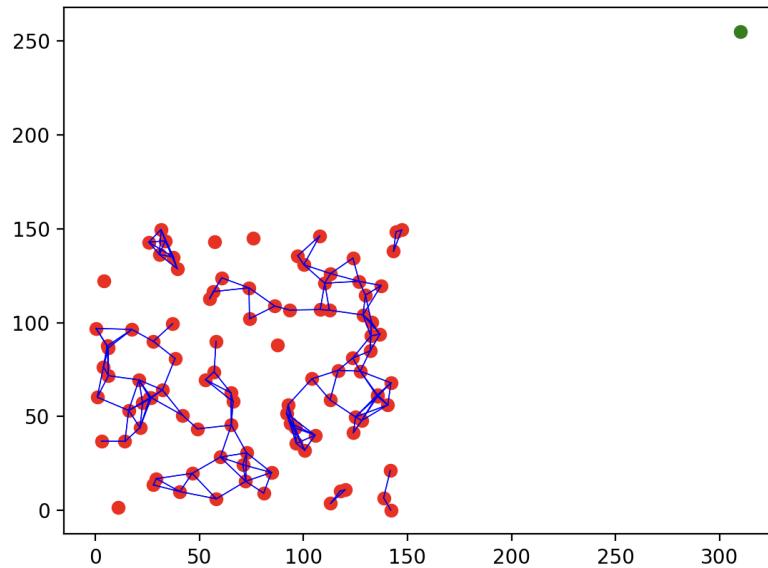
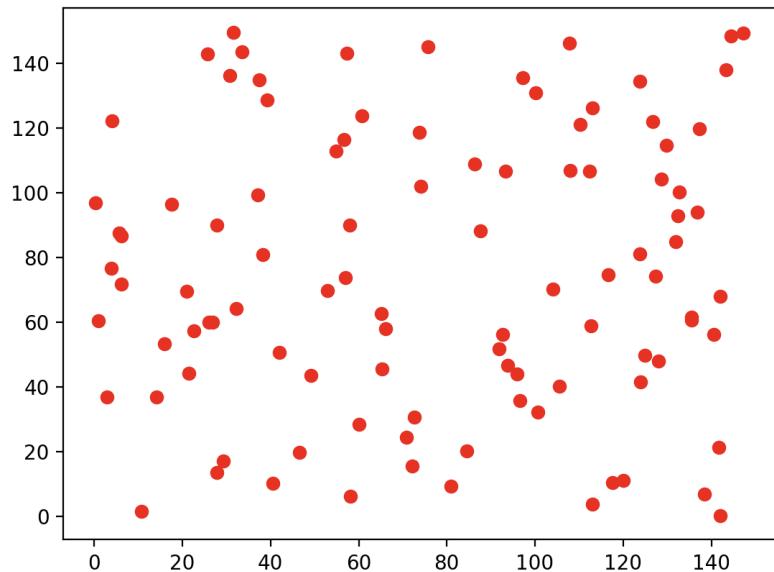
Case 4: MSN Quasi-Lattice Formation with Dynamic Target (Circular Trajectory)

Implementation Details:

In this case, we generate a connected network of 100 sensor nodes within the same larger area (150x150), with a target moving along a circular trajectory. Algorithm 2 is employed to achieve quasi-lattice formation around the moving target while ensuring collision avoidance among the sensor nodes.

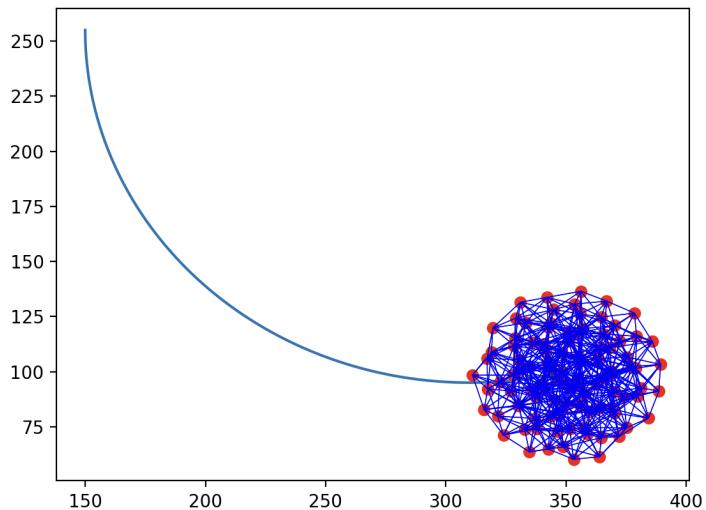
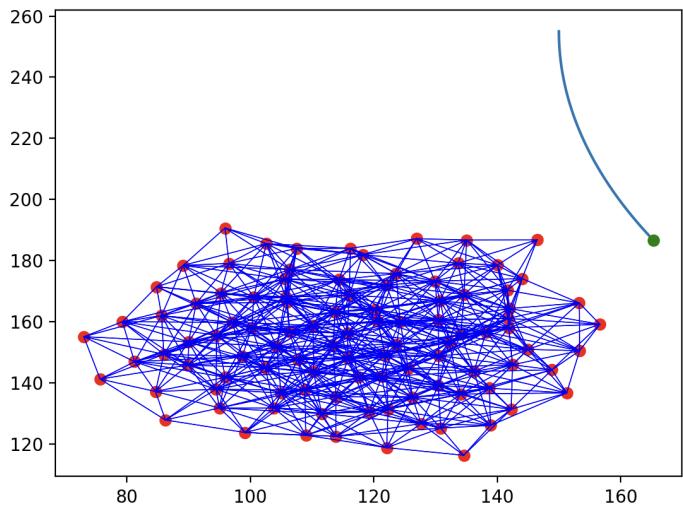
Results:

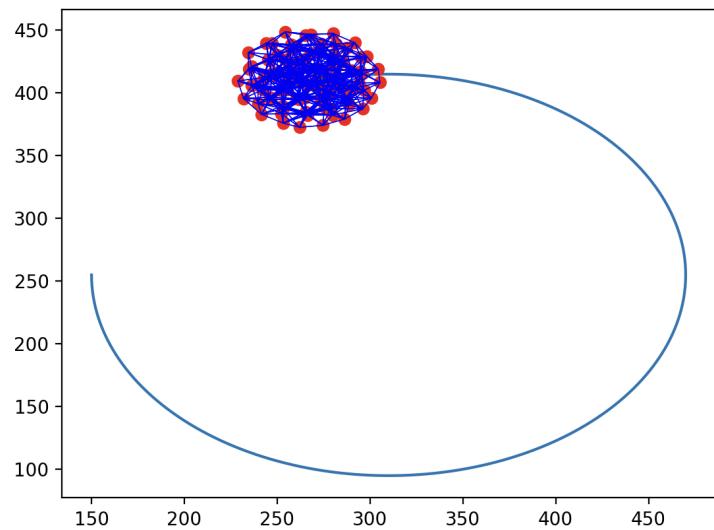
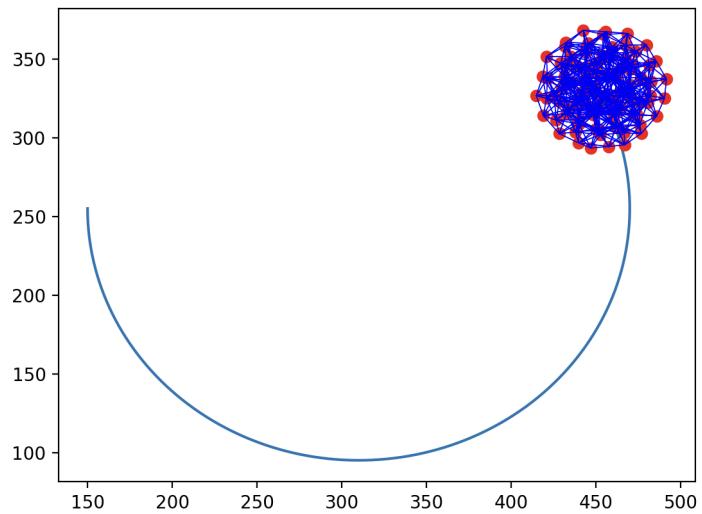
1. **Initial Deployment:**

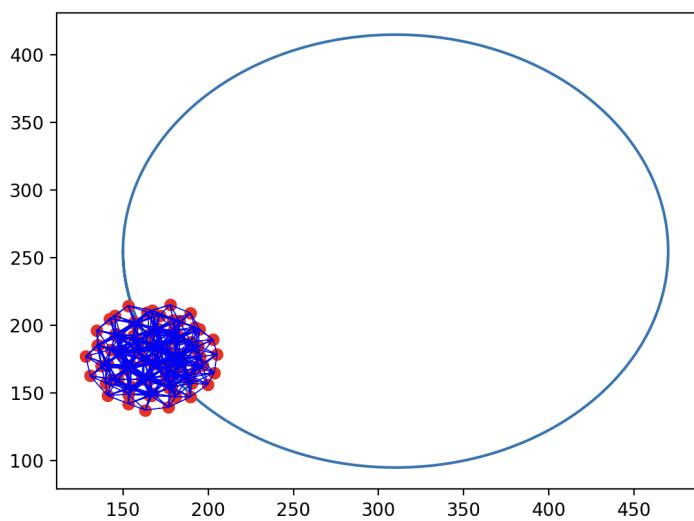
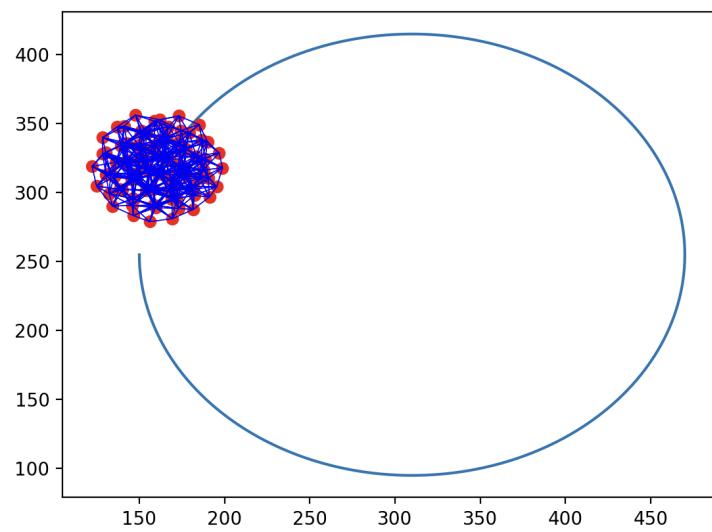


- We visualize the initial deployment of sensor nodes and the moving target, highlighting the connections between neighboring nodes.

2. **Flocking Snapshots:**

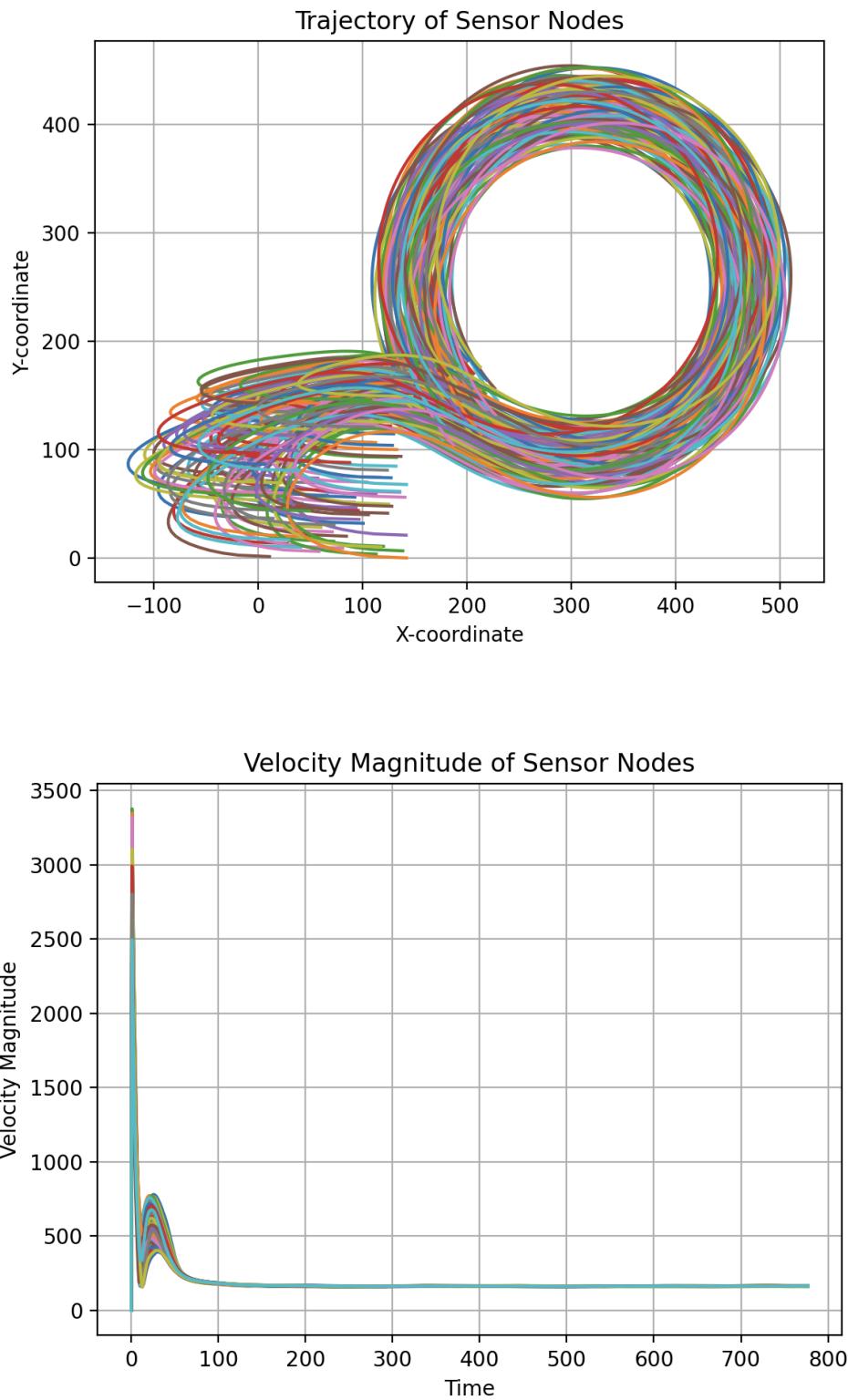






- A series of snapshots are presented to illustrate how sensor nodes dynamically adjust their positions to track the moving target following a circular trajectory.

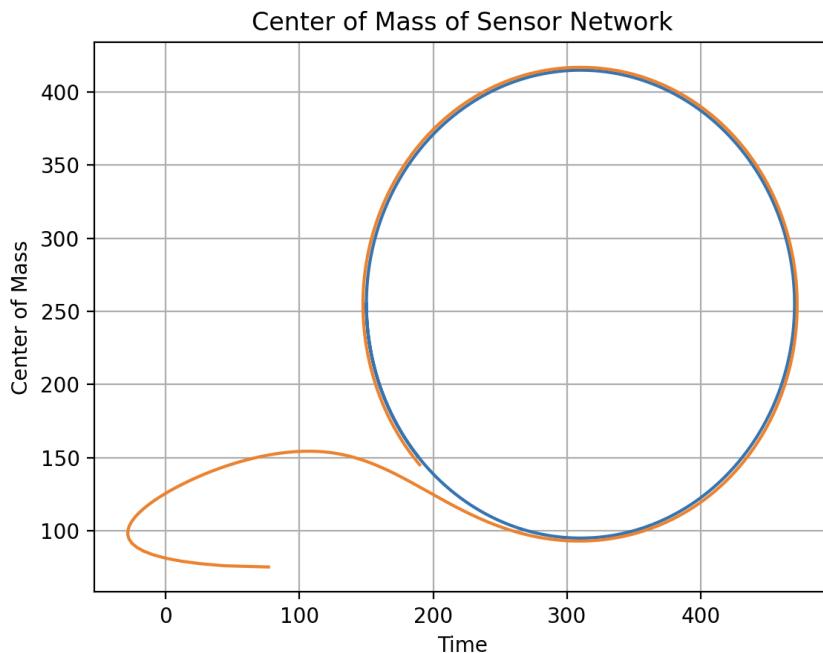
3. **Trajectory and Velocity:**



- Trajectory and velocity plots provide insights into the movement characteristics of sensor nodes and the target in the xy coordinate space.

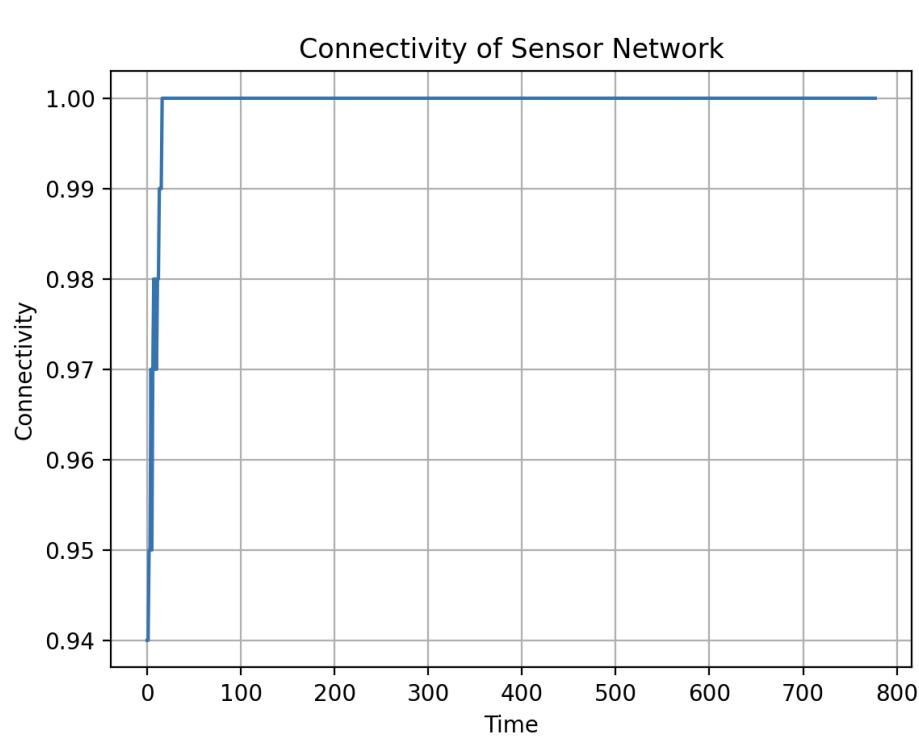
4. **Center of Mass Tracking:**

- We track the trajectory of the Center of Mass (COM) of the sensor nodes and overlay it with the trajectory of the moving target to showcase the tracking behavior of the MSN.



5. **Connectivity Check:**

- Connectivity analysis is conducted to evaluate the network's connectivity status over time, with results presented in a plot.



How to Run:

1. Extract the compressed files and open them in an EDI.
2. Install the required libraries mentioned below.
3. Run the script.
4. Run python file using “python filename.py”

Required Libraries:

- numpy: Install using pip install numpy
- matplotlib: Install using pip install matplotlib