MARS

Autonomous Navigation and Multi-Robot Coordination

Question 1.1

We can get For an all-to-all communication graph with N robots:

1. **Adjacency Matrix (A)**

Adjacency Matrix (A): In an all-to-all communication graph with N robots, each robot is connected to every other robot. The adjacency matrix A is defined as

A = ones(N,N) - I(N)

Where:

- ones(N,N) creates an N×N matrix of all 1's

- I(N) is the N×N identity matrix

- We subtract the identity matrix because robots don't connect to themselves

2. **Degree Matrix (D)**

The degree matrix DDD is a diagonal matrix where each diagonal element represents the degree (number of neighbors) of the corresponding node.

D = (N-1)I(N)

Since each robot connects to all other robots except itself, each node has degree (N-1).

3. Graph Classification

- This type of graph is called a Complete Graph or Fully Connected Graph

- It's often denoted as K_n where n is the number of vertices

4. Connectivity Properties:

The graph is:

- Strongly connected (directed case)
- Connected (undirected case)
- Regular (all nodes have same degree)
- Maximum connectivity (removing any edge still maintains connectivity)

MARS

Question 1.2

Definition of the Laplacian Matrix (L):
The Laplacian matrix for a graph is defined as:
$L = D - A$
where:

- D is the degree matrix.
- A is the adjacency matrix.

For the given N=4 robots, let's compute the Laplacian matrix for each graph.

Graph 1: Complete Graph

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$$A_1 = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \qquad D_1 = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} \qquad L_1 = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

Properties:

- Type: Complete graph (K_4) /Undirected Graph
- Connectivity: Fully connected
- Algebraic connectivity = 4
- Eigenvalues: {0, 4, 4, 4}
- Degree of each vertex: 3

Graph 2
Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \qquad D_2 = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} \qquad L_2 = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

Properties:

- Type: Undirected Graph
- Connectivity: Connected
- Eigenvalues: {0, 2, 3, 5}
- Algebraic connectivity = 2
- Degrees: {2, 3, 2, 3}
-

MARS

Graph 3

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$A_3 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ $D_3 = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$ $L_3 = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}$

Properties:

- Type: Cycle/Ring graph /Undirected Graph
- Connectivity: Connected
- Eigenvalues: {0, 2, 2, 4}
- Algebraic connectivity = 2
- Degree of each vertex: 2

Graph 4

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$A_4 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ $D_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $L_4 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$

Properties:

- Type: Path graph/Undirected Graph
- Connectivity: Connected
- Eigenvalues: {0, 0.586, 2, 3.414}
- Algebraic connectivity = 1
- Degrees: {1, 2, 2, 1}

Subgraph Relationships:

- ✓ G_4 is a subgraph of G_3 (removing one edge from G_3 gives G_4)
- ✓ G_3 is a subgraph of G_2 (removing one diagonal edge from G_2 gives G_3)
- ✓ G_2 is a subgraph of G_1 (removing two diagonal edges from G_1 gives G_2)

Therefore, we have a hierarchy:
$G_4 \subset G_3 \subset G_2 \subset G_1$
The connectivity strength decreases as we move from G_1 to G_4, which is reflected in the second smallest eigenvalue (algebraic connectivity) of each Laplacian matrix.

Question 1.3

MARS

Graph 5

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$A_5 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ $D_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $L5 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 2 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & -1 & 1 \end{bmatrix}$

Properties:

- Type: Directed cyclic graph
- Connectivity: Strongly connected
- Contains a directed cycle: 1→2→3→4→1

Graph 6

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$A_6 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ $D_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $L6 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$

Properties:

- Type: Directed tree with additional edges
- Connectivity: Not strongly connected
- Has a directed path from 2→1 and 3→2→1

Graph 7

Adjacency Matrix (A), Degree Matrix (D) and Laplacian Matrix (L)

$A_7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ $D_7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ $L_7 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Properties:

- Type: Directed path/chain
- Connectivity: Not strongly connected
- Simple directed path: 2→1, 3→2, 4→3

Subgraph Relationships:

- G_7 is a subgraph of G_6 (removing edge 4→1)
- G_6 is a subgraph of G_5 (removing edge 3→4)
- The connectivity decreases from G_5 to G_7, with:

MARS

- G_5: Strongly connected
- G_6: Weakly connected
- G_7: Weakly connected (directed path)

QUESTION 2

Question 2.1

We calculate a discrete updater so that the robots update their position each step with the
following code.

```
X(i,k)=X(i,k-1)-Ux(i)*dt;
Y(i,k)=Y(i,k-1)-Uy(i)*dt;
```

we realize that we have flipped the signs as we will later take the negative of the Laplacian matrix.
But this code provides the same results.

Question 2.2

```
Ux(i)=sum_x;
Uy(i)=sum_y;
sum_x = sum_x - alpha*A(i,j)*(X(i,k-1)-X(j,k-1))   + beta*norme(1) ;
sum_y = sum_y - alpha*A(i,j)*(Y(i,k-1)-Y(j,k-1)) + beta*norme(2)  ;
```

Question 2.3

The consensus is then reached as a control direction of X and Y for each robot and is calculated with the following formula.

```
for j = 1:N
    Ux(i) = Ux(i) + alpha*(X(i,k-1)-X(j,k-1));
    Uy(i) = Uy(i) + alpha*(Y(i,k-1)-Y(j,k-1));
end
```

MARS

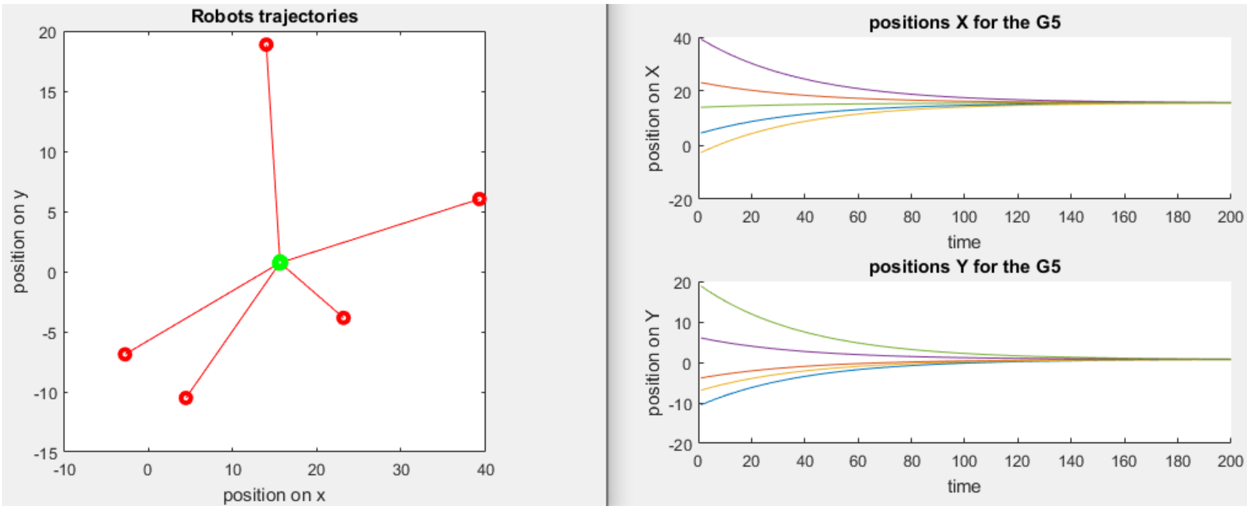


Figure 1: Simple Consensus for 5 Robots

- The signs are still flipped but get changed when we implement the Laplacian. If the initial position of the robots is known (and they all move at the same rate), we can calculate the consensus position to be the center of mass of the collective robots in 2D space. This is because of how we calculate where its next position to move is. It is interesting to see that the convergence rate is much quicker with more robots, as there is more "gravity" pulling them all to the consensus location.

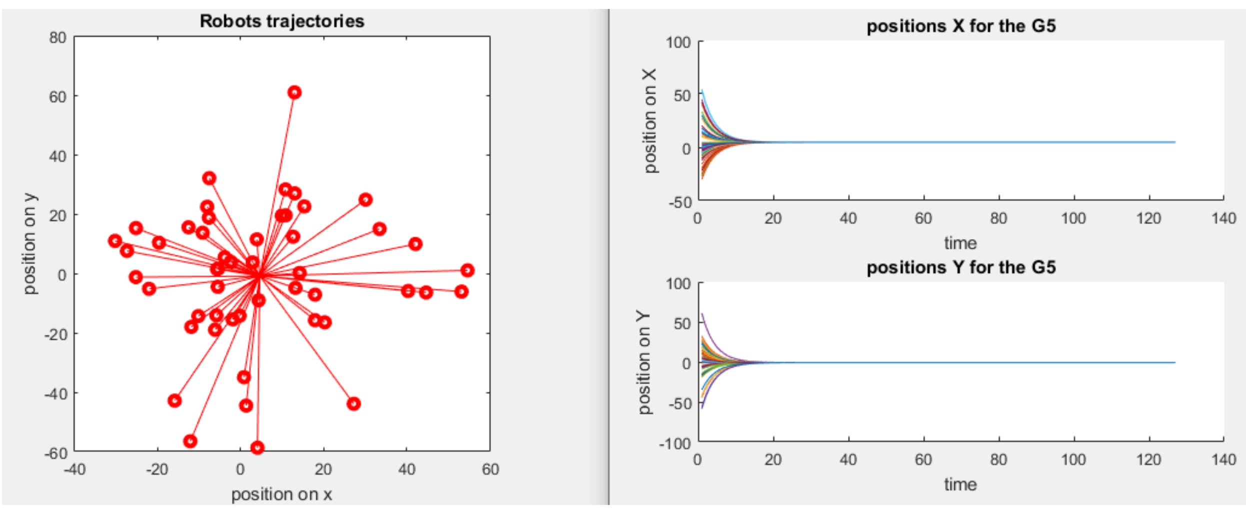


Figure 2: Simple Consensus for 50 Robots

MARS

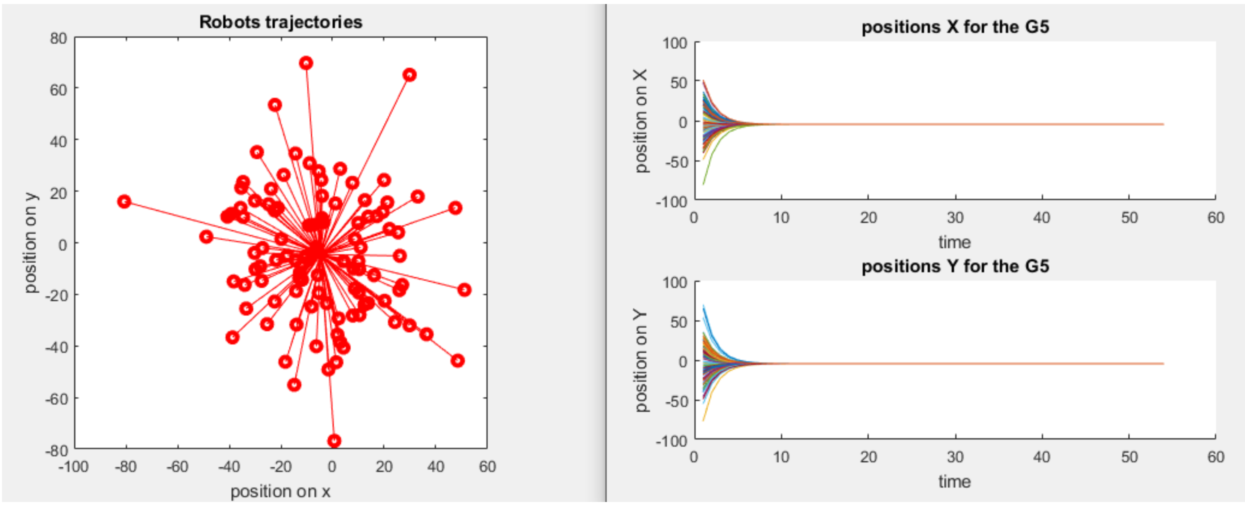


Figure 3: Simple Consensus for 100 Robots

Question 2.4

The Laplacian is just the Degree matrix minus the Adjacency Matrix. We are able to save many
for loops of calculation as we implement the following control.

```
Ux = L*X(:,k-1);
Uy = L*Y(:,k-1);
```

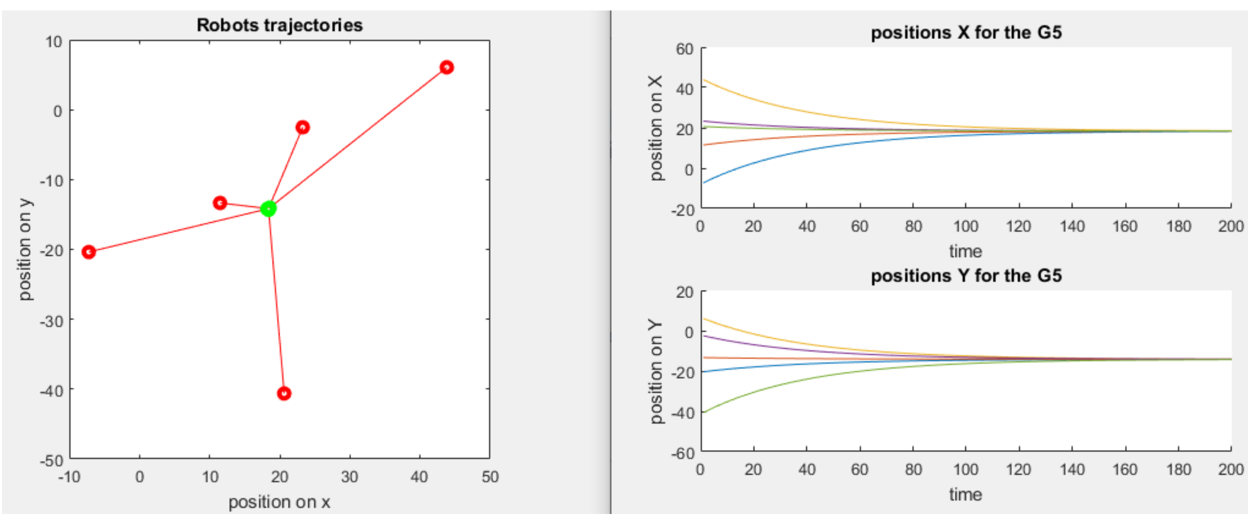


Figure 4: Laplacian Consensus for 5 Robots

QUESTION 3

Question 2.1

MARS

The Laplacian matrixes are implemented as below. We have skipped the step of creating the
Adjacency Matrix and Degree Matrix after the first one as it follows the same pattern.
Graph characteristics:
(a) - G1: Undirected Graph, Complete graph, algebraic connectivity = 4
(b) - G2: Undirected Graph, Connected, algebraic connectivity = 2
(c) - G3: Undirected Graph, Connected, algebraic connectivity = 2
(d) - G4: Undirected Graph, Connected, algebraic connectivity = 1

$$L_1 = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix} \qquad L_2 = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

$$L_3 = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \qquad L_4 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Question 2.2

Using the Laplacian for individual graph, we are able to achieve the results below

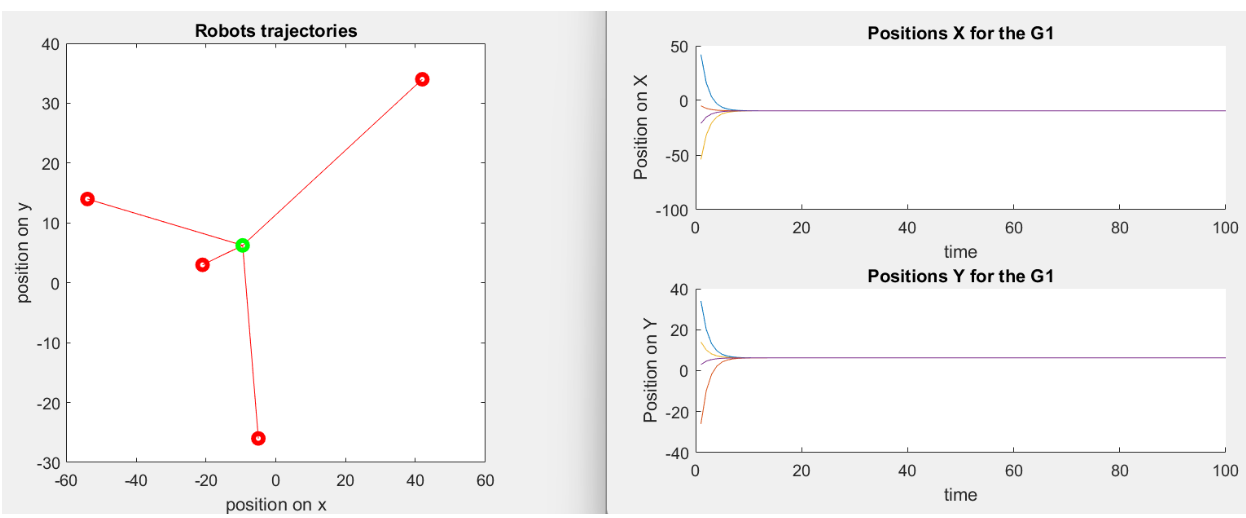


Figure 5: Laplacian Consensus for 4 Robots, G1 Connectivity Graph

MARS

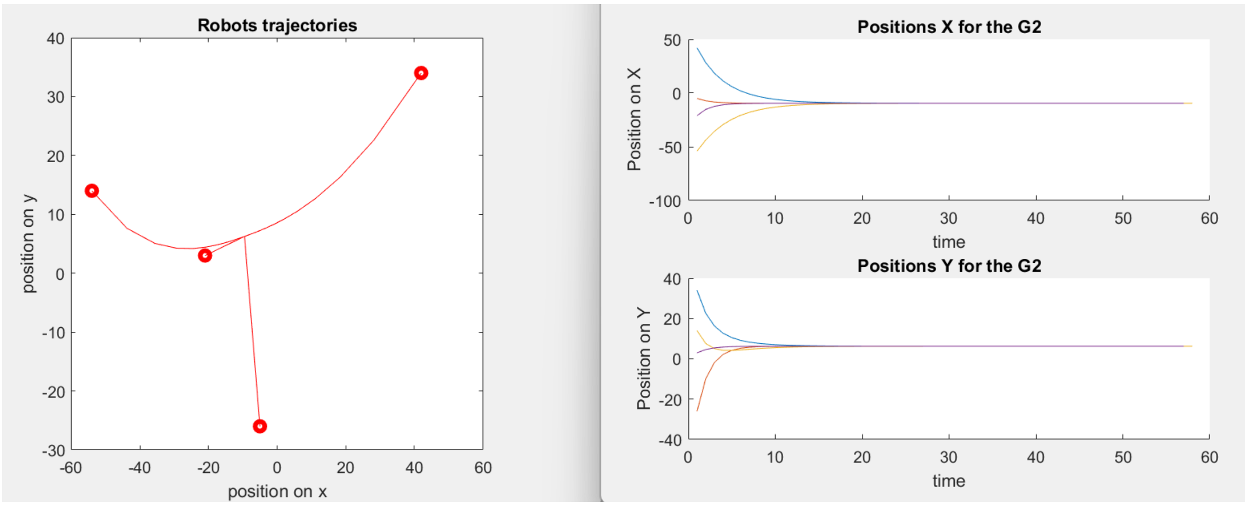


Figure 6: Laplacian Consensus for 4 Robots, G2 Connectivity Graph

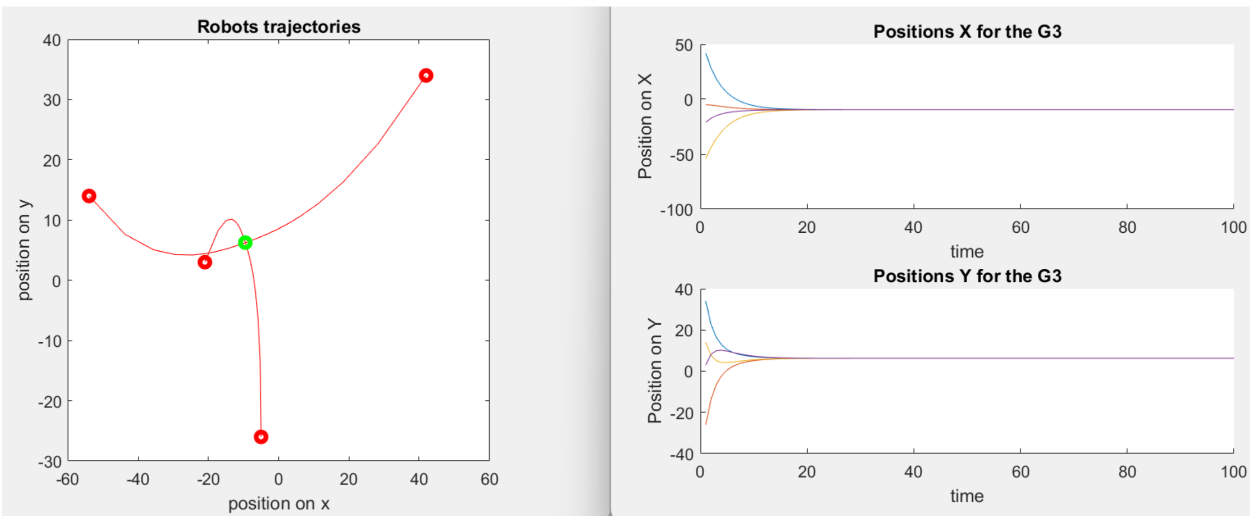
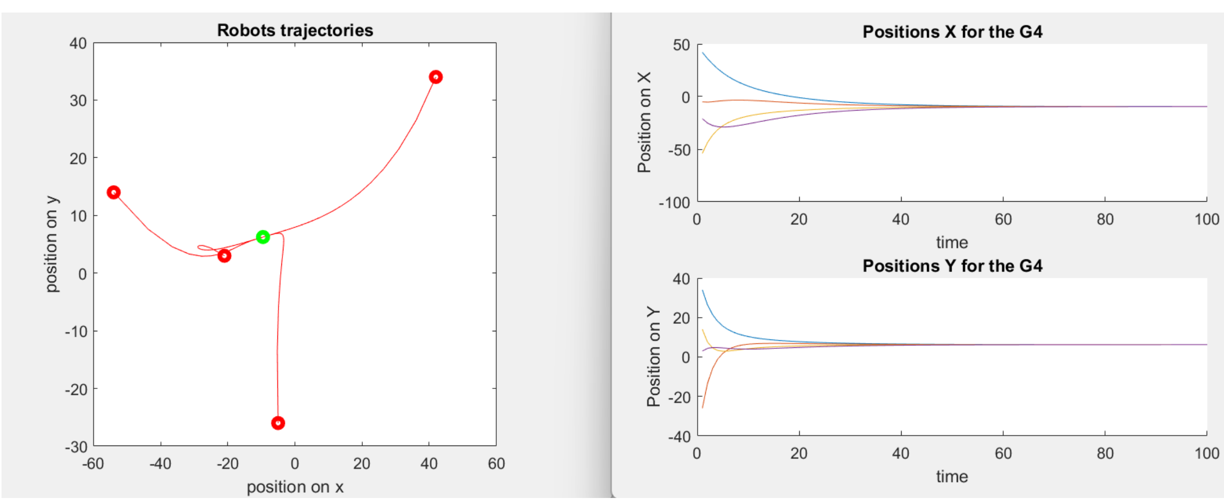


Figure 7: Laplacian Consensus for 4 Robots, G3 Connectivity Graph



MARS

Figure 8: Laplacian Consensus for 4 Robots, G4 Connectivity Graph

Question 2.3 and 2.4

$$L5 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 2 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & -1 & 1 \end{bmatrix} \quad L6 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \quad L_7 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

a) - G5: Directed Graph, strongly connected, algebraic connectivity = 1.5

b) - G6: Directed Graph, connected, algebraic connectivity = 1

c) - G7: Directed Graph, subgraphs = {1, 2} & {2, 3, 4}, algebraic connectivity = 0

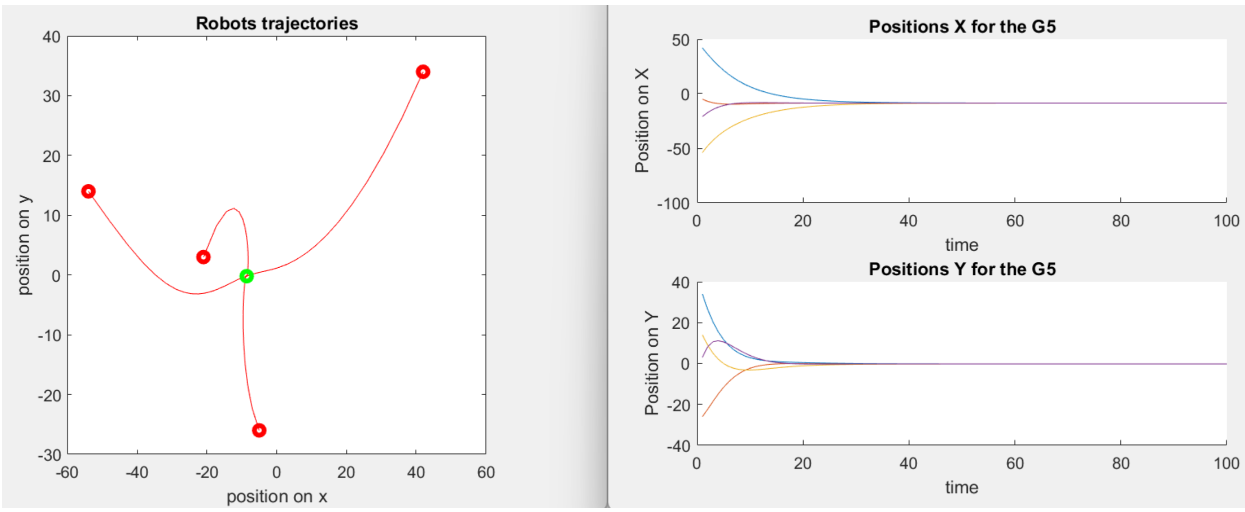


Figure 9: Laplacian Consensus for 4 Robots, G5 Directed Graph

- G5 has the most connections among all three graphs
- Node 4 receives information from all other nodes
- Strong connectivity leads to faster and more robust consensus
- The higher number of connections provides more paths for information flow

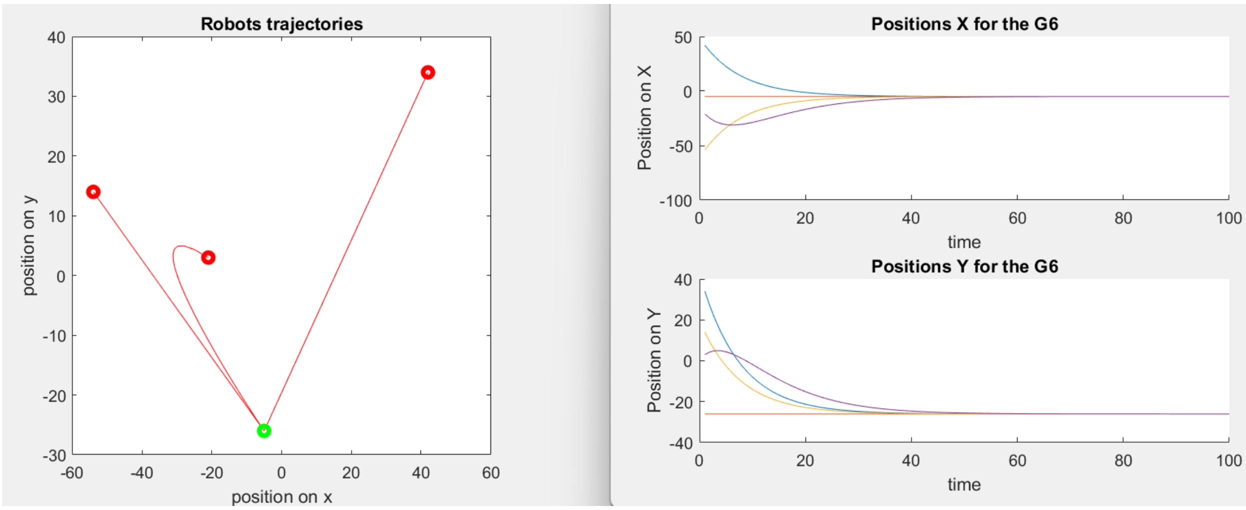


Figure 10: Laplacian Consensus for 4 Robots, G6 Direct Graph

Position Plots

X Positions:

- All robots' x-coordinates converge to the same value
- Convergence occurs around t=40
- Shows smooth exponential-like convergence

Y Positions:

- All robots' y-coordinates also converge to the same value
- Similar convergence time as x-coordinates
- Shows consistent convergence behavior

Why This Happens:

- G6 has better connectivity than G7
- The bidirectional link between nodes 1 and 2 helps information flow
- Node 4 receives information from both nodes 1 and 3
- This structure ensures information from all robots eventually reaches every other robot
- The graph is strongly connected, enabling global consensus

The key difference from G7 is that G6's structure allows information to flow throughout the network more effectively, leading to successful consensus among all robots.

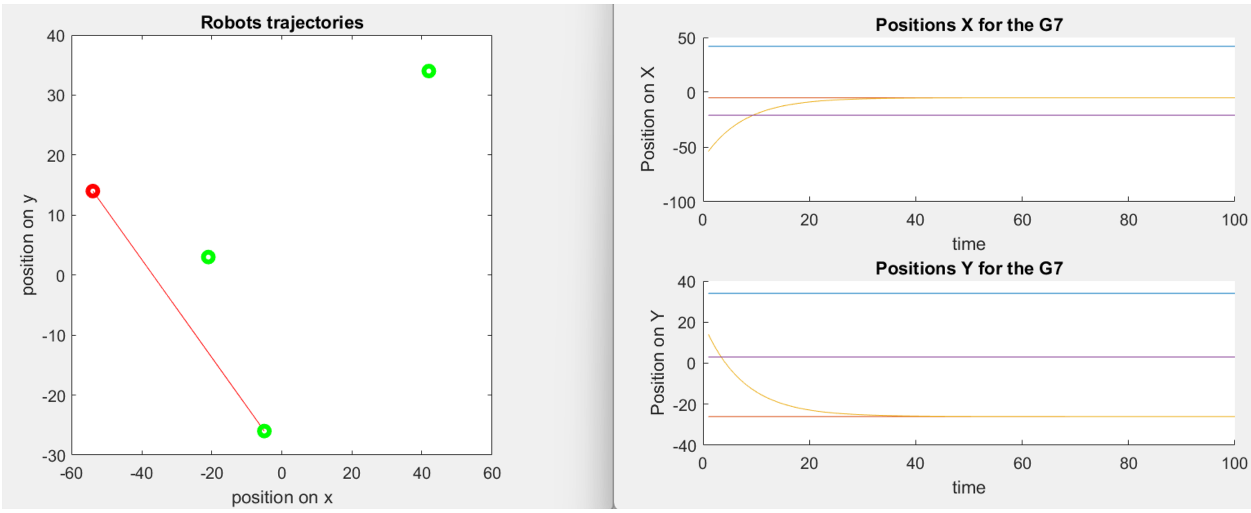


Figure 11: Laplacian Consensus for 4 Robots, G7 Directed Graph

Position Plots

X Positions:

- Different steady-state values for different robots
- Some robots reach consensus in pairs (1-2 form one group)
- Robots 3 and 4 settle at different positions

Y Positions:

- Similar behavior to X positions
- No global consensus is achieved
- Shows the formation of different steady-state groups

Why This Happens:

- Nodes 1 and 2 form a bidirectional connection, so they reach consensus
- Node 3 only receives from 2, so it follows but doesn't influence back
- Node 4 only receives from 3, making it the end of the chain
- This directed structure prevents global consensus

This result demonstrates that for directed graphs, strong connectivity is necessary for achieving global consensus. In G7, the lack of strong connectivity results in partial consensus or clustering of robots.

Algebraic Connectivity (λ2):

- G5: Highest λ2 → Fastest convergence
- G6: Medium λ2 → Medium convergence
- G7: λ2 = 0 → No global consensus

MARS

QUESTION 4

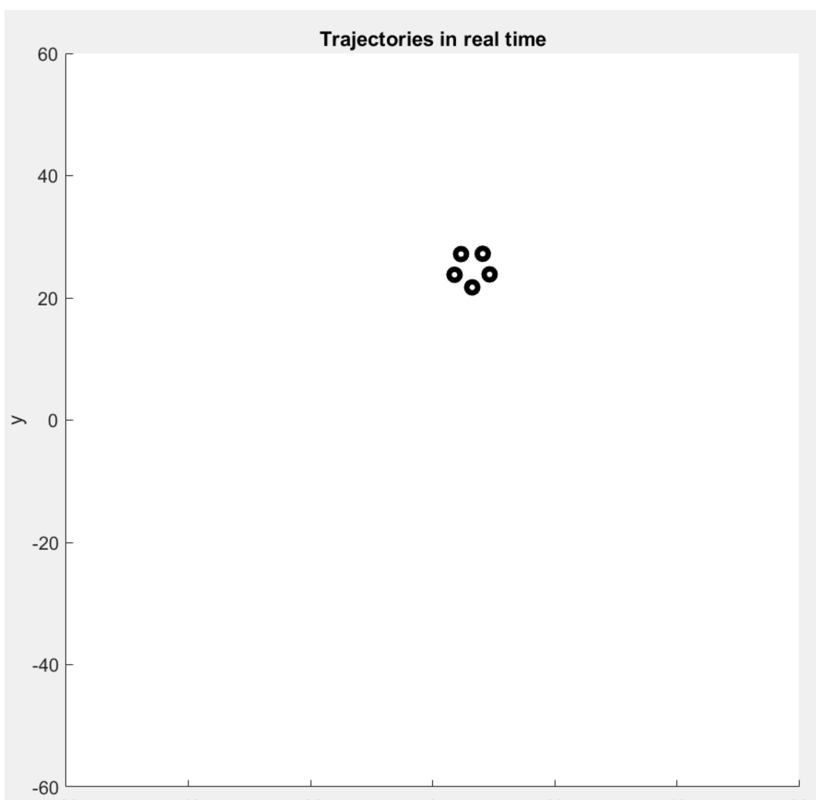
Question 4.1

We can implement a repulsive force between all the instances of the robot. The code below is an adaptation of the repulsion algorithm we calculated in the previous lab. The idea is it is a for loop that takes every other instance of the robot and gives a repulsive force as if each one was an obstacle. Not computationally friendly but it works.

```
if j ~= i
    vector = [X(i,k-1),Y(i,k-1)]-[X(j,k-1),Y(j,k-1)];
    d_obs = vector/(norm(vector)^3);
    Rx(i) = Rx(i) + beta*d_obs(1);
    Ry(i) = Ry(i) + beta*d_obs(2);
end
X(i,k)=X(i,k-1)-Ux(i)*dt+Rx(i)*dt;
Y(i,k)=Y(i,k-1)-Uy(i)*dt+Ry(i)*dt;
```

Question 4.2

Beta = 100



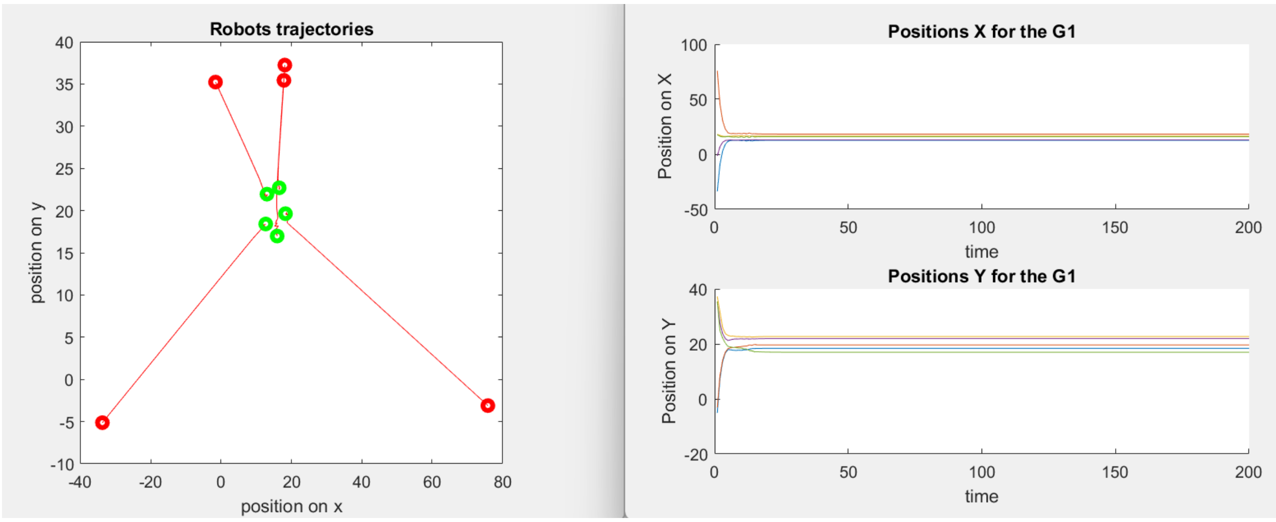


Figure 12: Laplacian Consensus for 5 Robots, G1 undirected Graph with repulsive effect Beta = 100

- Because of repulsive force these robots will not converge to meet same position.

QUESTION 5

Conclusion

In summary, our exploration of consensus algorithms reveals their ability to attract robots toward a shared goal position through agreement among them. Introducing a repulsive force alongside the consensus algorithm imposes limitations, allowing consensus only up to a certain number of robots. As the robot count increases, the interplay between repulsion and attraction induces oscillations, hindering consensus attainment in close proximity. The dynamics shift when examining unconnected graphs, where stability prevails despite the absence of consensus. Rapid convergence characterizes undirected graphs, showcasing the efficiency of consensus in such scenarios. Fundamental to these observations is the role of the Laplacian matrix in capturing the connectivity of the robot network. Graph connectivity emerges as a pivotal factor, influencing the exchange of information crucial for achieving consensus. Understanding these dynamics provides valuable insights into optimizing cooperative behaviors and decision-making within multi-robot systems.