ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

# DYNAMIC FORMATION CONTROL WITH HETEROGENOUS MOBILE ROBOTS

**Kadir ÇİMENCİ**

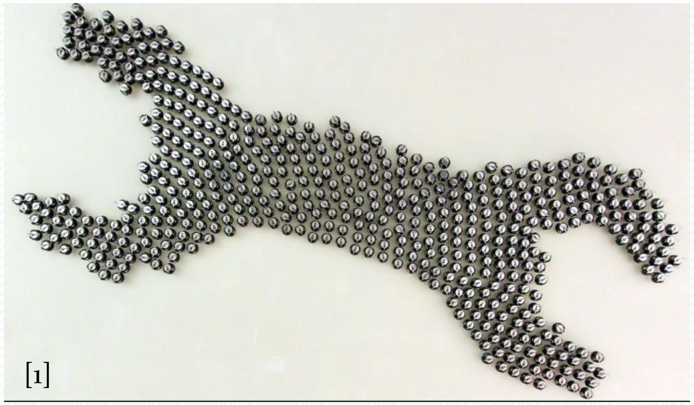
**June 27, 2016**
**Ankara**

# Outline

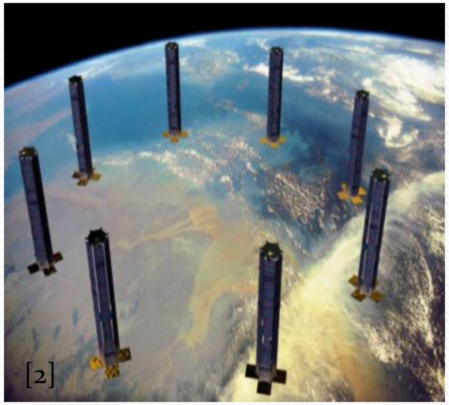
- Introduction
- Motivation
- System Overview
- Local Positioning System Design
- Formation Control System Design
- Results
- Conclusion and Future Works
- References

# Introduction

This thesis work focuses on dynamic adaptation to achieve changes in formation of swarms consisting of heterogenous mobile robots



[1]

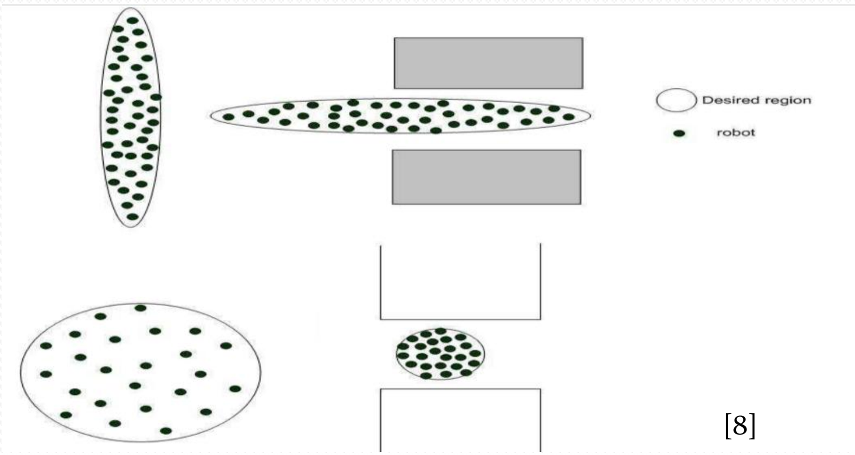


[2]

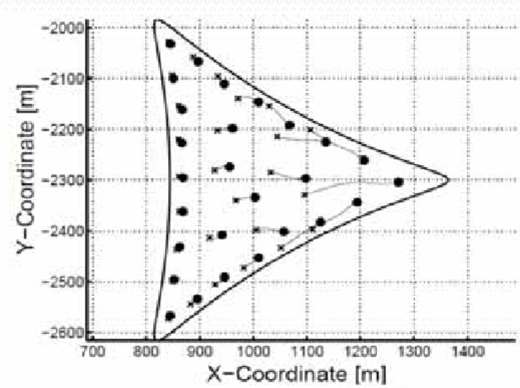
[3]

# Motivation - 1

Formation control solutions are generally implemented with simple geometrical shapes which don't change with time.



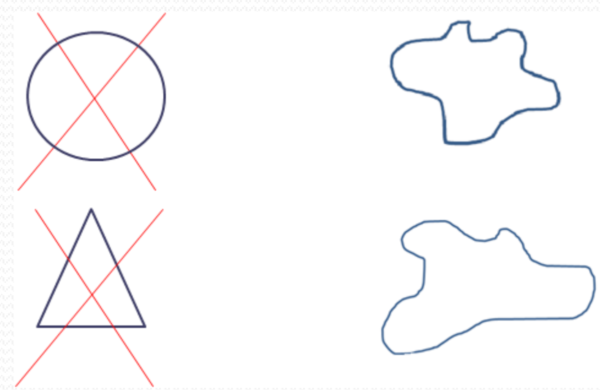

[8]




[9]

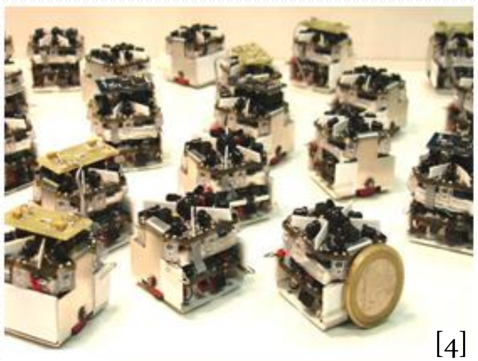
## Our aim

- Designing a formation control system for complex and dynamically changing shapes



# Motivation - 2

The research about the formation control, mainly focuses on swarms with homogenous agents.


[4]


[5]


[6]

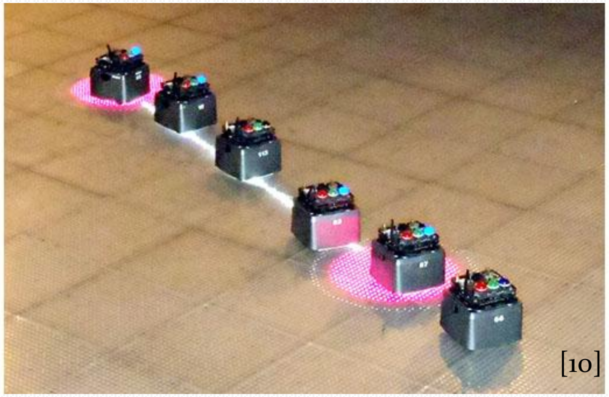

[7]

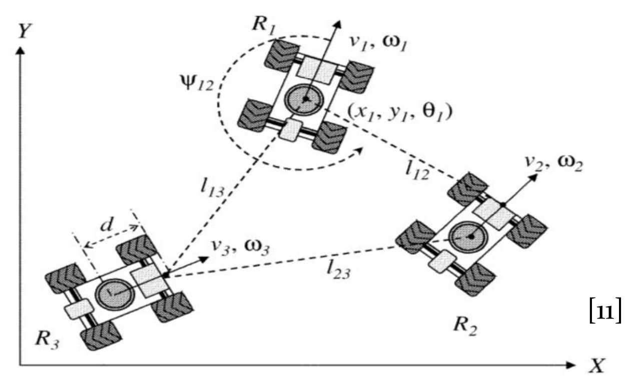
<u>Our aim</u>

•Designing a formation control system with heterogeneous agents

# Motivation - 3

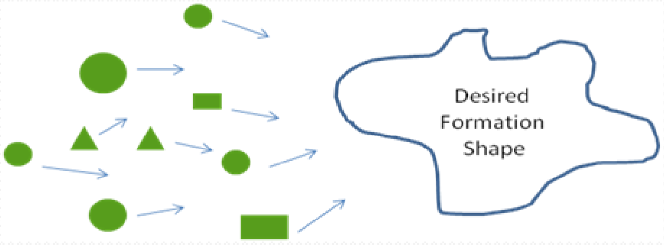
Centralized topologies create single point of failure type systems. We aim to implement a decentralized solution to increase the robustness of the system.



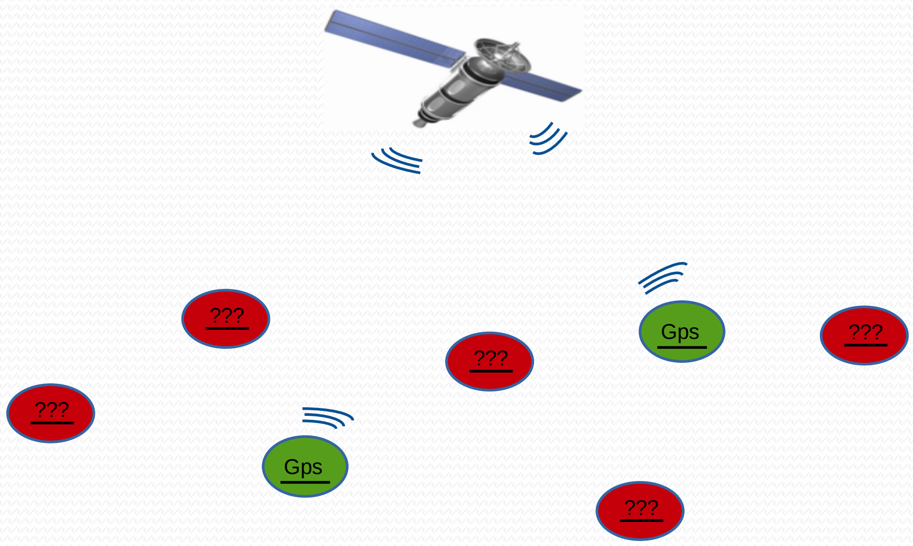

Our aim

- Designing a formation control system with a decentralized topology



# Motivation - 4

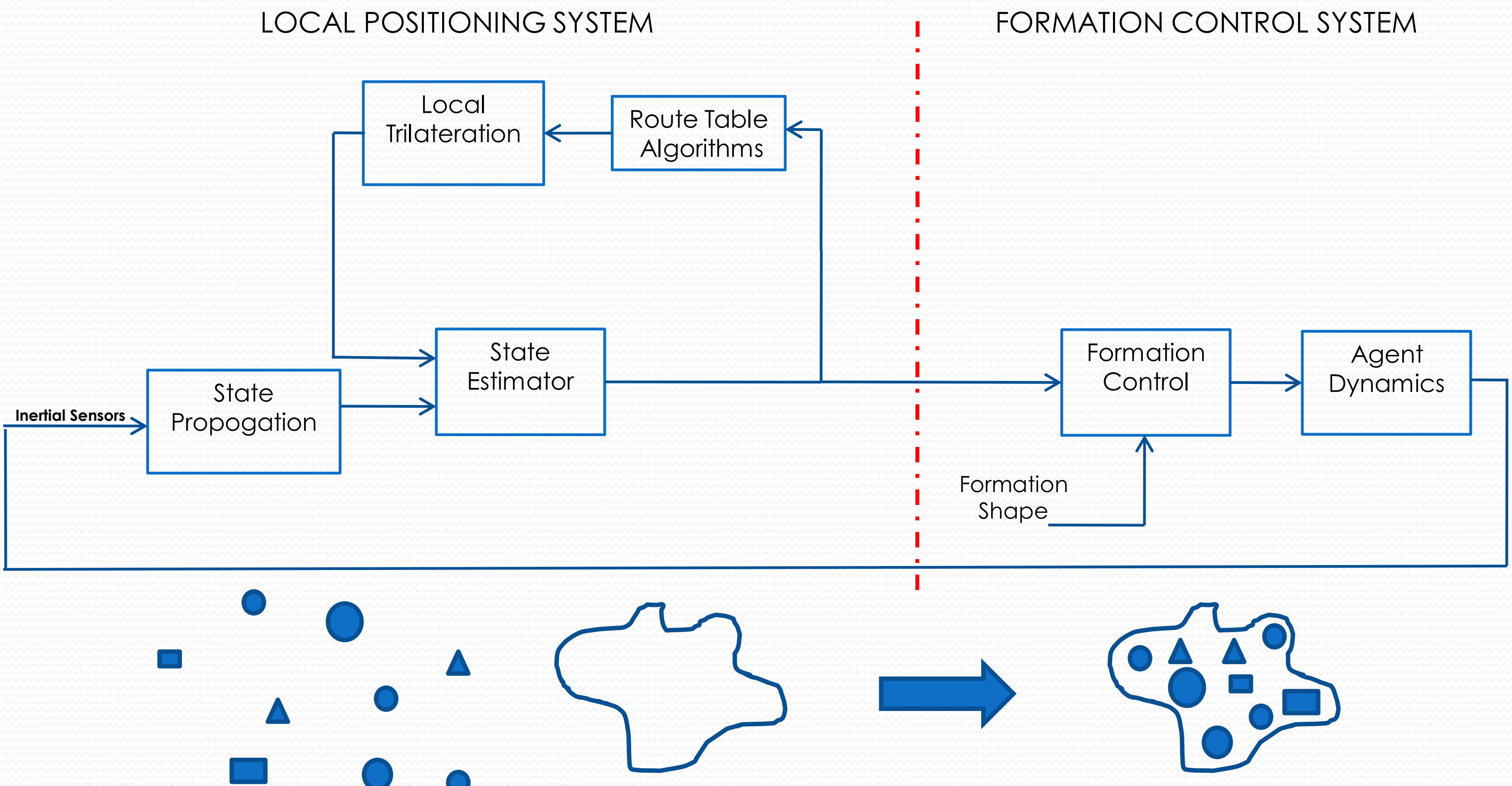
Most of the related works assume that the position data is always available (i.e. can be measured) for each agent in the workspace.




**Our aim**

Designing a local positioning system to provide position information to the second type agents with the help of position beacons.

# System Overview



# Local Positioning System (LPS)

Local positioning system is composed of two main parts.

<u>Local Trilateration</u>

This process calculates the positions of the second type agents with the help of position beacons which are their direct neighbors.

<u>Route Table Determination</u>

This process determines the route tables for agents in the swarm.

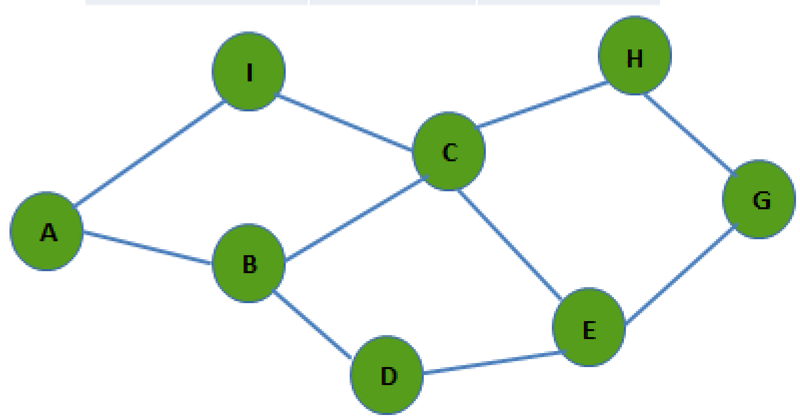
# Local Positioning System (LPS)

## 1)Route Table Determination

- Destination-Sequenced Distance Vector Routing Protocol (DSDV) algorithms are used to create the route tables.

**Route table for agent B**

| Destination | Next Hop | Cost |
|---|---|---|
| A | A | 1 |
| I | A | 2 |
| C | C | 1 |
| H | C | 2 |
| G | C | 3 |
| E | D | 2 |

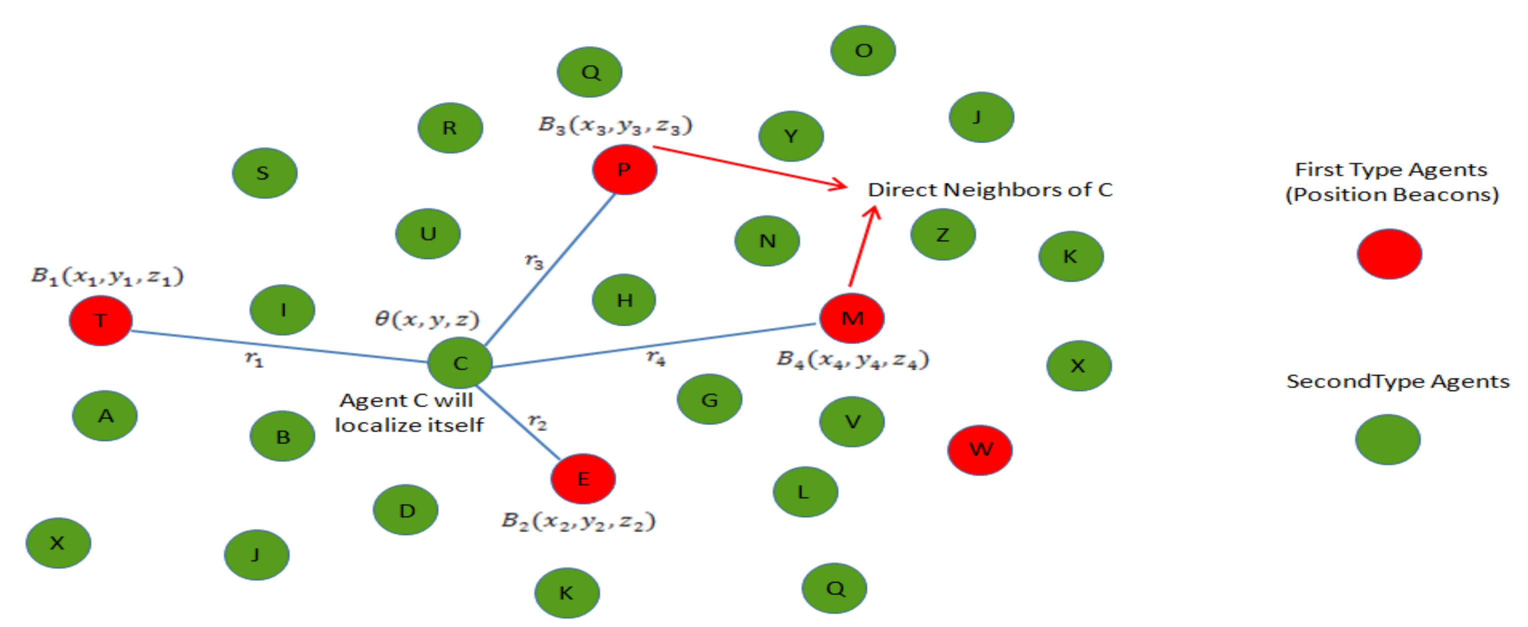


# Local Positioning System (LPS)

## 2)Local trilateration

• Calculates the position of an agent with the help of position beacons which are direct neighbors.

•The solution can be reduced to a problem of $A\vec{x} = \vec{b}$



# Formation Control System

Three different approaches are used to design the formation control system in this thesis work.

Formation Control Strategies

**Potential Field Based Approach**

1) Artificial Forces Method

- Directly calculates control laws based upon potential fields

**Shape Partitioning Based Approaches**

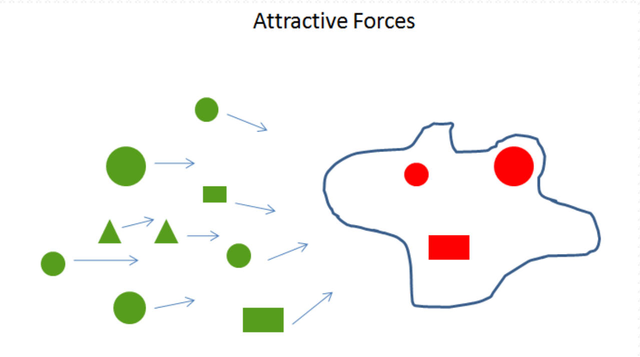
2) Bubble Packing Method
3) Randomized Fractals Method

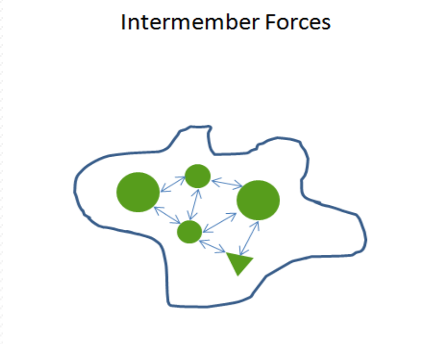
- Partitions the desired formation shape into goal states.
- Control laws are implemented to reach these goal states.

# Formation Control System

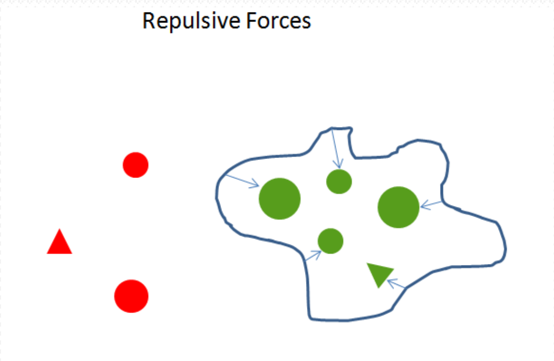
- Artificial Forces Method

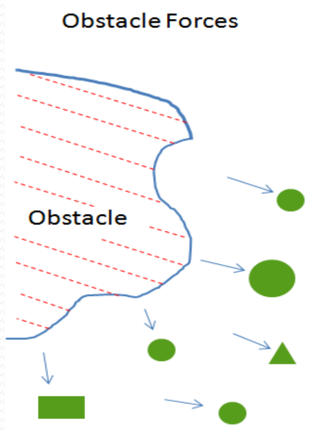
Directly defines the control law for individuals with different potential field components.




$$u_i = u_{att\_i} + u_{rep\_i} + u_{obs\_i} + u_{int\_i}$$



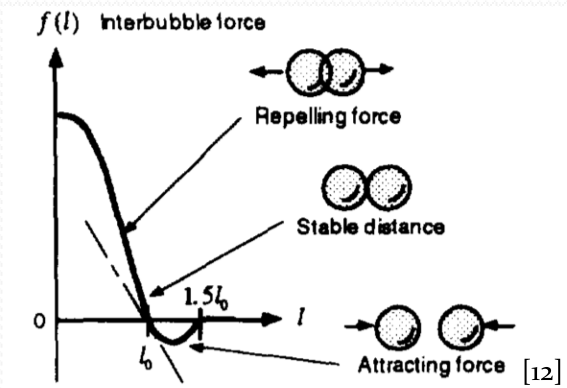



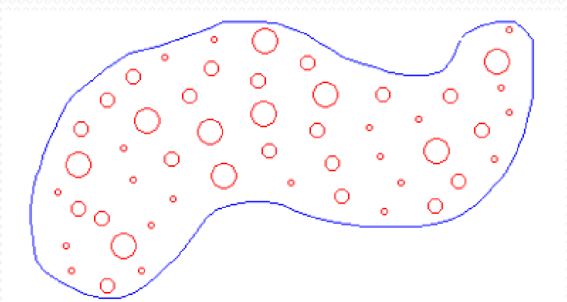
# Formation Control System

- Bubble Packing and Randomized Fractals Methods

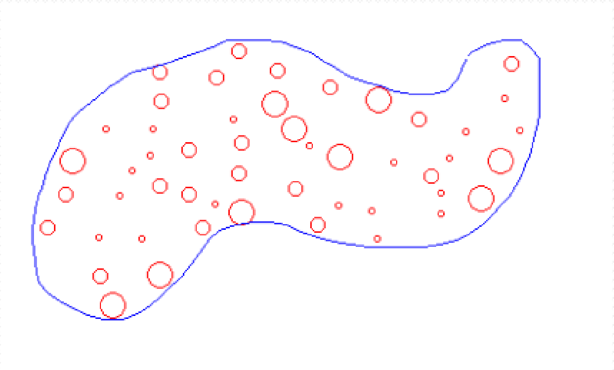
These two methods partition the desired formation shape into goal states with different approaches. The procedure of the assignment of the agents to these goal states are identical.

❖ Bubble Packing



❖ Randomized Factals

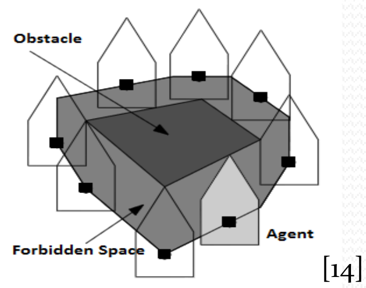


# Formation Control System

- Bubble Packing and Randomized Fractals Methods

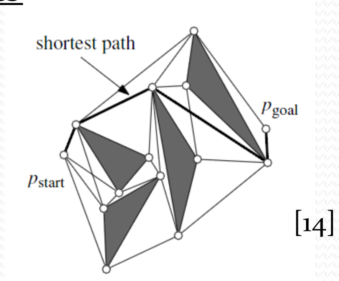
## Decision of Goal States

### *1)Calculation of Free Configuration Space*



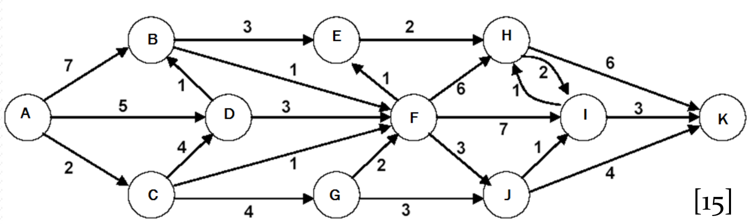

Forbidden Space

### *2)Visibility Graphs*



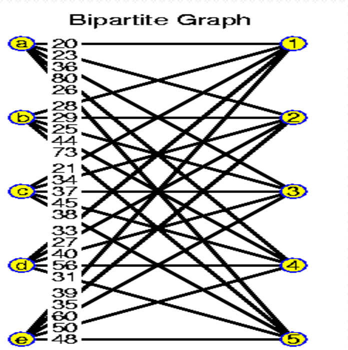

Visibility Graph

### *3)Dijkstra's Algorithm*




Calculation of Minimum Shortest Path From A to K

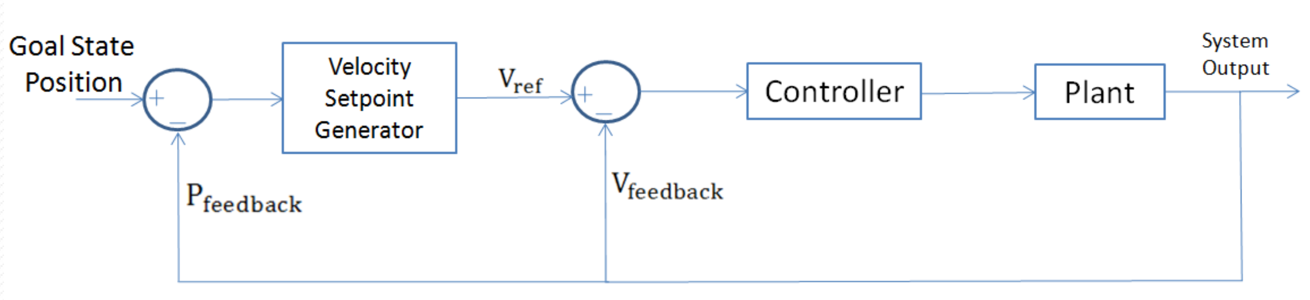
### *4)Hungarian Algorithm (Munkres Assignment Algorithm)*

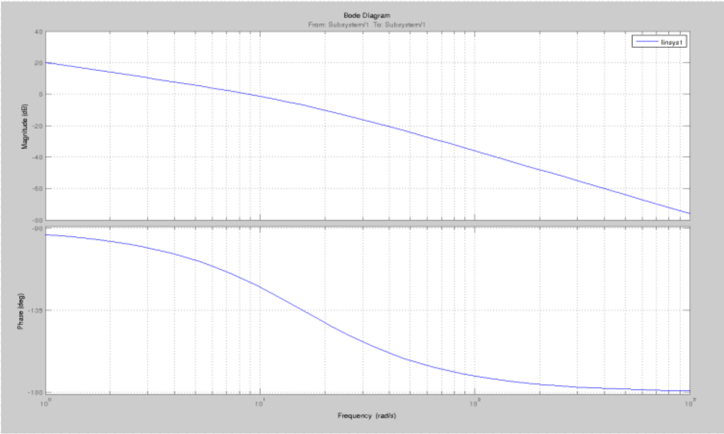


# Formation Control System

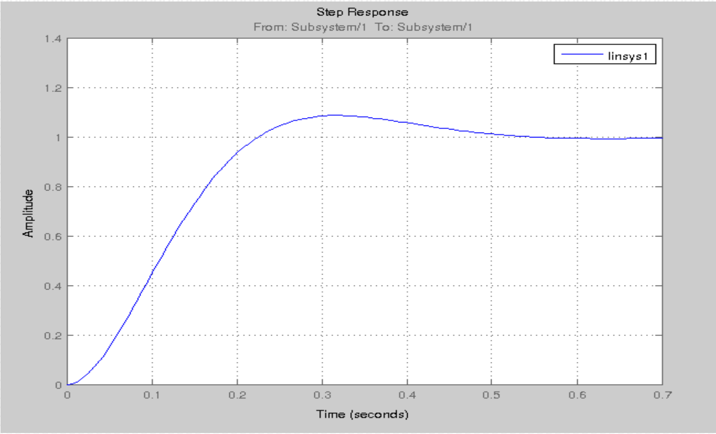
- Bubble Packing and Randomized Fractals Methods

Navigation to Goal States



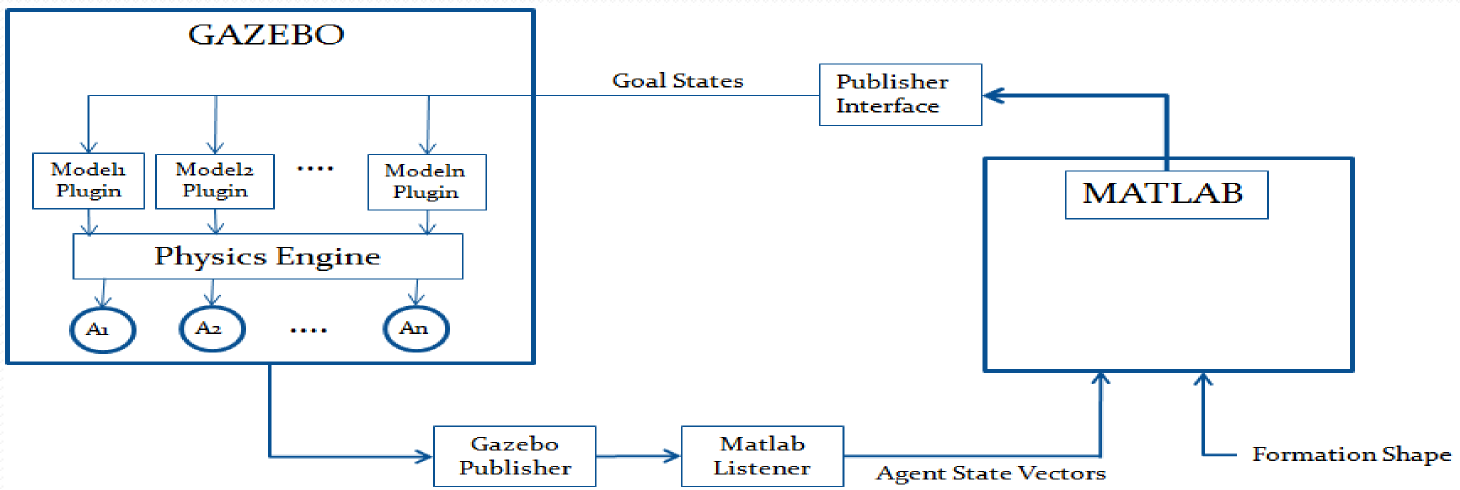

Open loop Bode plots

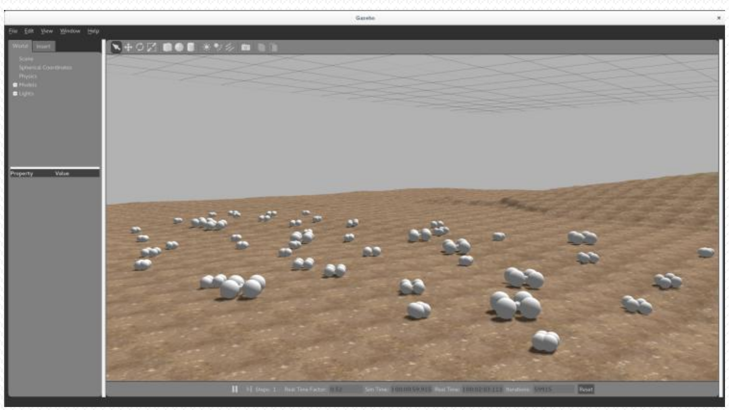



Step response

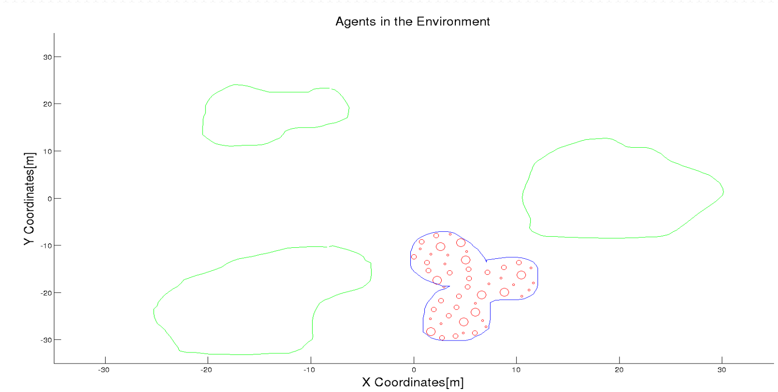
# Results

- Proposed solutions are implemented in a simulation environment.



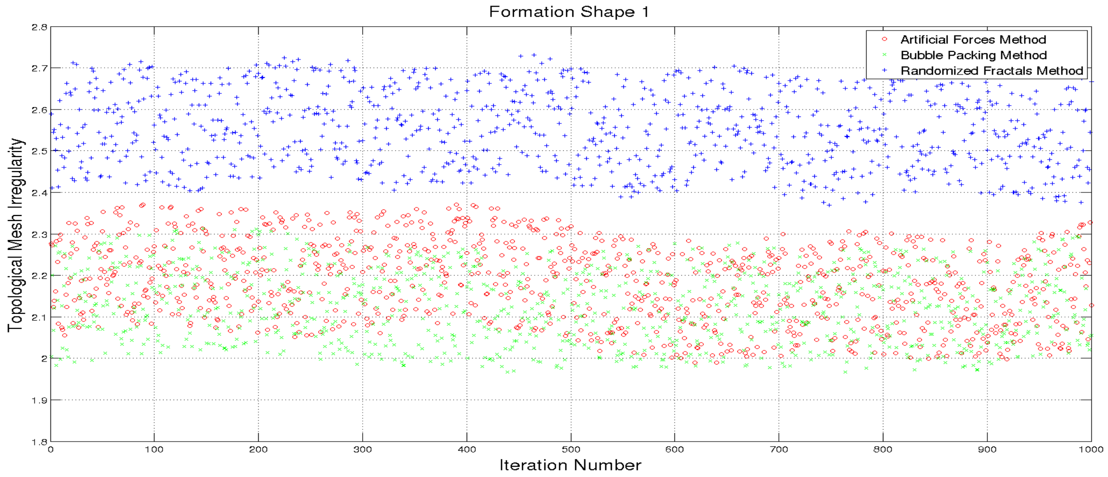



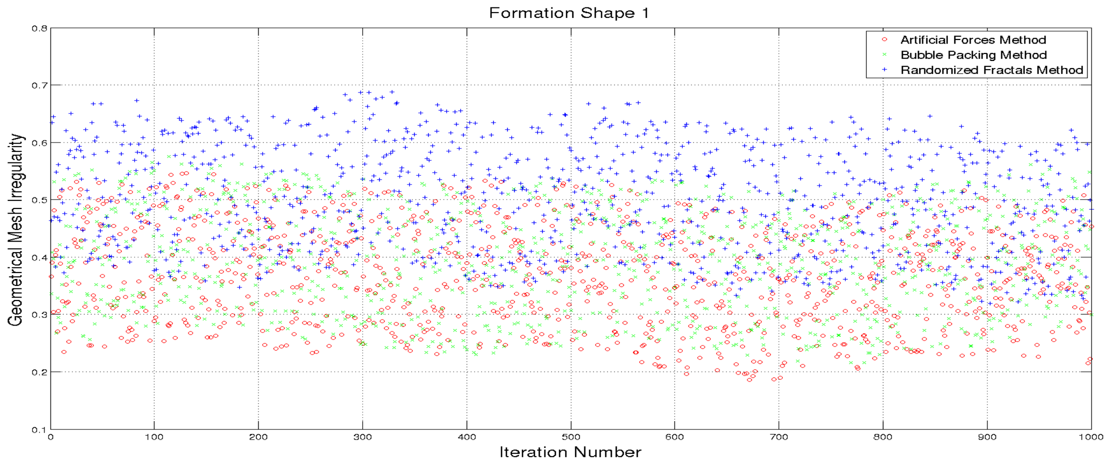
Gazebo Environment



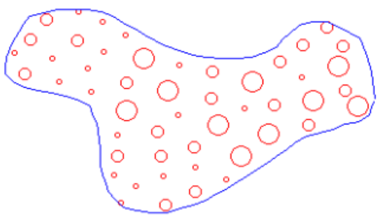

Matlab Environment

# Results – Mesh Quality

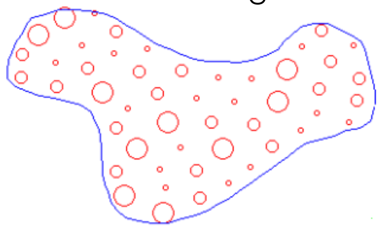


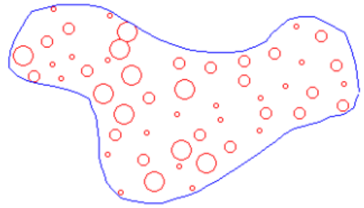

Artificial Forces Method



Bubble Packing Method



Randomized Fractals Method



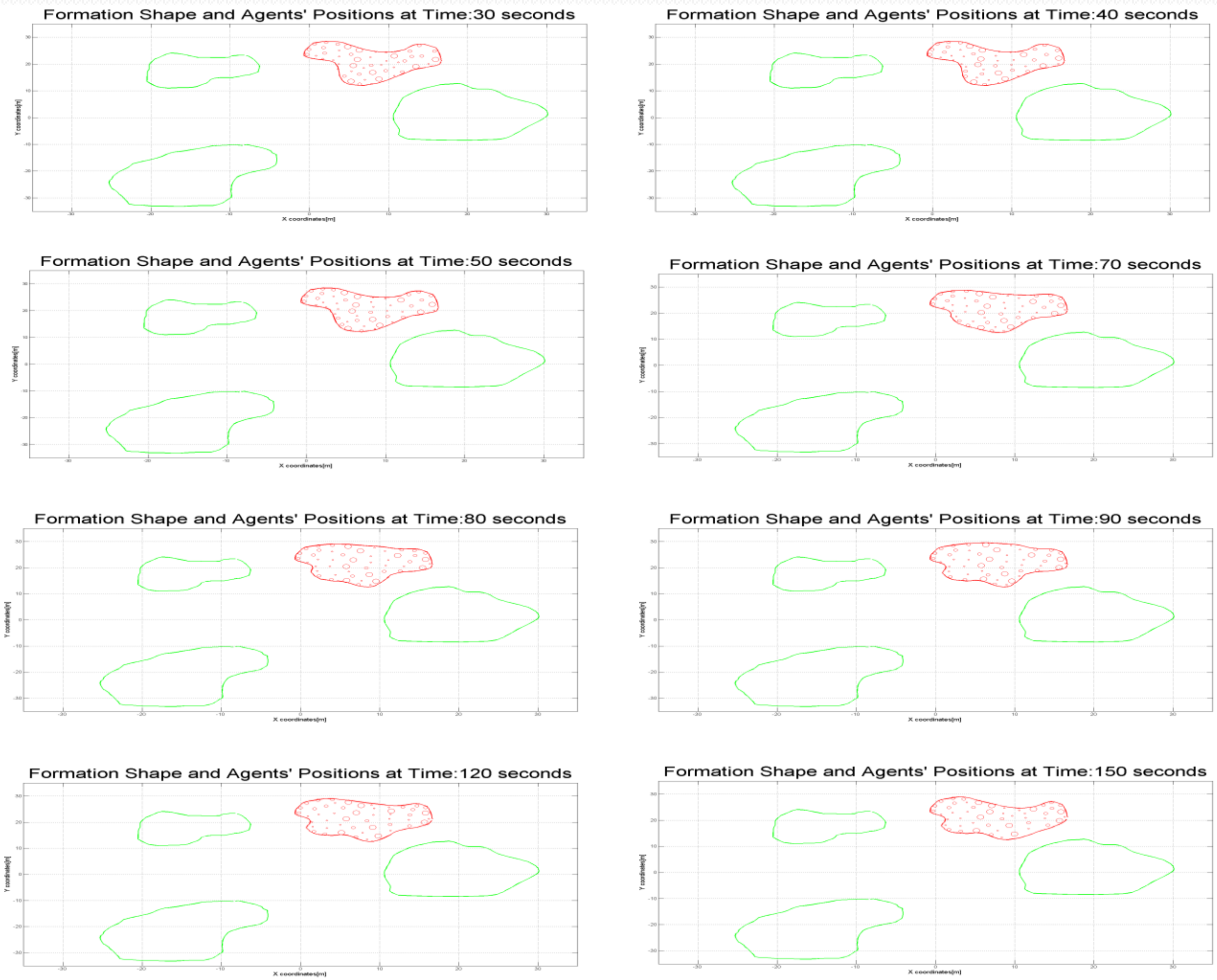
# Results – Comparison of 3 Different Methods

- Comparison of different solution methods are illustrated in Table -1

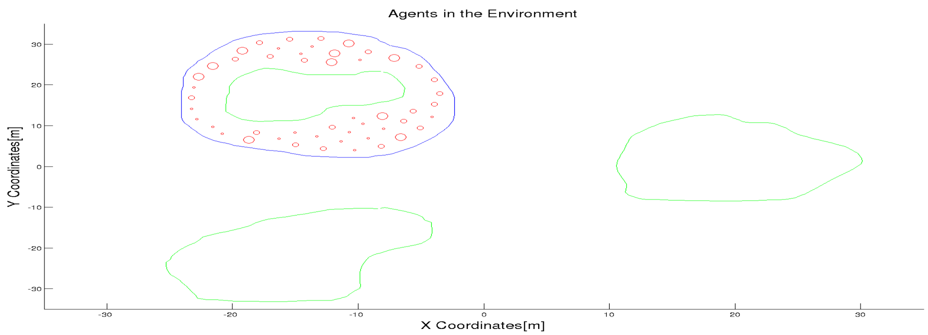
Table -1

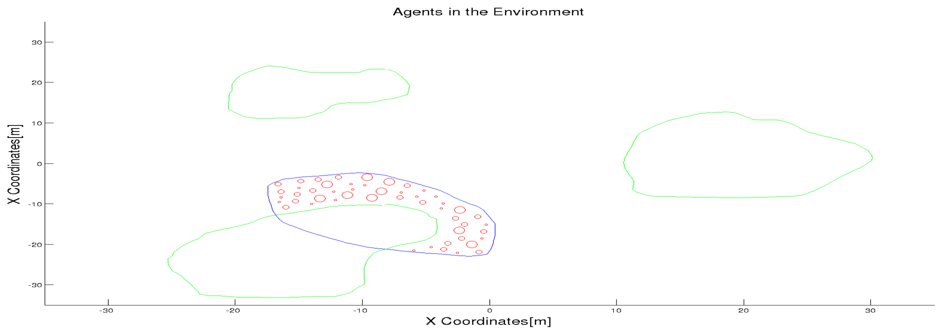
| Method/ Metric | Total Displacement | Settling Time | Mesh Quality |
|---|---|---|---|
| Artificial Forces | | | ✓ |
| Bubble Packing | ✓ | ✓ | ✓ |
| Randomized Fractals | ✓ | ✓ | |

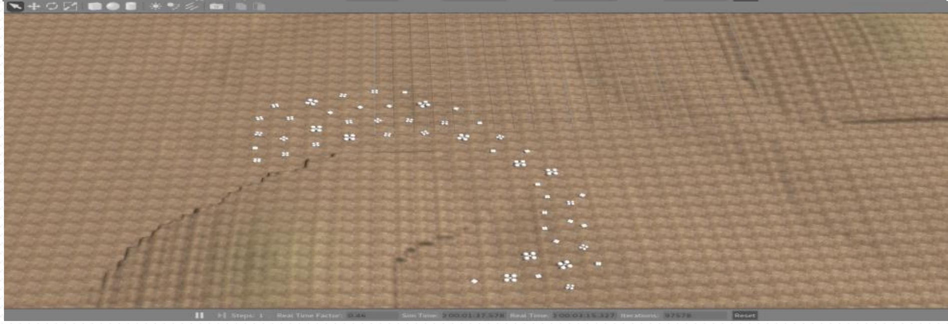
# Results – Dynamically Changing Formation Shapes

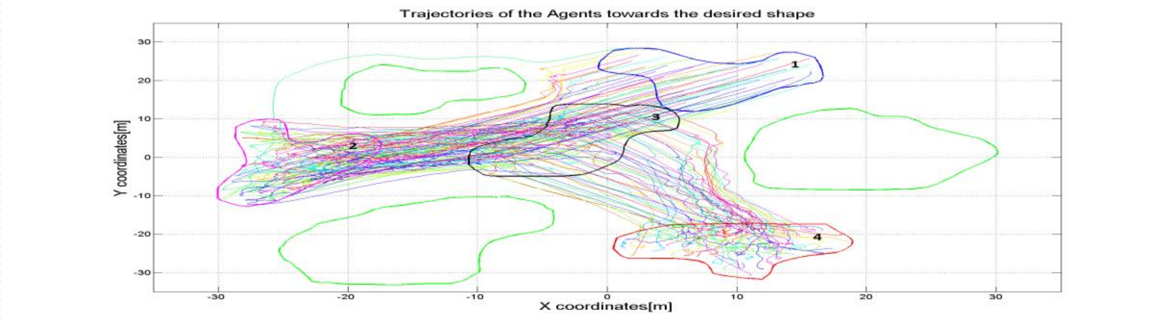


# Results – Various Formation Shape Trials



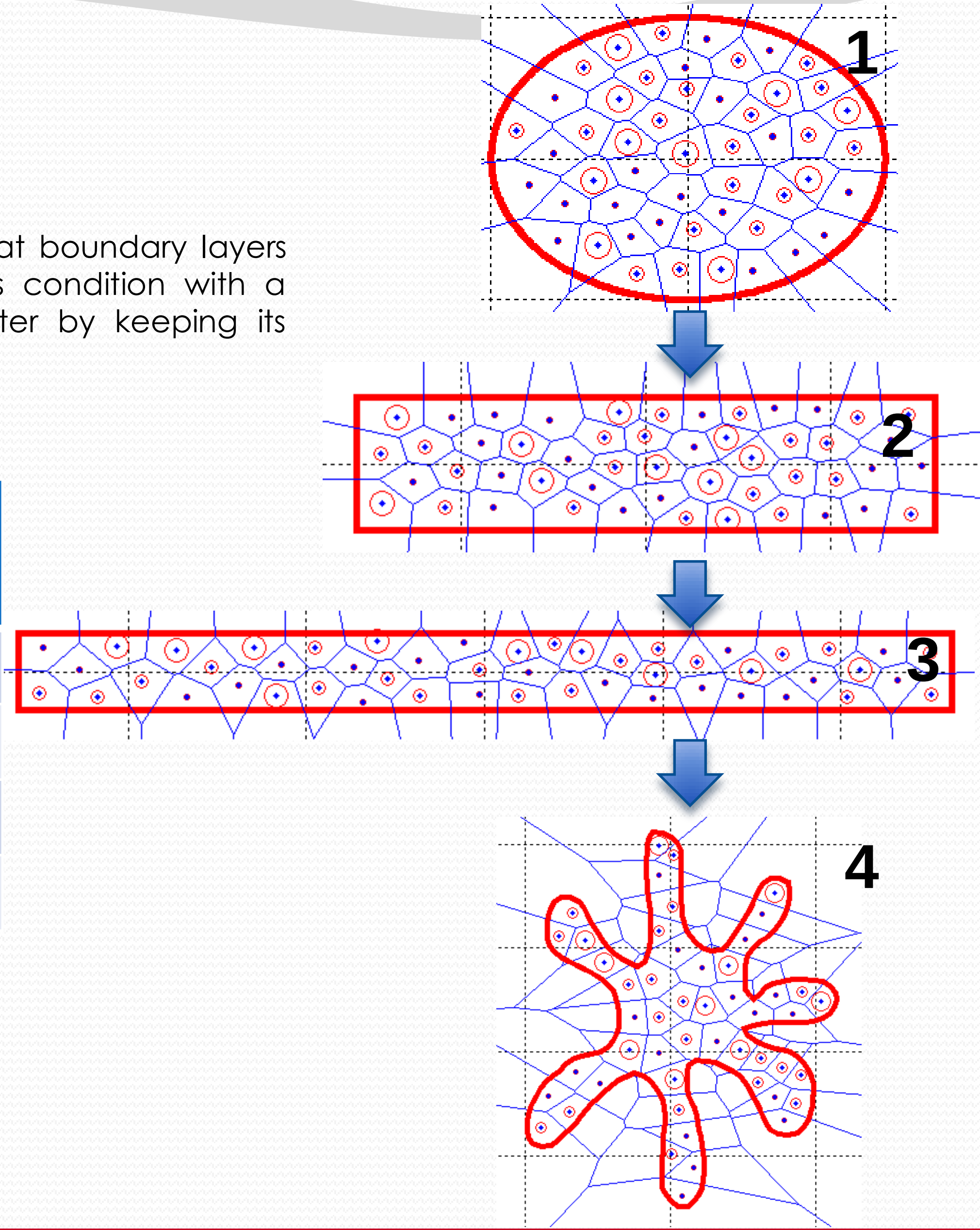





# Results - Sensitivity

• Mesh irregularities are expected to be increasing at boundary layers due to discontinuties[17]. It is possible to see this condition with a formation shape dynamically changing its perimeter by keeping its coverage area constant.

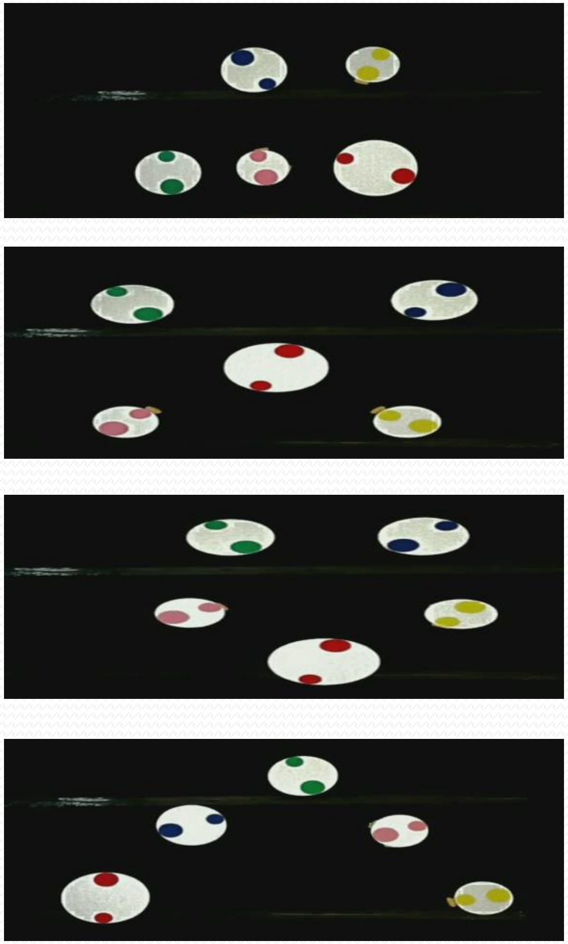


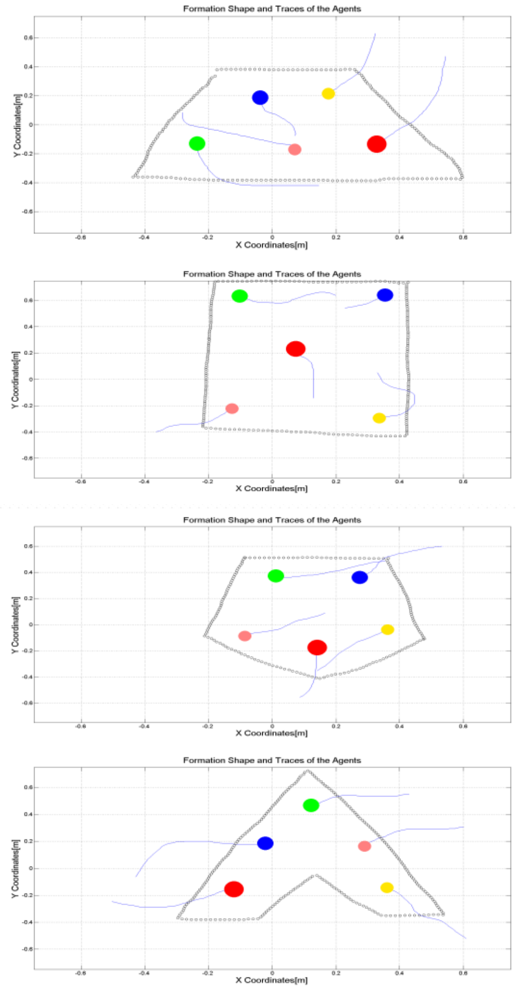

| Shape | Area | Perimeter | Topological Mesh Irregularity |
|---|---|---|---|
| 1 | 314[m^2] | 62,8[m] | 2,46 |
| 2 | 314[m^2] | 82,8[m] | 2,96 |
| 3 | 314[m^2] | 116,82[m] | 3,45 |
| 4 | 314[m^2] | 128,36[m] | 3,95 |

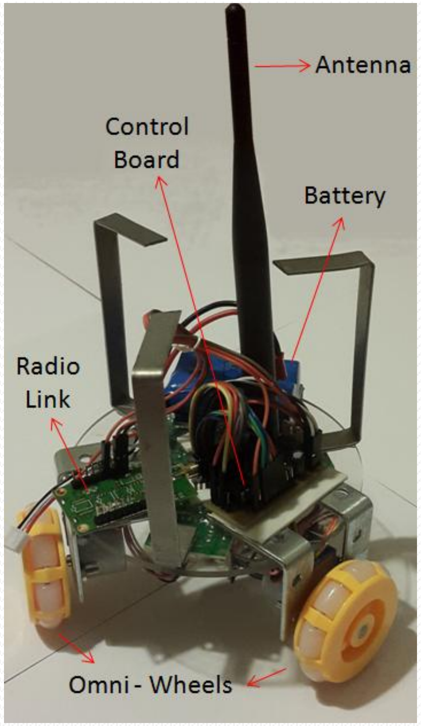
*In 2D, circle has the minimum perimeter to cover a fixed area[18], thus it has the lowest irregularity.

# Results – Hardware Implementation

•Hardware applications which implements the methods discussed in this thesis work are also developed





# Conclusion

• In this project, we have proposed a complete solution, including localization process and formation control.

•We aim to implement a formation control system with heterogenous agents and complex geometrical shapes which are changing dynamically.

• We have proposed different solutions to the formation control problem and discuss about their performance.

•We make hardware applications to demonstrate that the proposed methods can be implemented in real time applications

# Future Works

• Hardware implementation will be done with more agents

• We use heterogenous agents just to cover different formation shapes. We will try to achieve collaborative tasks, by using different functionalities of the agents.

•Obstacle avoidance is implemented with potential fields. To avoid unwanted equilibrium states, obstacle avoidance feature will be implemented with a more appropriate way (e.g. Tangent bug algorithm).

# References


- [1] http://www.eecs.harvard.edu/ssr/projects/progSA/kilobot.html
- [2] Martin N, Klupar P, "Techsat 21 And Revolutionizing Space Missions Using Mıcrosatellıtes",
- [3] http://www.swarmanoid.org/swarmanoid_hardware.php.html
- [4] S Kornienki, O. Kornienko, and Levi. P. Minimalistic approach towards communication and perception in microrobotic swarms. In IEEE International Conference on Intelligent Robots and Systems, 2005.
- [5] Farshad Arvin, John Murray, Licheng Shi, Chun Zhang, and Shigang Yue. Development of an autonomous micro robot for swarm robotics. In IEEE International Conference on Mechatronics and Automation, 2014.
- [6]Touraj Soleymani, Vito Trianni, Michael Bonani, Francesco Mondada, and Marco Dorigo. Bio-inspired construction with mobile robots and compliant pockets. Robotics and Autonomus Systems, 74:340–350, 2015.
- [7] Roderich Grof, Michael Bonani, Mondada Francesco, and Marco Dorigo. Autonomous self-assembly in a swarm-bot. 22:1115–1130, 2006.
- [8] S.P. Hou, C.C. Cheah, and J.J.E. Slotine. Dynamic region following formationcontrol for a swarm of robots. In ICRA, 2009.
- [9] Samitha Ekanayake and Pubudu Pathirana. Formations of robotic swarm: An artificial force based approach. International Journal of Advanced Robotic Systems,7:173–190, 2010
- [10] http://electronicdesign.com/systems/advanced-robots-swarm-nyc-s-museum-math
- [11] Aveek Das, Rafael Fierro, Vijay Kumar, James Ostrowski, John Spletzer, and Camilla Taylor. A vision-based formation control framework. IEEE Transactions on Robotics and Automation, 18:813–825, 2002.
- [12] Kenji Shimada and David Gossard. Bubble mesh: Automated triangular meshing of non-manifold geometry by sphere packing. In ACM Symposium on Solid Modeling and Applications, 1995.
- [13]John Shier and Paul Bourke. An algorithm for random fractal filling of space.Computer Graphics Forum, 32:89–97, 2013.
- [14] Mark Berg, Otfried Cheong, Kreveld. Marc, and Marc Overmars. ComputationalGeometry. Springer, 1998.
- [15] https://nevraa.wordpress.com/2012/08/08/dijkstra-algoritmasi-ile-en-kisa-yol-bulma/
- [16] Jop Frederik Sibeyn. Graph algorithms. https://www8.cs.umu.se/ jopsi/dinf504/chap14.shtml, last visited on April 2016.
- [17] Richard, Pletcher and John C. Tannehill, Computational Fluid Mechanics and Heat Transfer, Third Edition
- [18] B.V. Ramana, Higher Engineering Mathematics, The McGraw Hill Companies

Thank you for your attention.

# Formation Control System

- Bubble Packing and Randomized Fractals Methods

Velocity Controller

The dynamical system of agents is augmented with an artificial error state, to design an State feedback with LQR controller;

$$\begin{bmatrix} \dot{v} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} -b/m & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} v \\ e \end{bmatrix} + \begin{bmatrix} 1/m \\ 0 \end{bmatrix} F_{net} \qquad y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} v \\ e \end{bmatrix}$$

Q and R matrices used in solving Riccati equations,

$$Q = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix};\ R = \rho r_1 \qquad q_1 = \frac{1}{t_{s_1}(x_{1max})^2}\ ;\ q_2 = \frac{1}{t_2(x_{2max})^2}\ \text{and}\ r_1 = \frac{1}{(u_{1max})^2}$$

where,

$t_{s_i}$: desired settling time for $x_i$

$\rho$: tradeoff regulation vs control effort

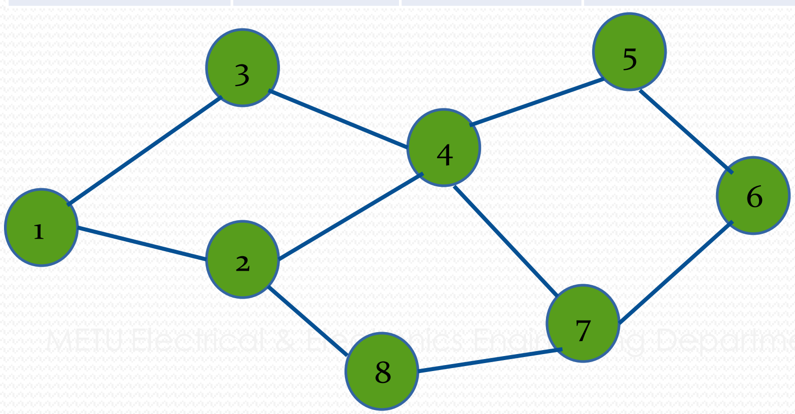
# Local Positioning System (LPS)

## 2)Route Table Determination

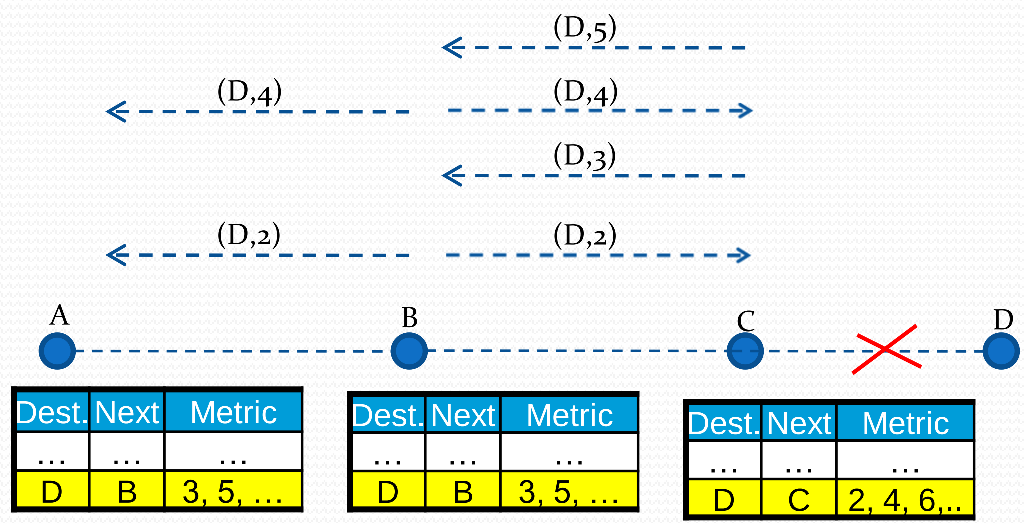
- DSDV is a table driven routing scheme based on Bellman Ford algorithm
- Used to create wireless mesh networks and ad-hoc mobile networks
- Solves routing loop problem in route table algorithms

**Route table for agent 2**

| Destination | Next Hop | Metric | Dest. Seq. No |
|---|---|---|---|
| 1 | 1 | 1 | 123 |
| 3 | 3 | 2 | 516 |
| 4 | 4 | 1 | 212 |
| 5 | 4 | 2 | 168 |
| 6 | 8 | 3 | 372 |
| 7 | 8 | 2 | 432 |



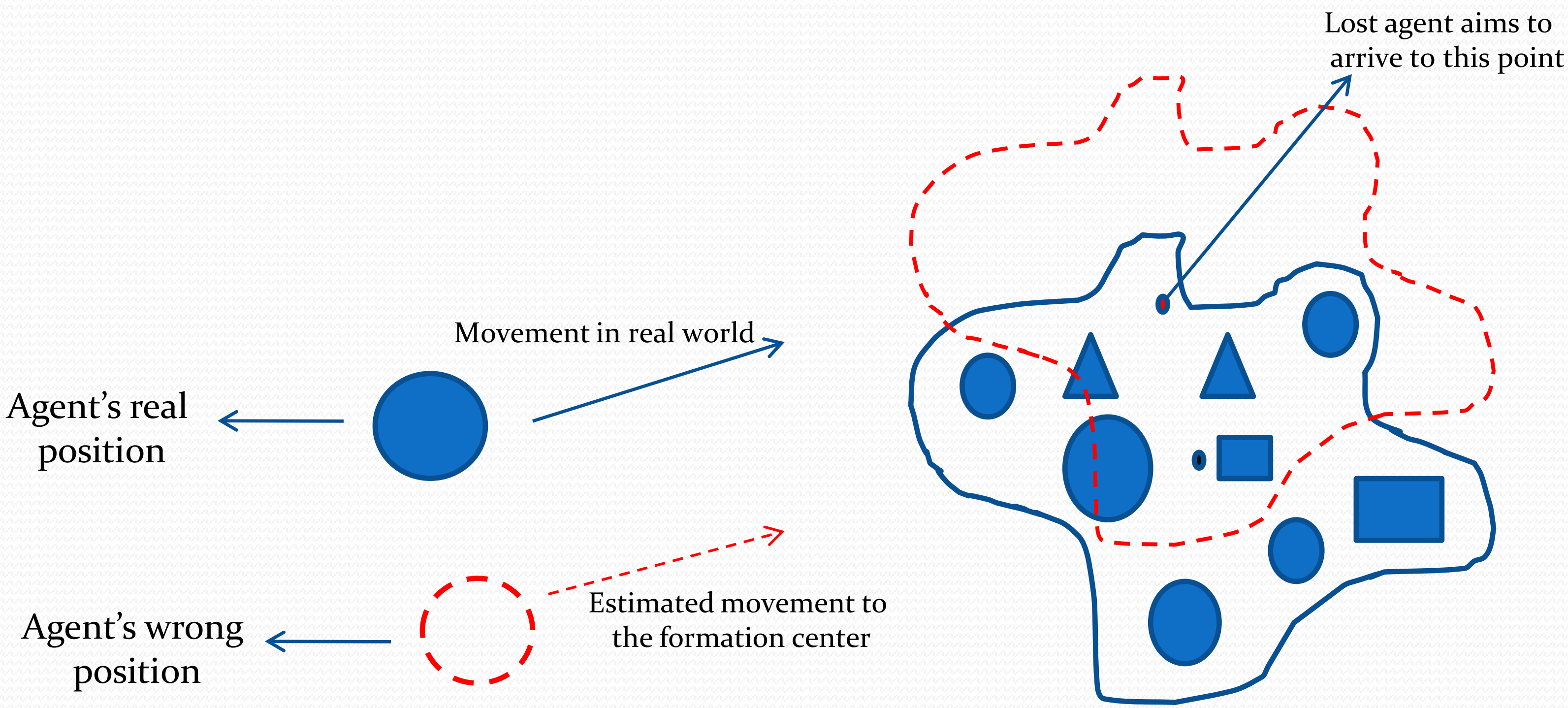

**Routing Loop Problem**



# Local Positioning System (LPS)

## 1)Local trilateration

### Return to Home Mode



# Local Positioning System (LPS)

## 1)Local trilateration

The solution of the position(P(x,y,z)) with the help of positions of neighbors can be reduced to a problem of;

$$A\vec{x} = \vec{b}$$

We have an A matrix with a dimension of [n-1] x 2 (where n is the number of neighbors). There are three options for the solution of the problem related with the condition of A matrix,

1) $\hat{x} = A^{-1}.b$ , unique solution (if there are 3 neighbors and A is full column rank matrix)
2) $\hat{x} = (A^T A)^{-1} A^T b$ , minimum norm solution (if there are more neighbors and A is full column rank matrix
3) Find the minimum error/norm solution with nonlinear least squares method, if rank(A) = 1

# Formation Control System

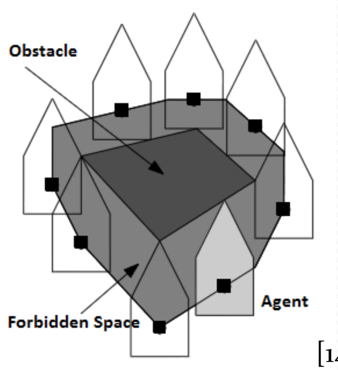
- Bubble Packing and Randomized Fractals Methods

## Decision of Goal States

### *1)Calculation of Free Configuration Space*

$$C(R_i) = C_{free}(R_i, S) + C_{forb}(R_i, S)$$

Forbidden Space : $S_1 \oplus S_2 := \{p + q : p \subset S_1, q \subset S_2\}$



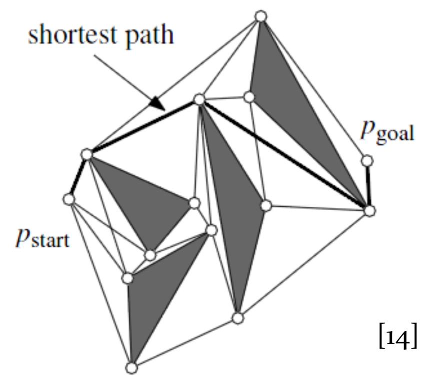

[14]

Forbidden Space

### *2)Visibility Graphs*

The shortest path between a start and goal among a set S of augmented polygonal obstacles consists of arcs of the visibility graph [14]

$$\gamma_{vis}(S^*) \text{ where } S^* := S \cup \{p_{start}, p_{goal}\}$$




[14]

Visibility Graph

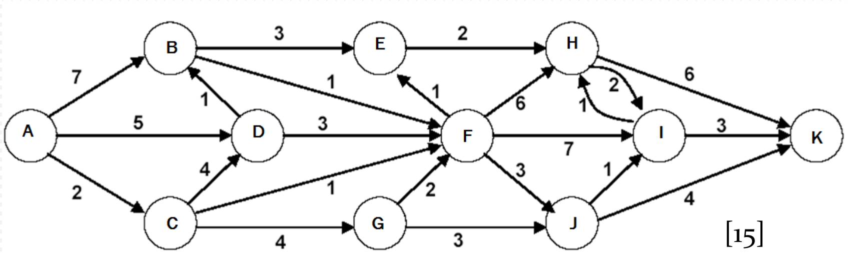
# Formation Control System

- Bubble Packing and Randomized Fractals Methods

## Decision of Goal States

### *3)Dijkstra's Algorithm*

Dijkstra's algorithm is a tree search algorithm for finding the shortest paths between nodes in a graph

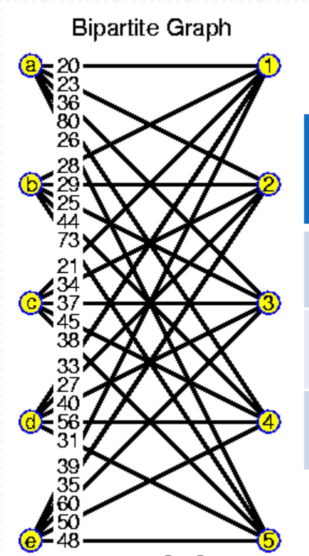



[15]

Calculation of Minimum Shortest Path From A to K

### *4)Hungarian Algorithm (Munkres Assignment Algorithm)*

The shortest path between a start and goal among a set S of augmented polygonal obstacles consists of arcs of the visibility graph




[16]

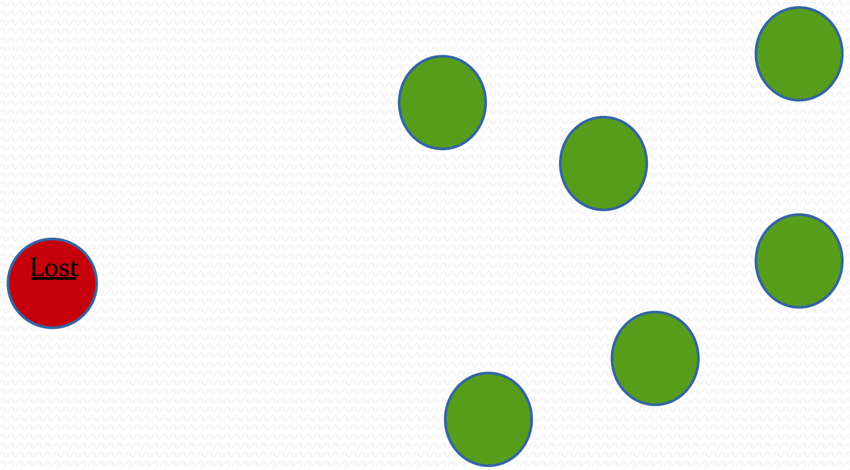
| | Clean Bathroom | Sweep Floors | Wash Windows |
|---|---|---|---|
| Jim | $3 | $2 | $7 |
| Steve | $2 | $5 | $3 |
| Alan | $4 | $3 | $2 |

# Local Positioning System (LPS)

## 2)Local trilateration

### Lost agent handling rules;

- An agent is called ‘lost’ when it doesn’t have minimum 3 neighbors
- If an agent is lost it cannot enter the localization process, and it enters ‘Lost’ mode in which it is directed to the center of formation shape.



# Results

Local Positioning System (LPS)