ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

DYNAMIC FORMATION CONTROL WITH HETEROGENOUS MOBILE ROBOTS

Kadir ÇİMENCİ

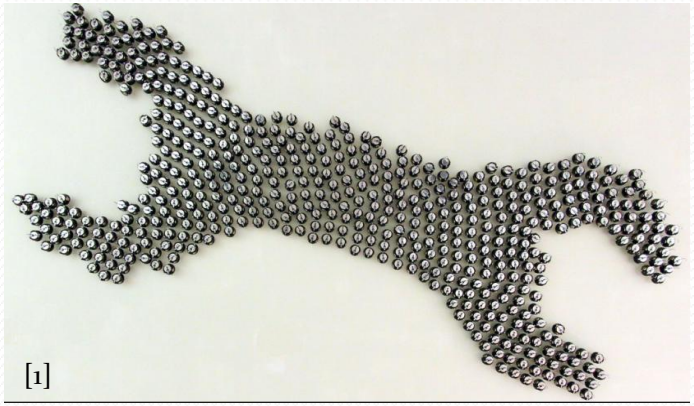
June 27, 2016
Ankara

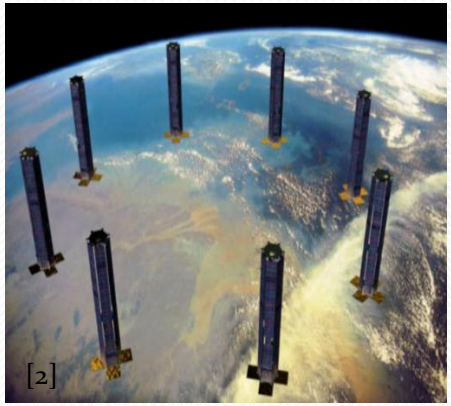
Outline

- 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

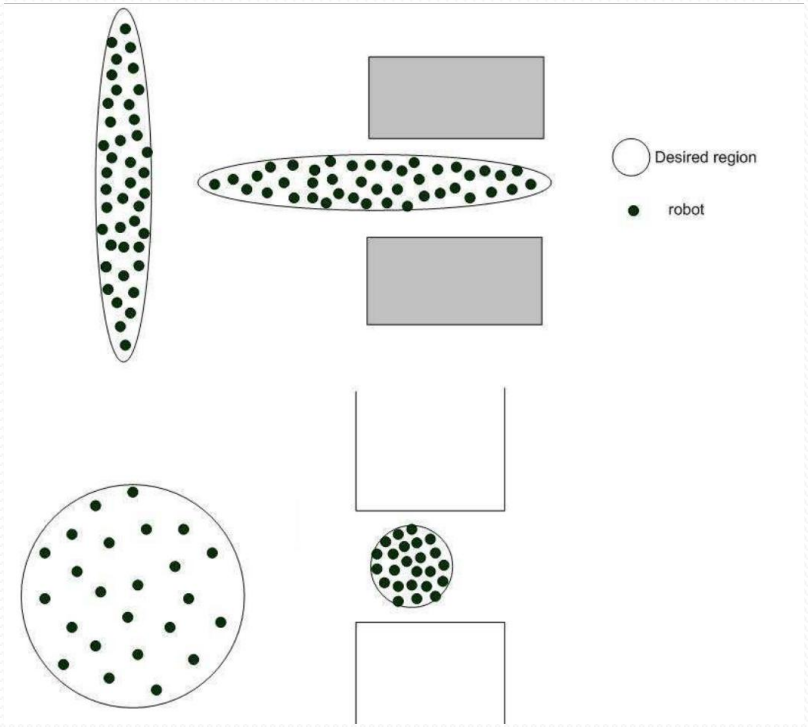

[1]


[2]

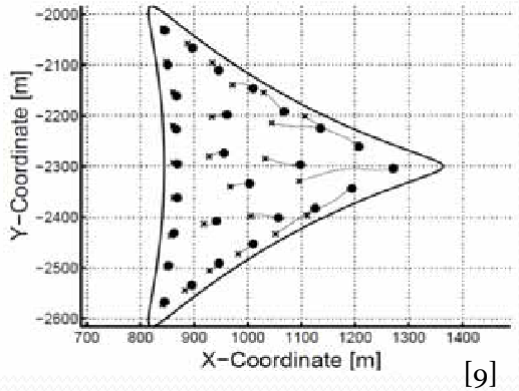

[3]

Motivation

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

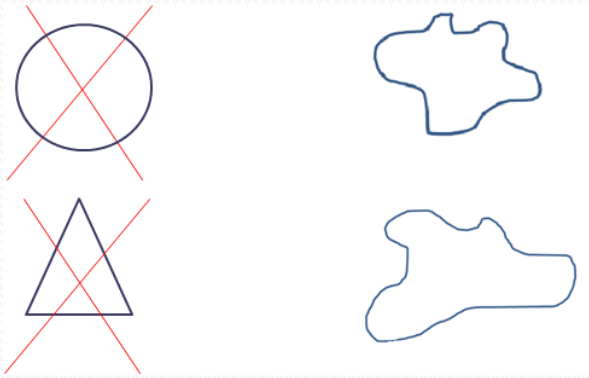


[8]



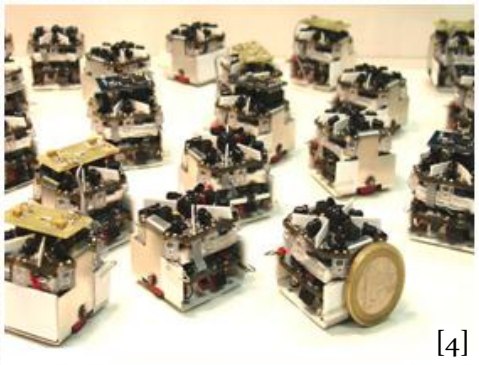
[9]

Our aim



Motivation

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


[4]

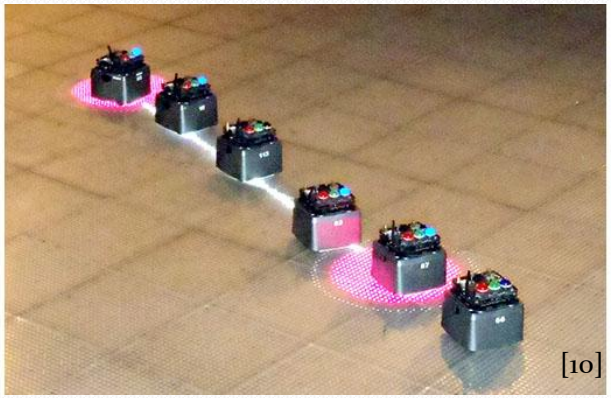

[5]

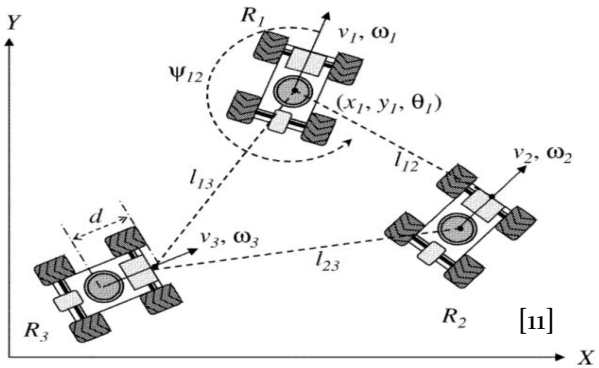

[6]


[7]

Motivation

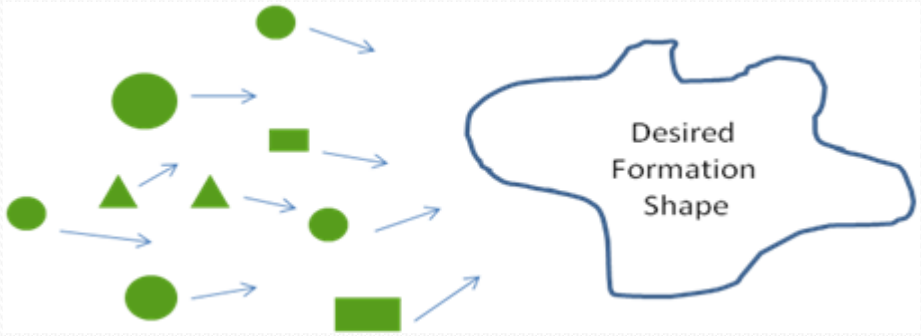
Centralized topologies create single point of failure type systems. We aim to implement a decentralized solution to increase the robustness of the system.


[10]



[11]

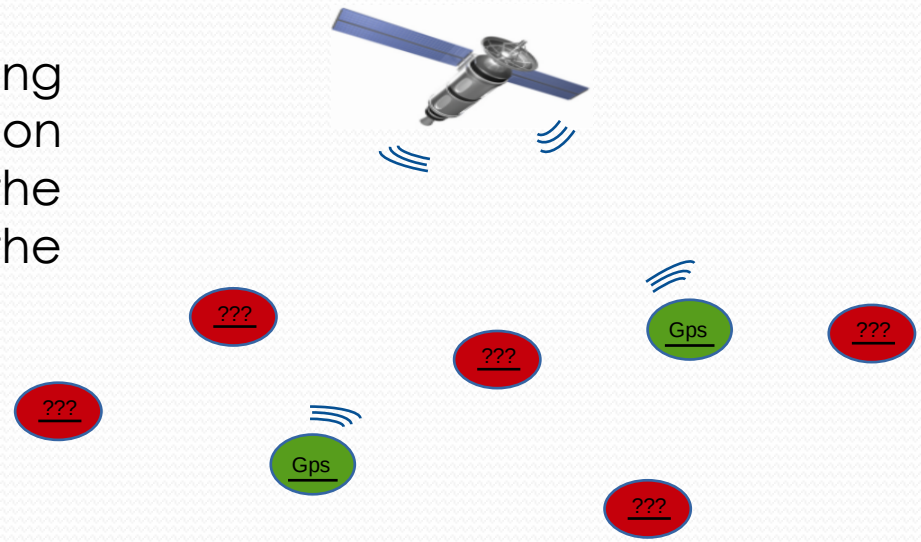
Our aim



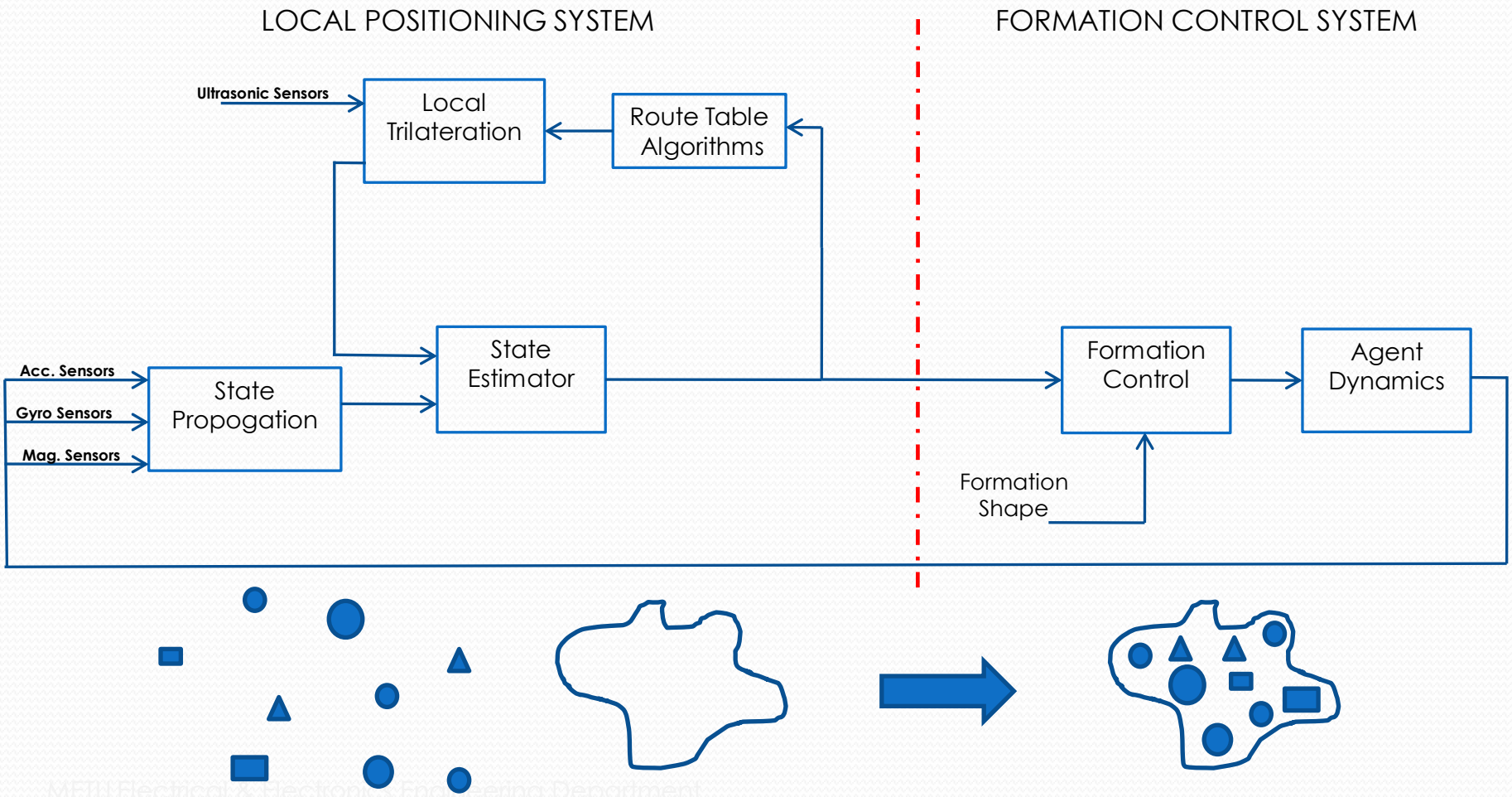
Motivation

One of the most important characteristic of an agent in the swarm is its simplicity and limited sensor & communication capability [12]. Most of the related work assumes that the position data is always available (i.e. can be measured) for each agent in the workspace.

Our aim is to design a local positioning system to provide position information to the agents which do not have the ability to measure its position in the workspace



System Overview



Local Positioning System (LPS)

Local positioning system is composed of two main parts.

1)Local Trilateration

This process calculates the position data for the agents which don't have position sensors by using local neighbors.

2)Route Table Determination

This process creates a mesh network which is a communication backbone in the swarm. It also determines the clusters around the position beacons and provides the order for the local trilateration process.

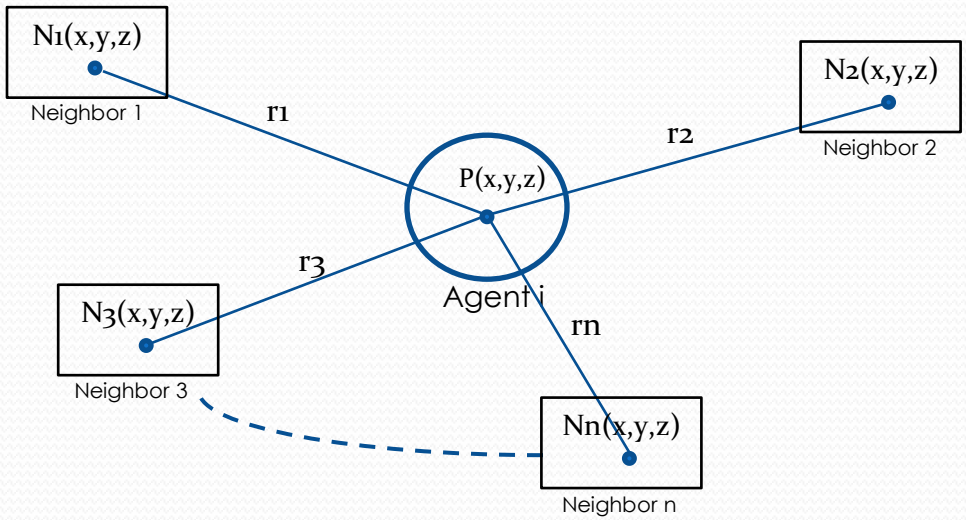
Local Positioning System (LPS)

1)Local trilateration

- Requires at least 3 neighbors.
- Calculates the position of P(x,y,z) with the help of positions of the neighbors.
- The solution of the position P(x,y,z) be reduced to a problem of $\vec{A}\vec{x} = \vec{b}$

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

- Unique Solution
- Minimum Norm Solution
- No Solution

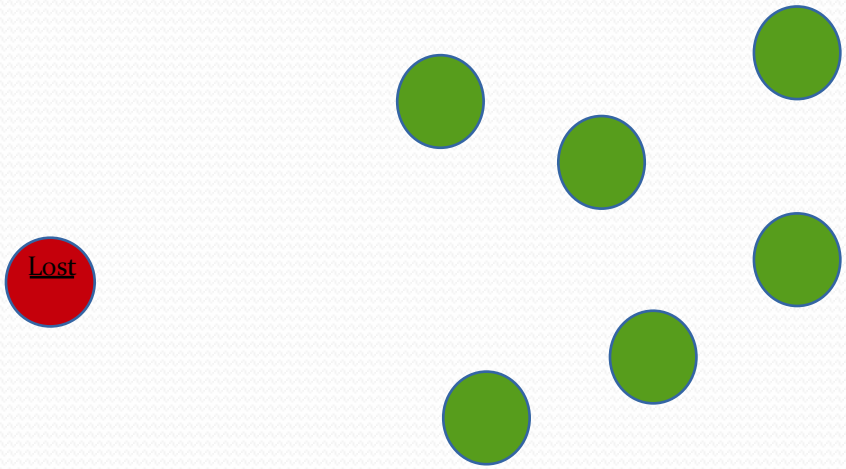


Local Positioning System (LPS)

1)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
- If an agent is missing the localization process for 3 times, then it enters 'Return to Home mode' in which it is directed to the formation shape center



Local Positioning System (LPS)

2)Route Table Determination

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

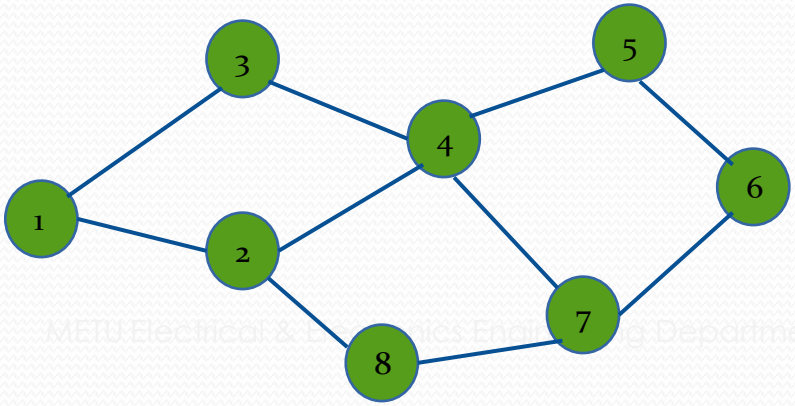
- With the help route tables, clusters around position beacons and the order of the localization is determined.

Local Positioning System (LPS)

2)Route Table Determination

Route table for agent 2

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

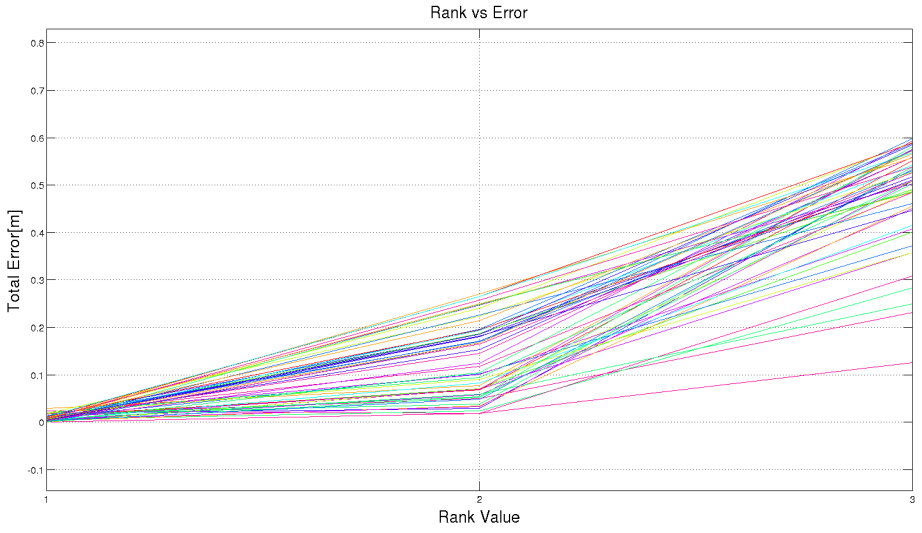


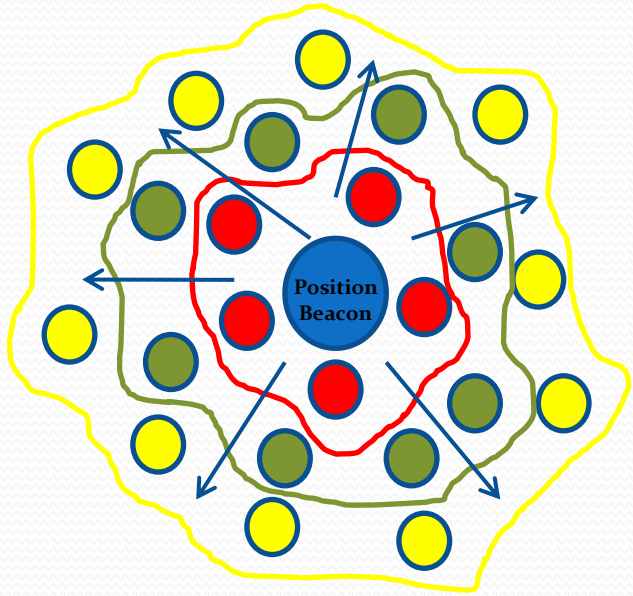
Local Positioning System (LPS)

2)Route Table Determination

Rules for Localization Problem

- Agents decide to join the cluster of position agent with minimum rank value.
- This approach minimizes the total error for the positions of agents which are relatively far away from the beacons.
- Localization process must be done with increasing order of ranks in a cluster.





Formation Control System

Three different approaches were 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

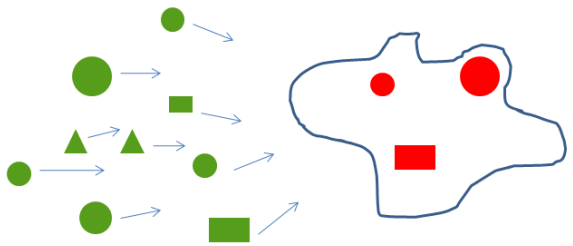
- Partitions the desired formation shape into goal states.
- Control laws are implemented to reach these goal states.

Formation Control System

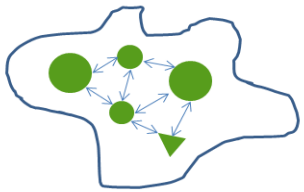
- Artificial Forces Method

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

Attractive Forces

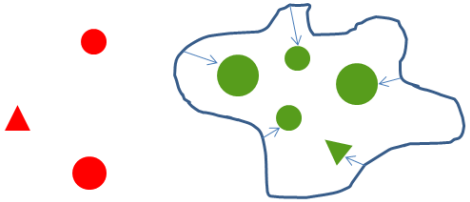


Intermember Forces

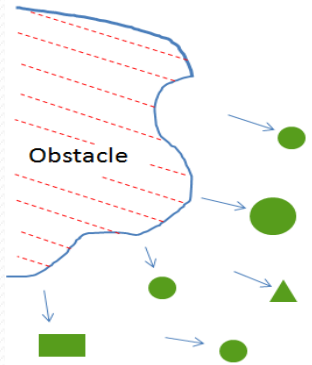


$$u_i = u_{att\_i} + u_{rep\_i} + u_{obs\_i} + u_{int\_i}$$

Repulsive Forces



Obstacle Forces

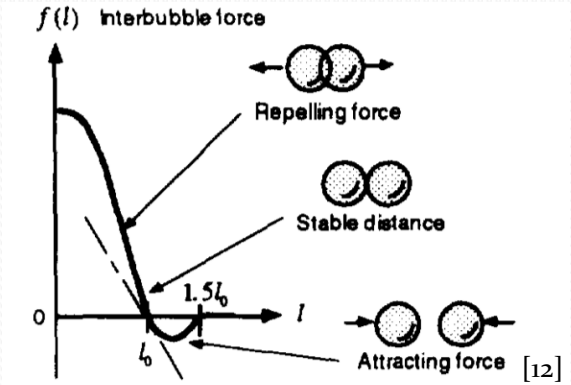


Formation Control System

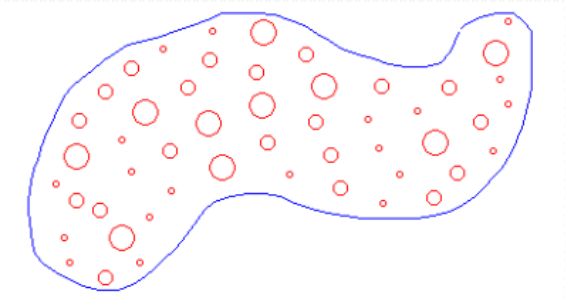
- Bubble Packing and Randomized Fractals Methods

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.

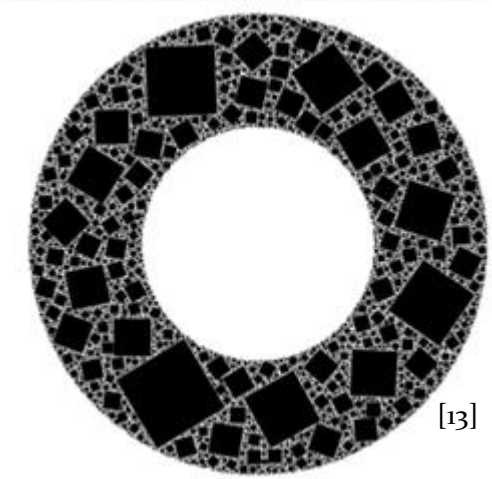
❖ Bubble Packing



[12]



❖ Randomized Factals

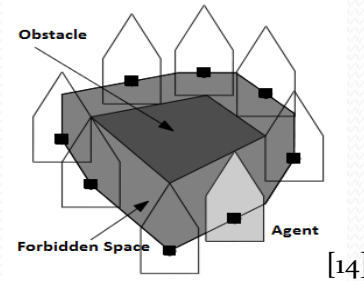

[13]

Formation Control System

- Bubble Packing and Randomized Fractals Methods

Decision of Goal States

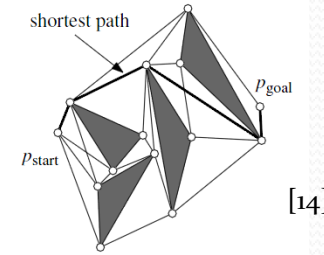
1)Calculation of Free Configuration Space



[14]

Forbidden Space

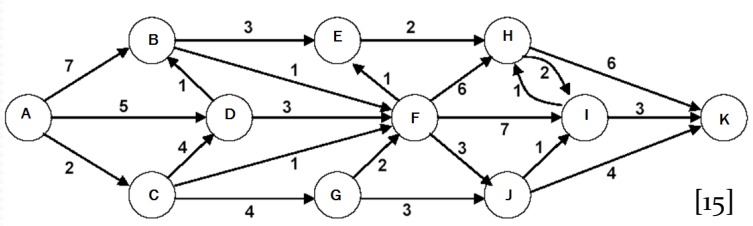
2)Visibility Graphs



[14]

Visibility Graph

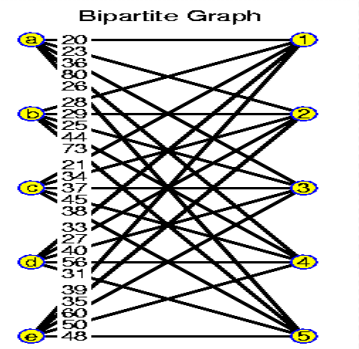
3)Dijkstra's Algorithm



[15]

Calculation of Minimum Shortest Path From A to K

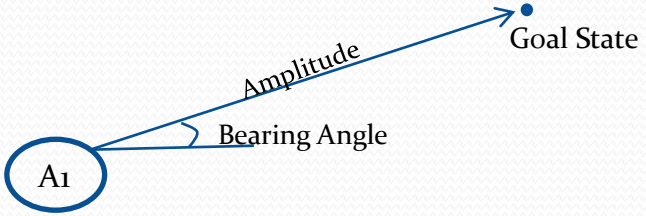
4)Hungarian Algorithm (Munkres Assignment Algorithm)

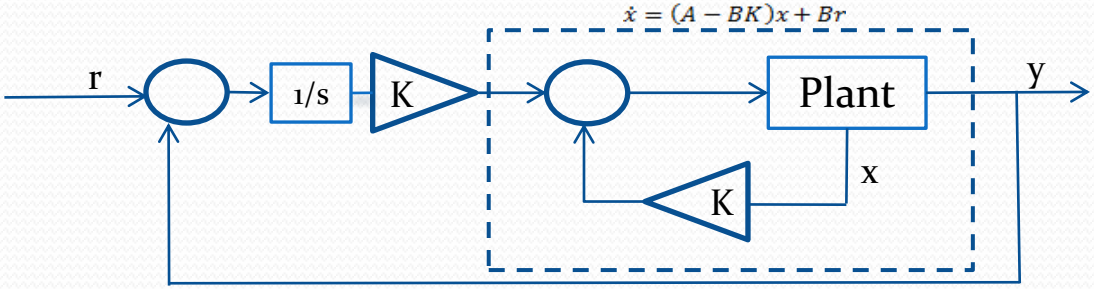


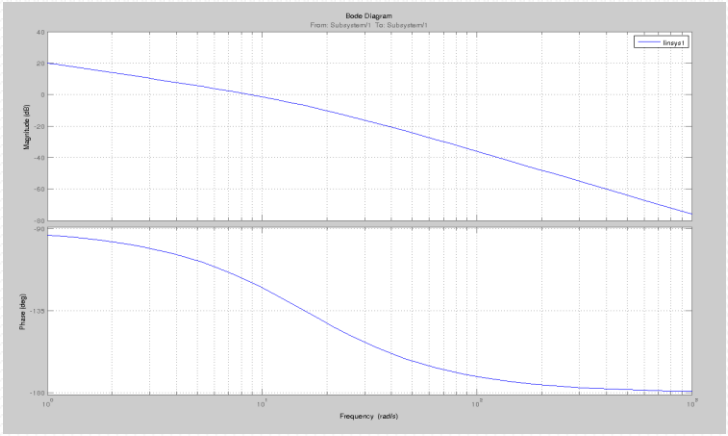
Formation Control System

- Bubble Packing and Randomized Fractals Methods

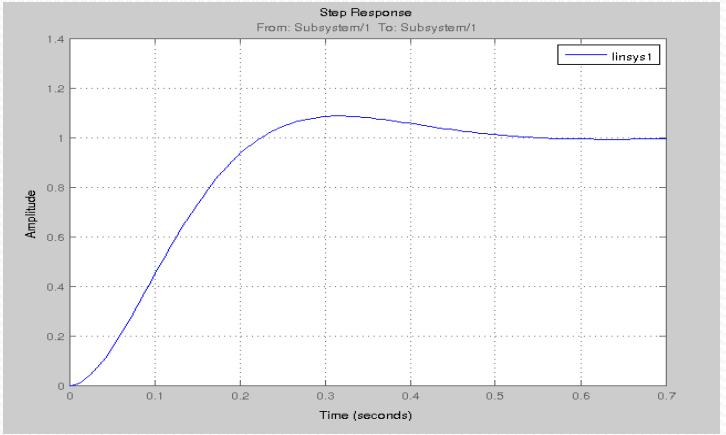
Navigation to Goal States







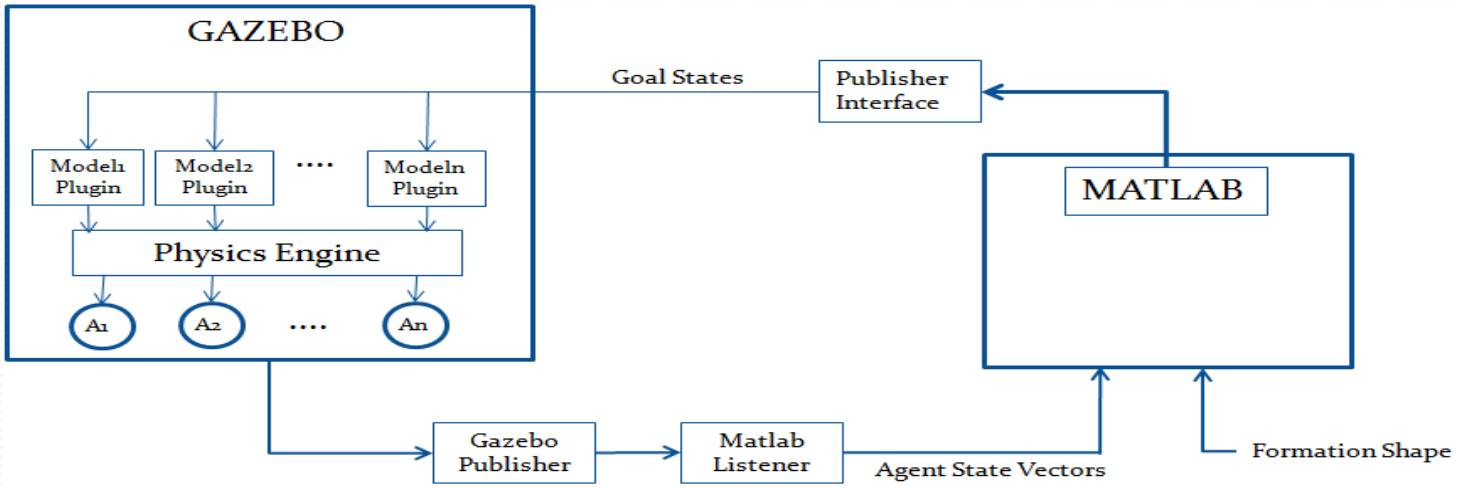
Open loop Bode plots

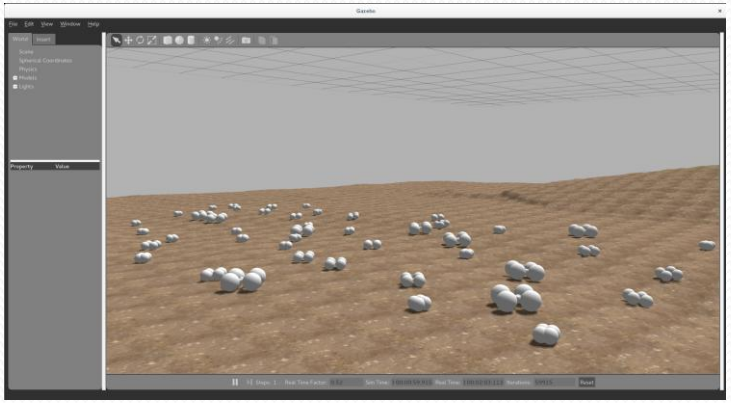


Step response

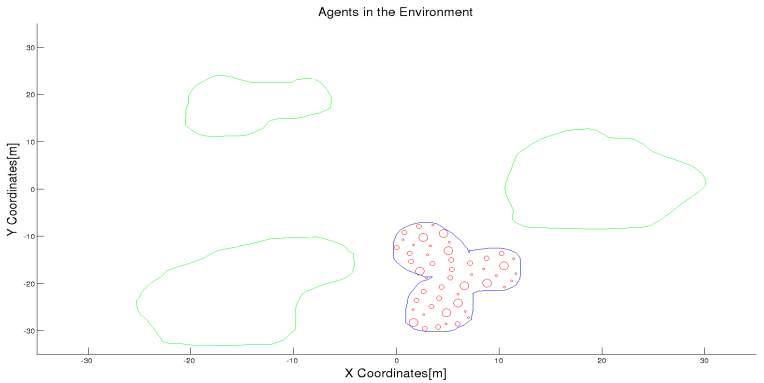
Results

- Local positioning system and formation control system is implemented in a simulation environment.





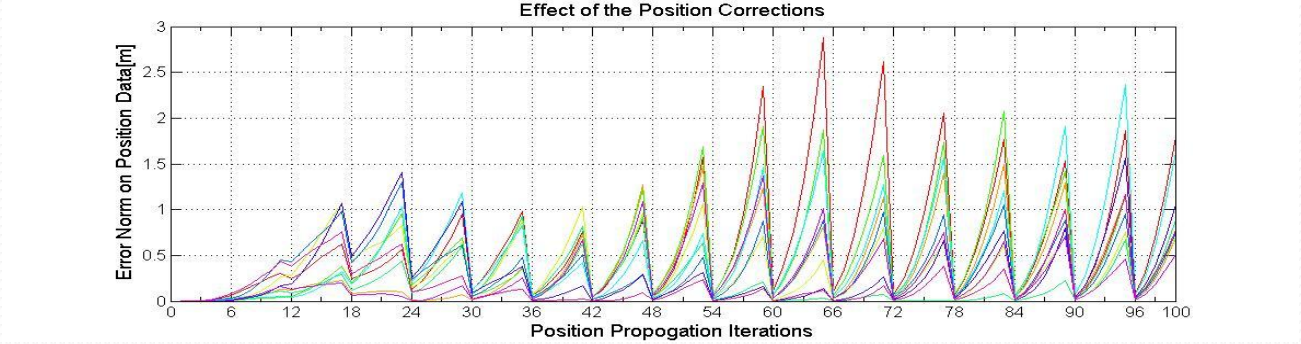
Gazebo Environment

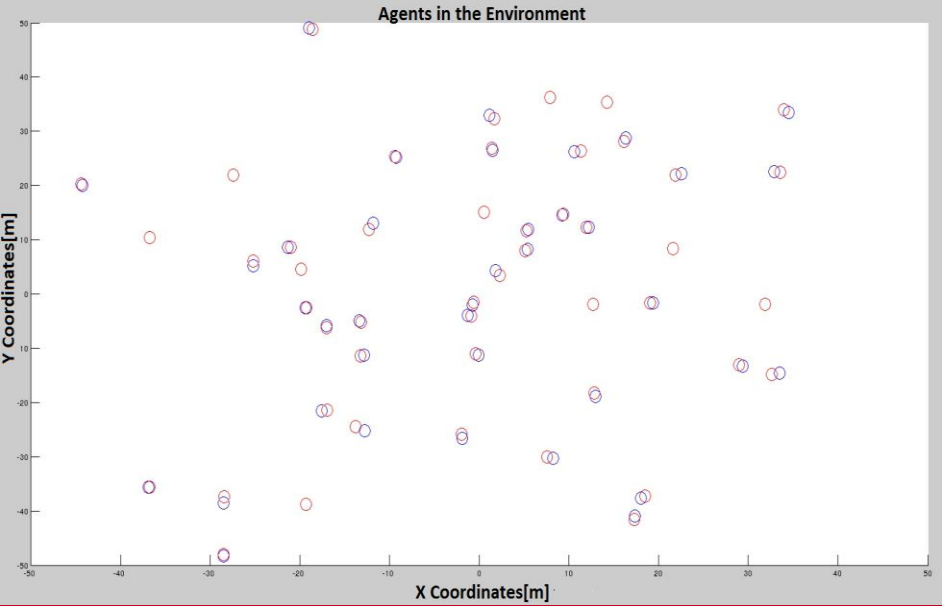


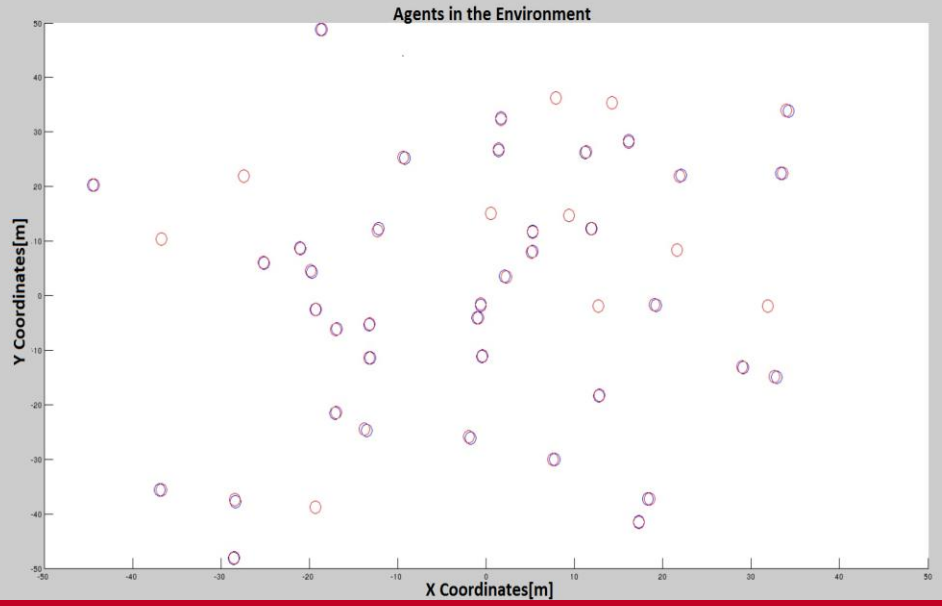
Matlab Environment

Results

Local Positioning System (LPS)

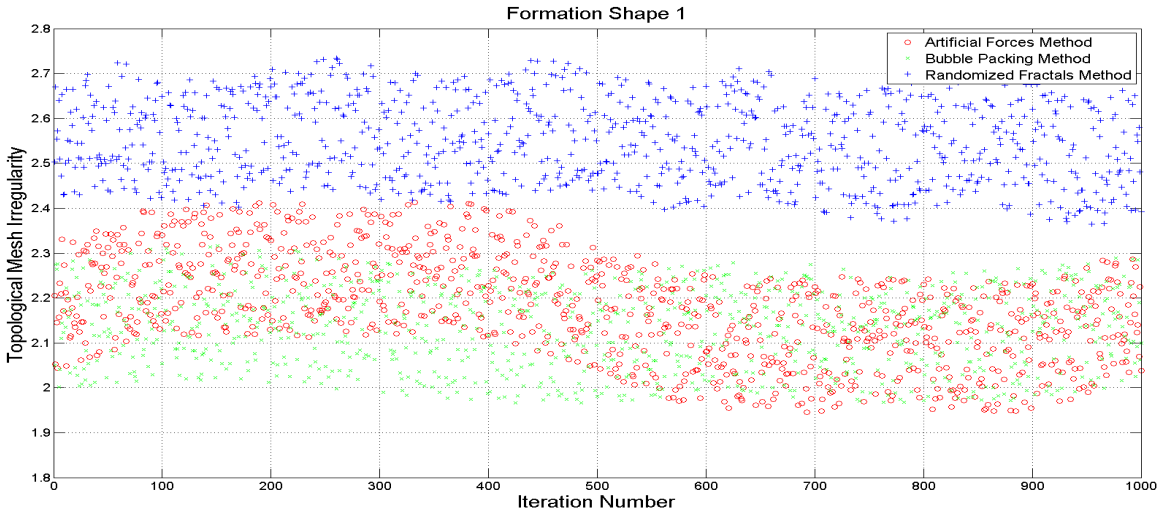


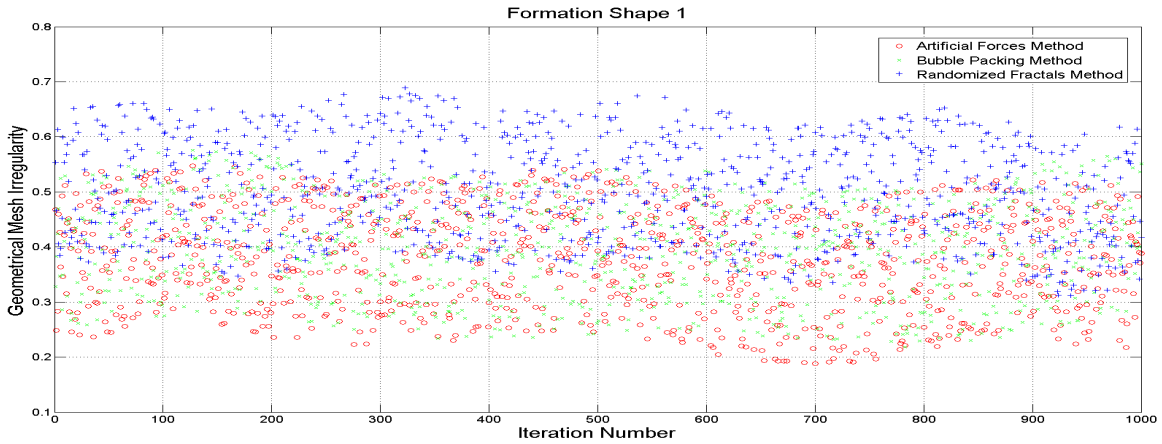




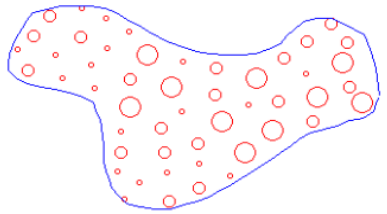
Results

Formation Control System

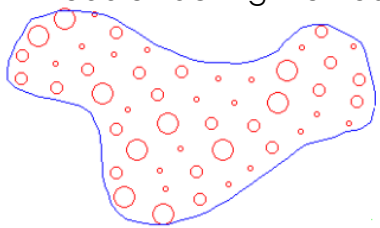




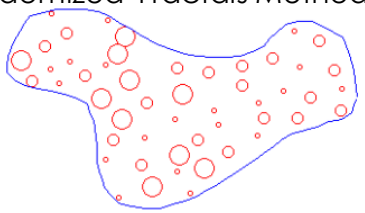
Artificial Forces Method



Bubble Packing Method

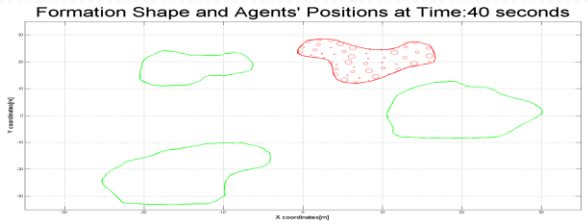


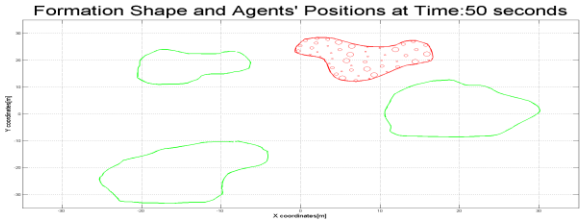
Randomized Fractals Method

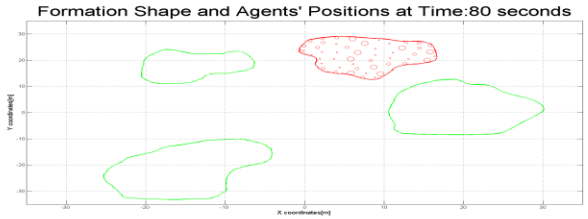


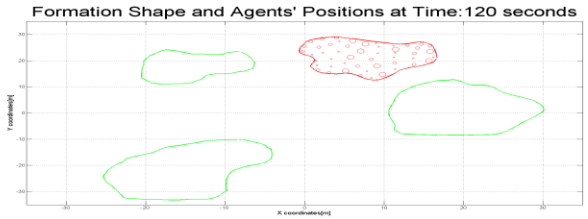
Results

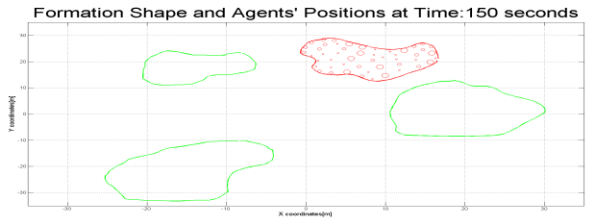
Formation Control System











Results

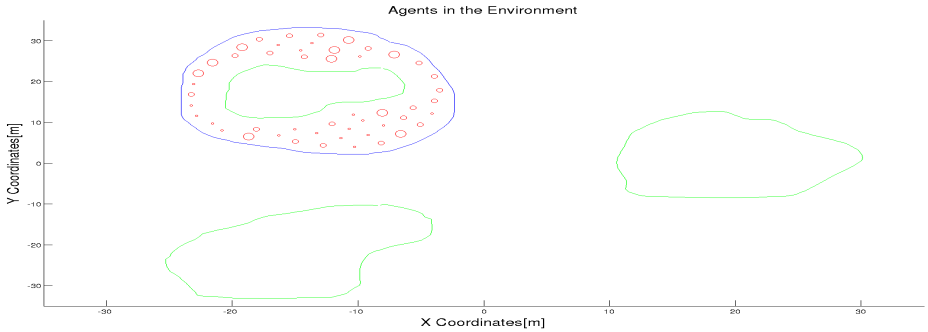
Formation Control System

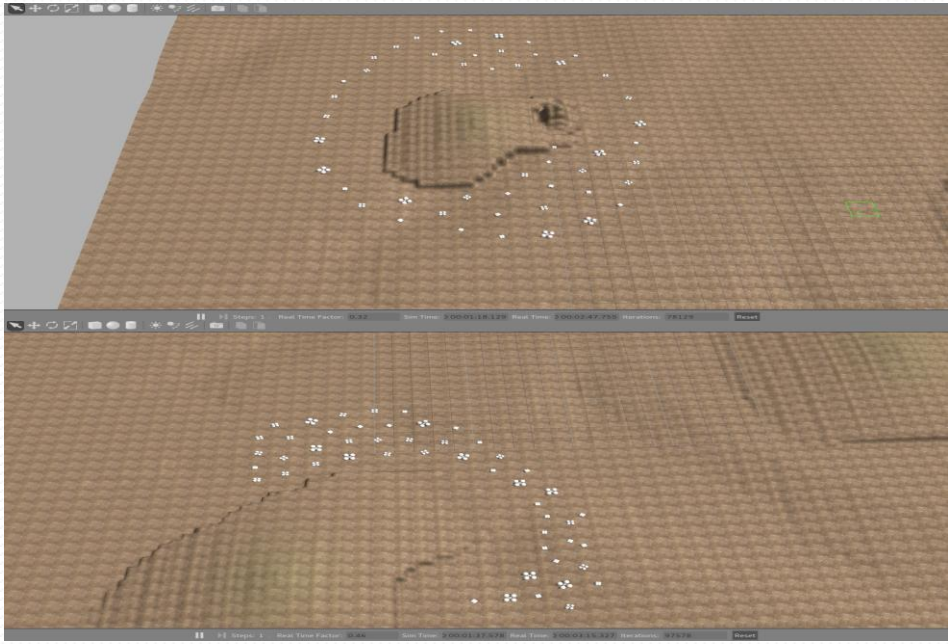
•Comparison of different solution method are illustrated in Table -1

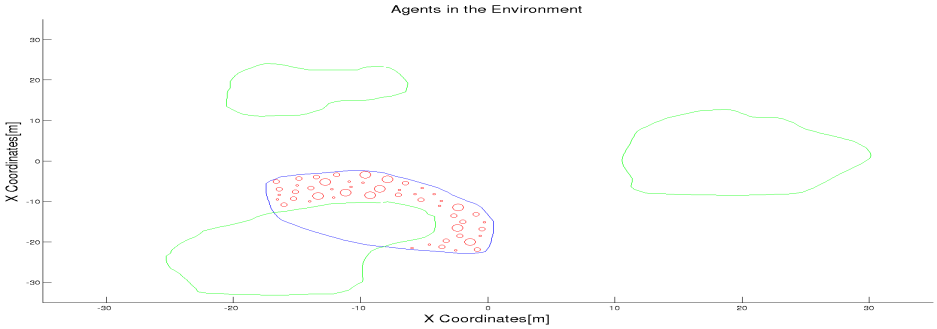
Table -1

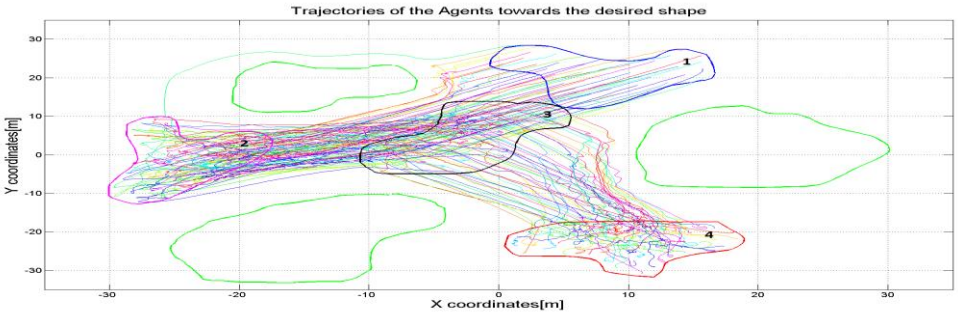
Method/ Metric	Total Displacement	Settling Time	Mesh Quality
Artificial Forces			
Bubble Packing			
Randomized Fractals			

Results


Agents in the Environment
Y Coordinates[m]
X Coordinates[m]

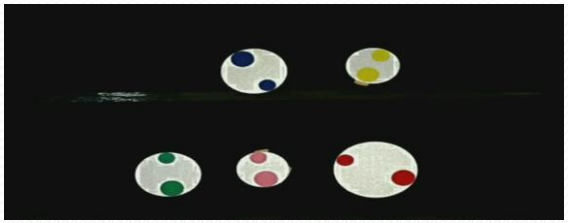


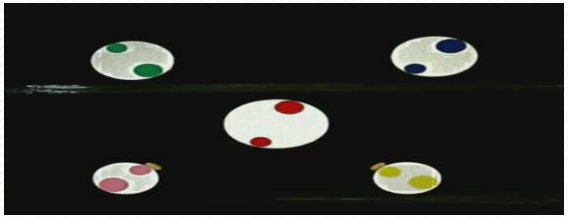

Agents in the Environment
X Coordinates[m]
X Coordinates[m]

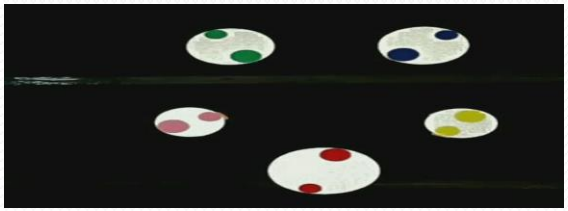

Trajectories of the Agents towards the desired shape
Y coordinates[m]
X coordinates[m]
1
2
3
4

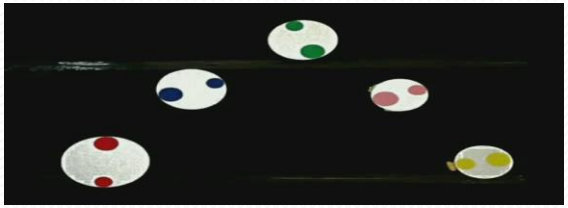
Results

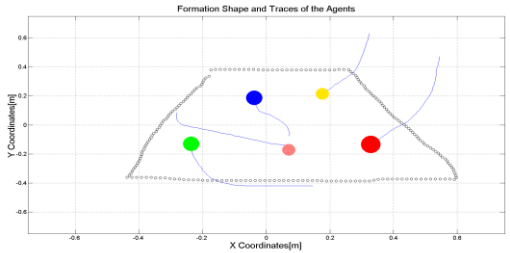
•Hardware applications which implements the methods discussed in this thesis work are also developed

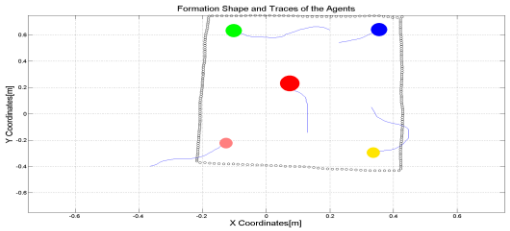


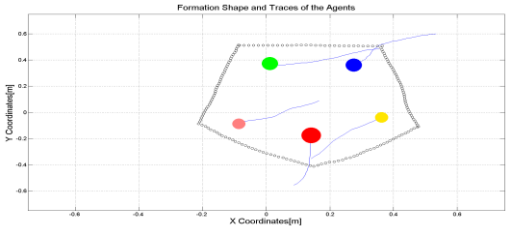


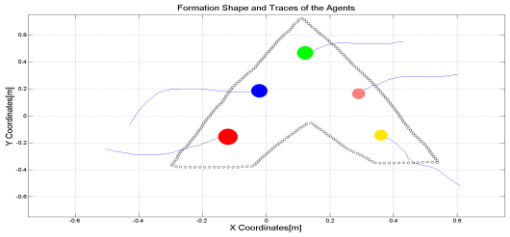


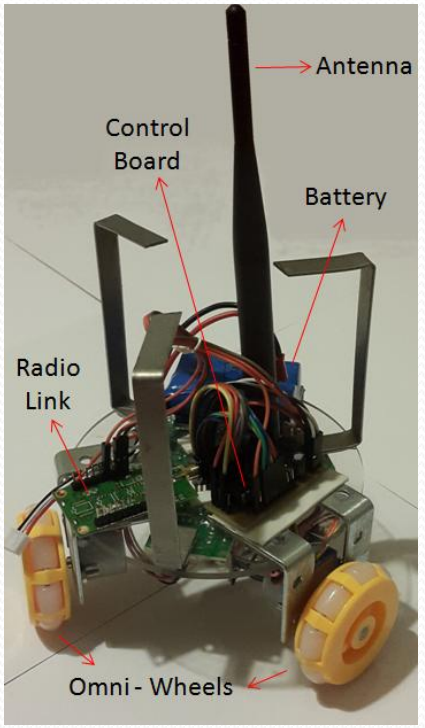












Conclusion

• In literature, generally it is assumed that swarms are composed of homogenous agents, and formation shapes are simple geometrical shapes which do not change dynamically.

• We have proposed a complete solution, including formation control and localization process. We aim to implement a formation control system with heterogenous agents and complex geometrical shapes which are dynamically changing.

• 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

• Bubble packing and randomized fractals method are not fully decentralized, since shape partitioning algorithms are not deterministic. To make the whole process decentralized, an alternative approach for shape partitioning process will be implemented.

•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).

•Heterogenous agents are used to achieve complex tasks with different functionalities of individual agents. We have focused on achieving complex formation shapes with heterogenous mobile robots rather than achieving complex tasks after getting the desired formation shapes. The next step for this project is to add the capability of doing these kind of tasks with the agents from different capabilities.

References

- [1] http://www.eecs.harvard.edu/ssr/projects/progSA/kilobot.html
- [2] Martin N, Klupar P, "Techsat 21 And Revolutionizing Space Missions Using Microsatellites",
- [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.

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

ρ: *tradeoff regulation vs control effort*

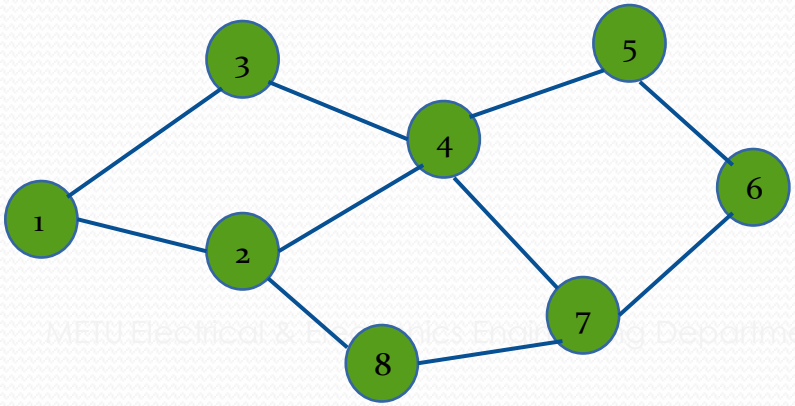
Local Positioning System (LPS)

2)Route Table Determination

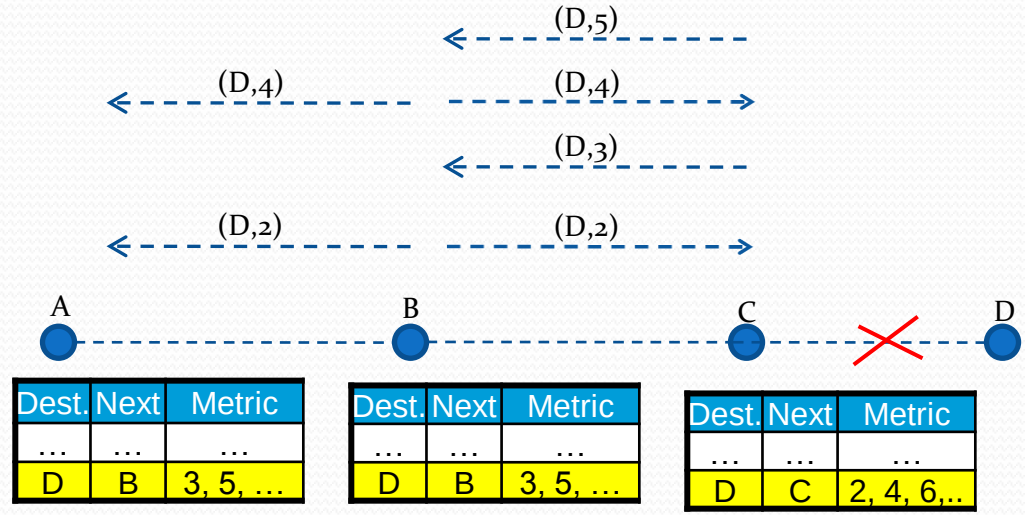
- DSDV is a table driven routing scheme based on Bellman Ford algorithm
- Used to create wireless mesh networks and ad-hoc mobile networks
- Solves routing loop problem in route table algorithms

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



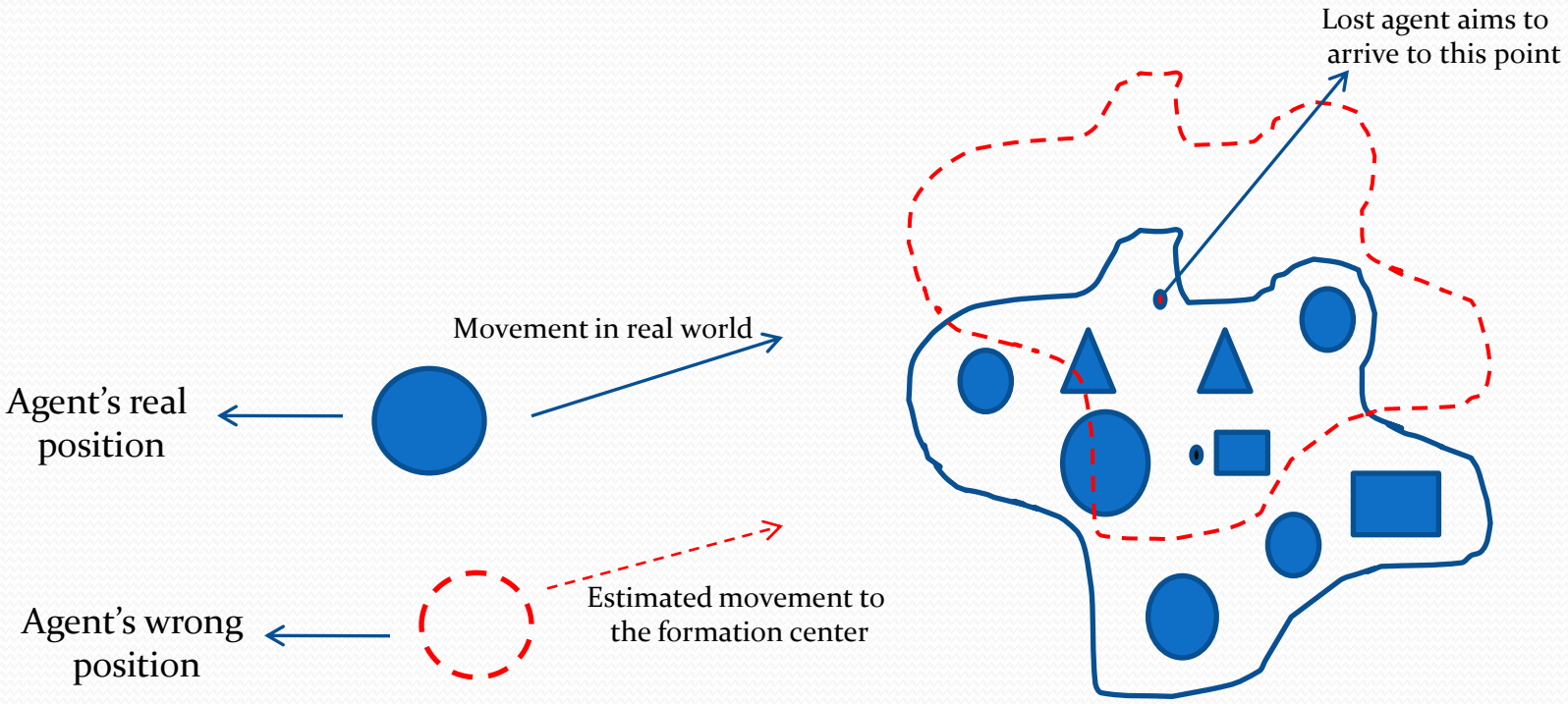
Routing Loop Problem



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

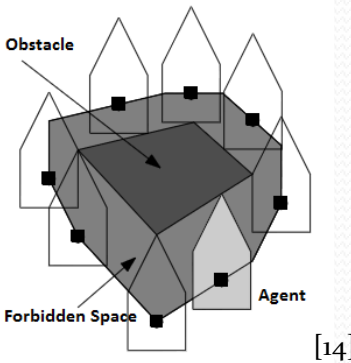
- Bubble Packing and Randomized Fractals Methods

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



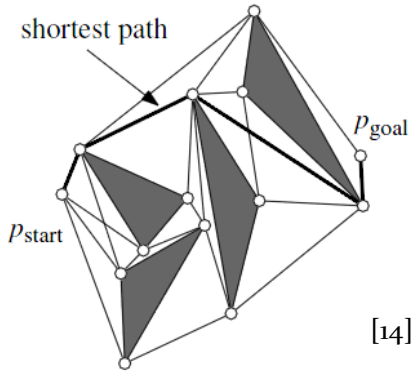
[14]

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

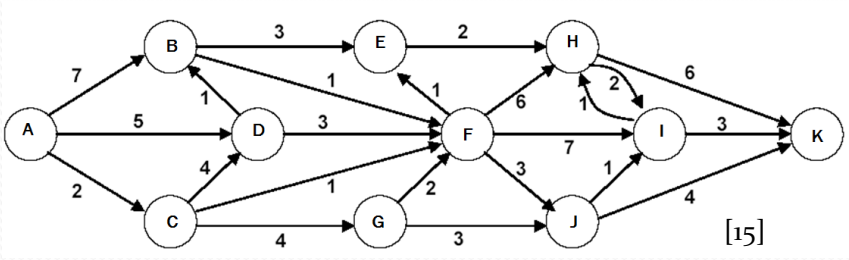
Formation Control System

- 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

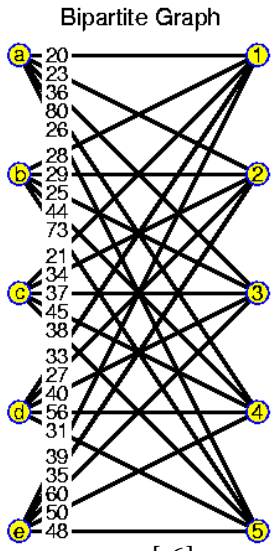


[15]

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