A Modified Null Space Strategy to Avoid Straight Edge Obstacles and Solve Consensus Problem for Multi Robot System

Dissertation *submitted in partial fulfillment of the requirements for the degree of*

Bachelor of Technology

in

Electrical and Electronics Engineering

Submitted by

Mr. Naga Ganesh Kurapati : (Roll No. 10EEE086)

Mr. Aman Sharma : (Roll No. 10EEE075)

Mr. Pranay Bolloju : (Roll No. 10EEE081)

Mr. Tejkaran Charan : (Roll No.10EEE100)

Under the Supervision of

Mr. Matam Manjunath

Department of Electrical and Electronics Engineering

National Institute of Technology, Goa

May 2014

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, GOA

GOA, INDIA - 403401

CERTIFICATE

This is to certify that the work contained in this Dissertation entitled "**A Modified Null Space Strategy to Avoid Straight Edge Obstacles and Solve Consensus Problem for Multi Robot System"** submitted by the group members **Mr. Naga Ganesh Kurapati** (Roll No: **10EEE086**), **Mr. Aman Sharma** (Roll. No: **10EEE0075**), **Mr. Pranay Bolloju** (Roll No: **10EEE081**) and **Mr. Tejkaran Charan** (Roll No: **10EEE100**) to the Department of Electrical and Electronics Engineering, **National Institute of Technology Goa**, for the partial fulfillment of the requirements for the degree of **Bachelor of Technology** in **Electrical and Electronics Engineering**. They have carried out their research work under my supervision. This work has not been submitted else- where for the award of any other degree or diploma.

The Dissertation, in my opinion, has reached the standard fulfilling of the requirements for the degree of Bachelor of Technology in Electrical and Electronics Engineering in accordance with the regulations of the Institute.

Mr. Matam Manjunath,

Assistant professor,

Dept. of Electrical and Electronics Engineering

National Institute of Technology - Goa

E-mail: manjunath@nitgoa.ac.in

ACKNOWLEDGEMENTS

The presented dissertation's existence, in the form it is today, is owed to the encouragement, immense support and feedback by number of people during past two semesters. We would like to take an opportunity, by means of this section, to thank them for their valuable support. Firstly, we would like to thank our advisor **Mr. Matam Manjunath**, Asst. Professor, for being a constant source of motivation. We are extremely thankful to him for being an epitome of optimism and encouragement for us, apart from being *professionally humorous*, against all thicks and thins during the project. We also thank him for always being easily approachable during difficulties.

We grateful to **Dr. Ravi Prasad K.J.**, Asst. Professor, National Institute of Technology Goa, for helping us in modeling the trajectory of encounter point. Without his guidance and feedback, it would have been very difficult for us to find a right approach to prove the proposition presented in this dissertation. Apart from providing technical inputs, we also thank him for being very friendly and cheerful during all discussions.

We thank **Professor Patrice Wira**, Université de Haute Alsace, France and **Dr. A.V. Narasimhadhan**, National Institute of Technology, Karnataka for providing remarks and feedback about the work presented in this dissertation. We extend our gratitude to them for finding time, from their schedule, to review our work and suggest important revisions. We are indeed thankful to Prof. Patrice Wira for suggesting a title, which matches with presented work by good margin, to this dissertation.

We thank our Director – Dr. GRC Reddy, Head of the Department – Dr. B. Venugopal Reddy, faculty, staff and institute administration for rendering help and support. We are gracious to our friends, who have been a gentle source of moral support and of course entertainment during odd times. Last but not the least, we would like to thank our parents for inspiration in life. We are also indebted to the almighty for his blessings.

ABSTRACT

Consensus problem is a famous swarm robotics problem, where mobile robots, initialized at known position coordinates, converge at a point, also known as Rendezvous or consensus point, as the system drives toward steady state. Velocity is, in general, chosen as the control signal for driving each individual robot of the Multi Robot System. Presence of obstacles in the environment makes the agreement of all robots at the consensus point challenging and thus, there is a need of generating appropriate control signals, which ensure both the objectives, i.e. reaching the goal (Rendezvous point) and avoiding all intermittent obstacles. Also, the control strategy should restrain a robot to collide with other robots in it's neighborhood. Null space behavioral control is a very powerful tool, which can be exploited to accomplish all the aforementioned objectives simultaneously, but, in case of straight edge obstacles, for example family of rectangles, it will be mathematically proved that combining the behaviors, which ensure reaching the goal and avoiding intermittent obstacles, in the framework of null space based control, the robot fails to find itself a way to the goal and exhibits type-1 Zeno phenomenon. In this dissertation, a technique is proposed, by incorporating another behavior, which directs a robot to follow the boundary of the obstacle, in Null space based approach to accomplish navigation objectives and solve the consensus problem. Finally, the proposed mathematical formulations will be validated by means of computer simulations and relevant conclusion and inference will be made.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

MRS	Multi Robot System.
NSB	Null Space Behavior.
PCC	Point of Complete Confliction.
GTG	Go To Goal.
OA	Obstacle Avoidance.
FW	Follow Wall.
LTI	Linear Time Invariant.
BIBO	Bounded Input Bounded Output.

NOTATION

$\mathbb{R}^n$	n-dimensional Euclidean Space.
$\|\| \cdot \|\|$	Euclidean norm.
$\mathbf{x}_i$	Position of i^{th} robot in $\mathbb{R}^2$.
$\mathbf{x}$	State vector for a multi robot system.
$\dot{\mathbf{x}}$	Time derivative of $\mathbf{x}$.
$\mathbf{x}_0$	Initial state of multi robot system.
$\mathbf{x}_f$	Final state of multi robot system.
$\mathbf{x}_g$	Coordinates of goal.
$\mathbf{x}_{ob}$	Coordinates of sensed obstacle.
N	Number of robots.
$\mathbf{u}_i$	Control Signal for i^{th} robot.
$\mathbf{u}_{GTG_i}$	Control signal corresponding to goal oriented behavior.
k_{GTG}	Proportional gain corresponding to goal oriented behavior.
$\mathbf{u}_{OA_i}$	Control signal corresponding to obstacle avoidance behavior.
k_{OA}	Proportional gain corresponding to obstacle avoidance behavior.
$\mathbf{u}_{FW_i}$	Control signal corresponding to follow wall behavior.
$\mathbf{u}^{C}_{FW_i}$	Control signal corresponding to clockwise follow wall behavior.
$\mathbf{u}^{CC}_{FW_i}$	Control signal corresponding to counter-clockwise follow wall behavior.
k_{FW}	Proportional gain corresponding to follow wall behavior.
$\mathcal{R}(\theta)$	Rotation matrix.
σ_i	Task function corresponding to a behavior for i^{th} robot.
σ_{d_i}	Reference value of the task function corresponding to the behavior of i^{th} robot.
q	Priority index assigned to task.

$\mathbf{I}$	Identity Matrix.
$\mathbf{J}_i$	Configuration-dependent task Jacobian Matrix for i^{th} robot.
$\mathbf{J}_i^{\dagger}$	Pseudo inverse of $\mathbf{J}_i$.
$\sigma_i^{(1)}$	Task function corresponding to obstacle avoidance behavior for i^{th} robot.
$\sigma_i^{(2)}$	Task function corresponding to goal oriented behavior for i^{th} robot.
$\mathbf{J}_i^{(1)}$	Jacobian of task function associated with obstacle avoidance behavior.
$\mathbf{J}_i^{\dagger^{(1)}}$	Pseudo inverse of $\mathbf{J}_i^{(1)}$.
$\mathbf{N}_{1_i}$	Null space projector matrix of $\mathbf{J}_{1_i}$.
$\mathbf{K}_{GTG}$	Proportional gain matrix for the task corresponding to reaching the goal.
$\mathbf{J}_i^{(2)}$	Jacobian of task function associated with goal oriented behavior for i^{th} robot.
d	Safe distance from the obstacle.
λ	Eigen value.
$\mathbf{\Lambda}$	Jordan Matrix.
$\mathbf{S}$	Modal Matrix.

CHAPTER 1

MOTIVATION

A group of co-ordinated robots can perform the tasks efficiently compared to a single robot. This led to the attention of creating groups of multiple robots to collaborate and accomplish a predefined task in an effective way. Research on multiple robot models started in the beginning of early 1980's. The basic principle for this research is inspired by the observation of nature and different ecosystems. Several examples of collaborative work is observed in nature: group of animals or birds collectively working together to accomplish their goal. Typical example can be found in air, sea, on the ground. School of fishes travel in groups to protect themselves from enemies as shown in fig. 1.1; swarm of bees move collectively, without colliding with each other in search of nectar. A typical swarm of bees, resting in it's hive is portrayed in fig. 1.2. As the animal behavior has evolved in collaborative direction to increase the survival probability of species, the research in the domain of multi robot systems (MRS), to accomplish a predefined task, has also increased. In the nature, among all living beings, humans represent high collaborative level. By the observations, researchers have identified two main collaborative paradigms- Intentionally cooperative and Collectively cooperative.

The idea behind first paradigm is that all robots in the group have knowledge about the presence of other members and are able to coordinate with each other, exploiting global information. The systems based on this approach usually present different kind of vehicles with different abilities [1][2].

The idea behind second paradigm is that a large group of agents with same characteristics, called *homogeneous swarm*, where each robot performs its own task with little information about state of other members. The difference between above two approaches is that in Collective cooperative system where no member in team is important and every one can be sacrificed while it is not in case of Intentionally Cooperative system.

Two different architectures can be used for the control of MRS.

Figure 1.1: School of fish.
Image Courtesy: photography.nationalgeographic.com.

Figure 1.2: Swarm of bees.
Image Courtesy: latimesblogs.latimes.com.

Centralized: - In a centralized control architecture, a robot acts as global supervisor which receives information from other members and the behaviors are coordinated from single point as reported in [3][4]. The limitation with the systems, controlled using this architecture, is vulnerability to failure due to an undesired fault, which may occur to supervising robot.

Decentralized: - In this control architecture, each robot acts on knowledge of local teammates' state and environment and few models are developed based on this approach [5].

The first experiment on swarm robotics system- the Nerd Herd experiment[6]-deals with a decentralized control architecture where 20 robots use very little communication rather than stigmergy (communication through environment modification), in order to perform foraging, dispersion, surrounding and herding. In recent experiments where 100 robots have pointed out the possibility of merging different simple tasks in order to achieve complex goals[7].

Both Homogeneous and heterogeneous robot groups focus on the possibility to obtain a large spectrum of emergent behaviors such as multi-agent manipulation, traffic control, path planning, foraging, flocking and formation keeping. In fact flocking and foraging can be modeled from animals like birds and ants. In foraging domain, typical real applications of foraging problem is de-mining, search and rescue. The main issue of this application is how to coordinate robots for fast exploration without interfering with each other, that is foraging problem is related to coverage problem and their solutions are reported in[8][9]. Flocking and formation keeping can be considered like different realizations of same problem, that is to find a way to coordinate a group of mobile robots in order to make them move together while preserving group compactness. One of the solutions to coordination of mobile agents flocking together can be found in[10].

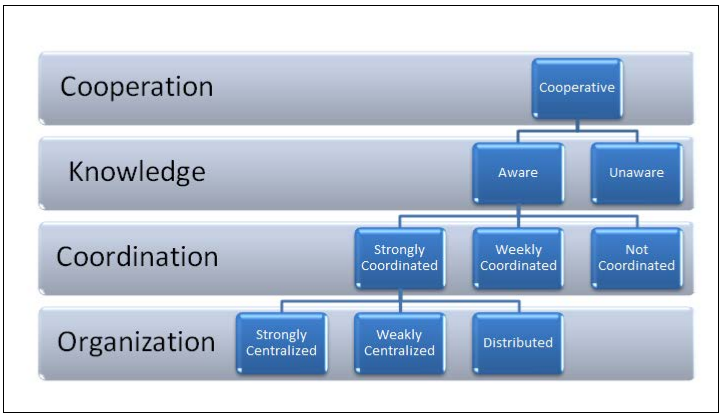


Figure 1.3: Taxonomy of Multi Robot Systems.

1.1 Taxonomy of Multi-Robot System

According to [11], six classification dimensions are individuated: Cooperation, Knowledge, Coordination, Organization, Communication and System Composition.

1.1.1 Cooperation Level

Cooperation level basically underlines the interaction robots have among themselves while executing some task . Some of the definitions which are mostly accepted of cooperation are: a joint collaborative behavior, directed toward goal; in which there is a common interest or reward [12]; communication based interaction [13]; joint effort for doing a task that creates a progressive result, for example increasing performance or saving time [14].

1.1.2 Knowledge Level

The first important characterization can be based on how much knowledge each robot has about the presence of other robots in its group. An aware robot [15] can be defined as the robot which has knowledge of all the neighboring robots, whereas an unaware robot considers itself to be the only robot in the environment. Cooperation amongst unaware robot is the weakest form of cooperation.

1.1.3 Coordination Level

The coordination mechanism can be defined as the way each robot takes into account, the action of other members of the group, to achieve the common global goals. In strongly coordinated systems,each robots exerts its influence on the behavior of the other robots, based on the predefined rules concerning the way the robots have to interact.
Weakly coordinated systems do not rely on a predefined coordination protocol and finally not coordinated systems do not rely on any form of coordination.

1.1.4 Organizational Level

It can be defined as the level of centralization of the cooperation mechanism which play a central role. A centralized system is more likely to have a hierarchical structure, wherein robots operating under the supervision of a leader, can be leaders of sub-groups constituting the MRS. In a strongly centralized systems, the leader remains the same during the mission but in certain situations, leader is dynamically selected based on the changes in environment.

1.1.5 Communication Level

Strongly coordinated systems need to communicate in order to execute the coordination protocol, while weakly coordinated systems and not coordinated systems can be developed with or without communication while unaware systems do not use communication at all.

1.1.6 Composition Level

The last level can be considered to the group composition. In particular, groups can be heterogeneous or homogeneous. Heterogeneous refers to robots having differences either in hardware devices or software procedures as found in [1][2]. Homogeneous refers to robots which are exactly same both in hardware devices and software procedures. In an homogeneous system, every agent can execute the same actions as other team members with same results by which a failure of member can be compensated.

1.2 Applications - MRS

This section gives an overview of some of the applications of MRS. Being specific, applications pertaining to addressing foraging, coverage problems is discussed followed by a brief discussion of prior work in solving exploration and flocking problems and their associated applications.

1.2.1 Foraging and Coverage

This task requires the components of MRS to pick up objects in the environment such as toxic waste cleaning, mine cleaning and service robots [16][17]. A major task in this is how to avoid interference and collisions among robots during the execution. The coverage task is very similar to foraging, since it require robots to explore all the points of the free space in the environment and main issue for coverage is to find effective techniques for cooperatively scanning all the environment. Applications for coverage are: de-mining, snow removal, lawn mowing and car body painting.

1.2.2 Exploration and Flocking

Exploration and flocking indicate different tasks that differ in the way they are realized, but have common feature which require MRS to coordinate their motion in order to improve performance. In case of flocking, formation maintenance or cooperative map building can be considered in same class. In exploration task, robots need to distribute in the environment in order to collect as much information as possible about surrounding area. In flocking, the task and goal for robotic agents is to move together, such as in a flock. The formation task involves moving robots in the environment, forming particular shapes to fuse information acquired from the environment. The problem of exploration and flocking is related to several applications, such as motion coordination in industrial application and exploration of dangerous environments. Another example of exploration task is given by the RoboCup Robot-Rescue league [18], that is a setting for experimenting MRS involved in searching victims in an unknown environment, representing a disaster, a natural calamity scenario.

1.3 Outline of Thesis

- Chapter 2 introduces paradigm of behavior based control and the way, the behaviors are put into action (arbitration mechanism). In this chapter basic behaviors like *Go To Goal*, *Obstacle Avoidance* and *Follow Wall* will be discussed followed by merits and demerits of switching and blending arbitration mechanisms

- Chapter 3 illustrates Null Space based behavioral control for a Multi Robot System and also intuitively points failure of Null space behavior in presence of rectangular obstacle for a single mobile robot.

- Chapter 4 presents a mathematical proof of failure of combining Go to Goal and Obstacle avoidance behaviors in NSB framework for a single mobile robot in presence of rectangular obstacle. Modified Null space based control strategy to accomplish MRS navigation in an environment containing rectangular obstacles is also presented.

- Chapter 5 validates theoretical and mathematical formulations presented in previous chapter by means of computer simulations carried out in MATLAB and JAVA.

- In chapter 6, conclusions based on theoretical and experimental formulations are made. A scope to prove the failure of conventional Null space based control strategy in presence of concave obstacles is also presented.

CHAPTER 2

INTRODUCTION TO BEHAVIORAL CONTROL

This chapter explores control strategies to drive a mobile robot from origin or any initial to a predefined goal position. The world is dynamically changing and fundamentally unknown and hence the controller should respond dynamically to the corresponding changes in environment. Building separate behavior for each task and employing arbitration mechanism may serve the purpose instead of building one complicated controller to accomplish navigation of mobile robots. Section 2.1 discusses the behaviors, which will be put into use in upcoming chapters : Go-to-goal (GTG), Obstacle Avoidance (OA) and Follow-wall (FW). Section 2.2 will elucidate arbitration mechanisms.

2.1 Behaviors

Behaviors may be thought of as actions executed by a robot under the purview of control signal. Control signal basically decides the way, robot shall execute a predefined task at some particular instant of time. For example, let us consider objective to reach the goal: in this case, there is a need to generate appropriate control signal, which can drive the robot towards the goal. What may happen if there is an obstacle, which lies in between robot and goal? There is a need of another behavior, which can prevent robot from colliding with obstacles.

2.1.1 Goal Oriented Behavior

Let $\mathbf{x}$ be position vector of the robot and $\mathbf{x}_g$ be the coordinates of goal as shown in fig. 2.1. The objective is to drive a robot towards goal and hence velocity of robot due to GTG behavior is chosen to be proportional to a vector pointing from robot to the goal. The state

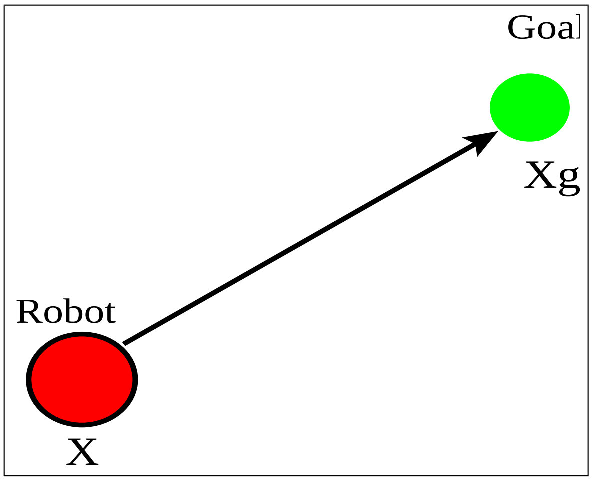


Figure 2.1: Go To Goal.

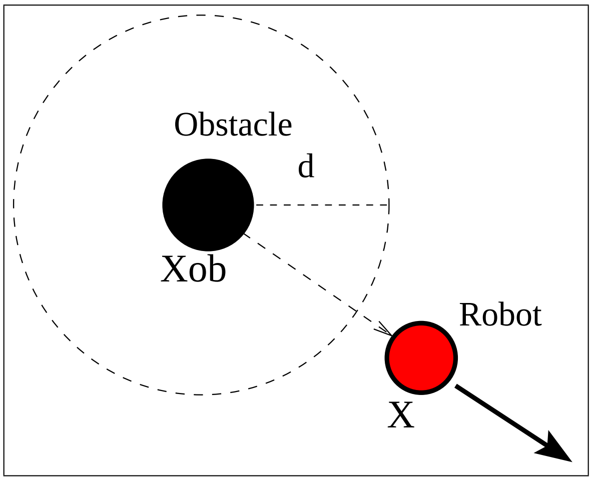


Figure 2.2: Avoid Obstacle.

of the system is governed by the equation:

$$\dot{\mathbf{x}} = \mathbf{u}_{GTG} = k_{GTG}(\mathbf{x}_g - \mathbf{x}), \tag{2.1}$$

where $\mathbf{u}_{GTG}$ denotes control signal due to GTG behavior and $k_{GTG} \in \mathbb{R}^+$.

2.1.2 Obstacle Avoidance Behavior

There exists many directions, a robot can move, to avoid collision with obstacles. The robot can move in pure OA mode or move in direction perpendicular to obstacle. It can also move in other directions which is combined behavior of GTG and OA. In this section

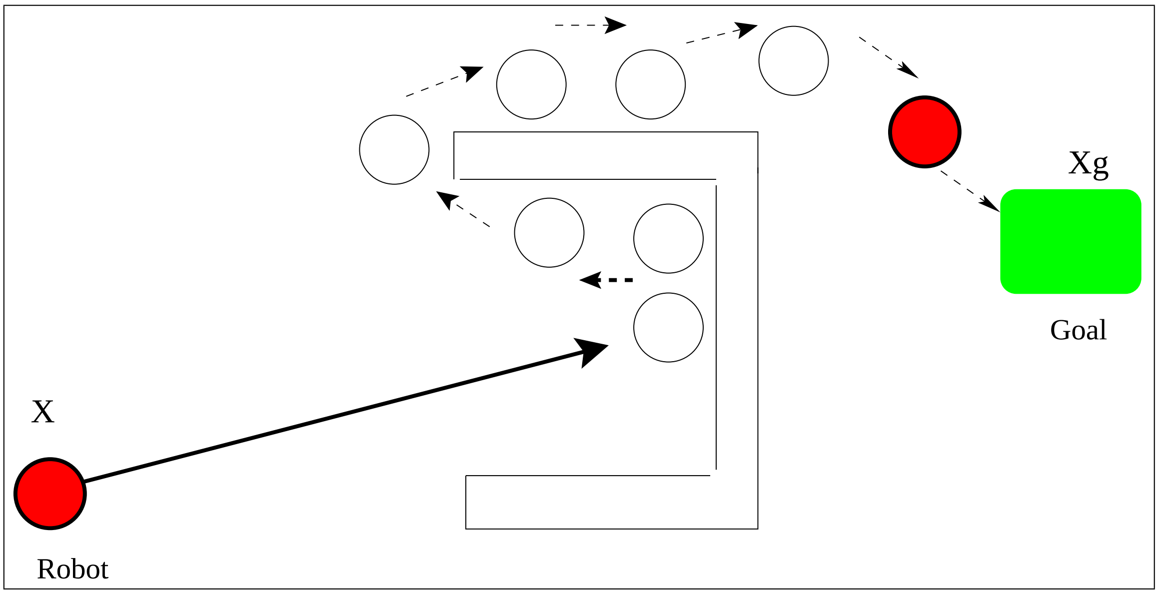


Figure 2.3: Follow wall behavior.

pure OA mode is discussed. Pure OA mode is achieved by making robot move with a velocity proportional to a vector pointing from obstacle (sensed by the robot) towards robot itself (normal to the surface of the obstacle) as shown in fig. 2.2. The state of system is governed by following dynamical relation:

$$\dot{\mathbf{x}} = \mathbf{u}_{OA} = k_{OA}(\mathbf{x} - \mathbf{x}_{ob}), \tag{2.2}$$

where $\mathbf{x}_{ob}$ is the position of obstacle sensed by robot and $k_{OA} \in \mathbb{R}^+$.

2.1.3 Follow Wall Behavior

The behavior which drives a robot such that it follows boundary of obstacle, is known as 'Follow wall' (FW) behavior. The trajectory of robot under the action of FW behavior is shown in fig. 2.3.

The Boundary can be followed either clockwise $(\mathbf{u}^{c}_{FW})$ or counter-clockwise $(\mathbf{u}^{cc}_{FW})$ as shown in fig. 2.4. Concretely, Follow wall behavior is governed by following mathematical equation:

$$\mathbf{u}_{FW} = k_{FW}\ \mathcal{R}(\pm\pi/2)\mathbf{u}_{AO} \tag{2.3}$$

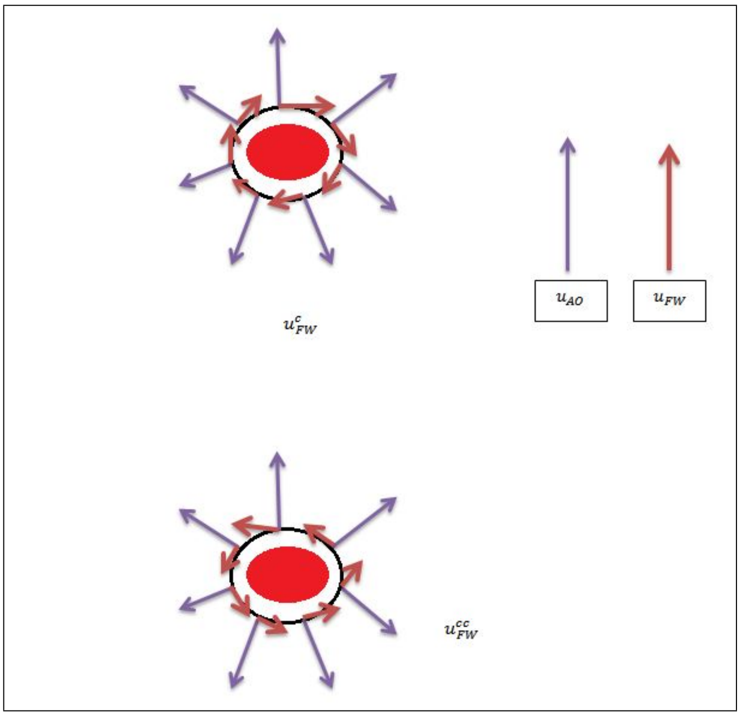


Figure 2.4: Following the boundary of the obstacle.

where k_{FW} is scalar and $\mathcal{R}(\theta)$ is rotation matrix with $\theta = \pm\pi/2$:

$$\mathcal{R}(\theta) = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & \cos\theta \end{bmatrix}$$

This illustration is the basis for the initial understanding of the FW behavior. However it fails to answer fundamental questions like “Whether to Follow wall Clockwise or Counter-Clockwise?" and “When to stop following walls?". These questions are catered in subsequent section.

2.1.4 Exploring ‘Follow Wall’ behavior

This section discusses the decision making involved in choosing, whether to follow the walls clockwise or counter-clockwise with respect to OA behavior. The section also discusses the stopping criterion to terminate following of walls of the obstacle.

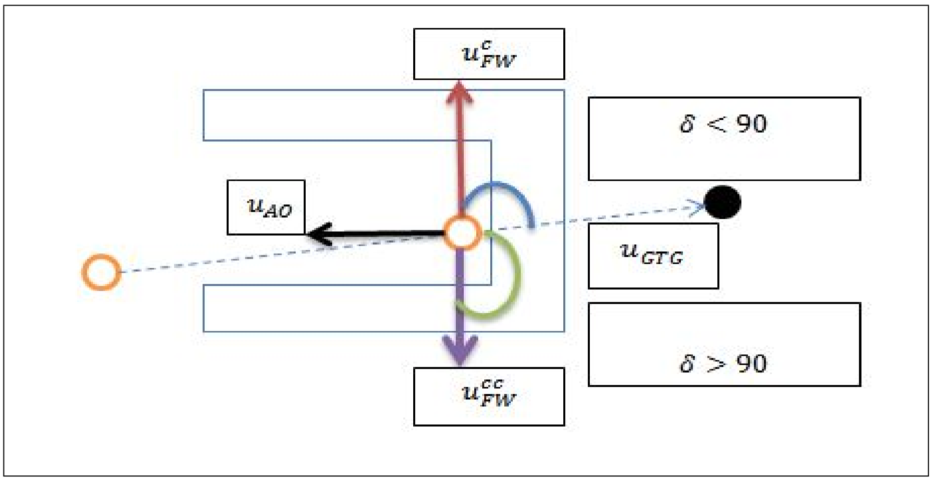


Figure 2.5: Whether to Follow Walls Clockwise or Anti-Clockwise?

2.1.4.1 Whether to follow walls clockwise or counter clockwise?

The answer to this fundamental question is given by inner product between the goal oriented and wall following oriented control signal; robot follows the wall (or surface of obstacle) in a direction where inner product between the goal oriented control signal and corresponding follow wall (clockwise and counter-clockwise) control signal is positive. Concretely,

$$\mathbf{u}_{FW} = \begin{cases} \mathbf{u}_{FW}^{C} & \text{if } \mathbf{u}_{FW}^{C} \cdot \mathbf{u}_{GTG} > 0 \\ \mathbf{u}_{FW}^{CC} & \text{if } \mathbf{u}_{FW}^{CC} \cdot \mathbf{u}_{GTG} > 0 \end{cases} \tag{2.4}$$

Consider a mobile robot as shown in fig. 2.5. δ is the angle between goal and follow wall oriented control signal. In accordance to aforementioned rule, robot should follow wall clockwise.

2.1.4.2 When to stop following walls?

Intuitively one may identify right place to stop following walls, as shown in fig. 2.3 however, computers do not have intuition. A mathematical model to identify this situation and stop following the wall is illustrated with the help of fig. 2.6, which depicts trajectory of a mobile robot reaching the goal through avoiding intermittent obstacles. With reference

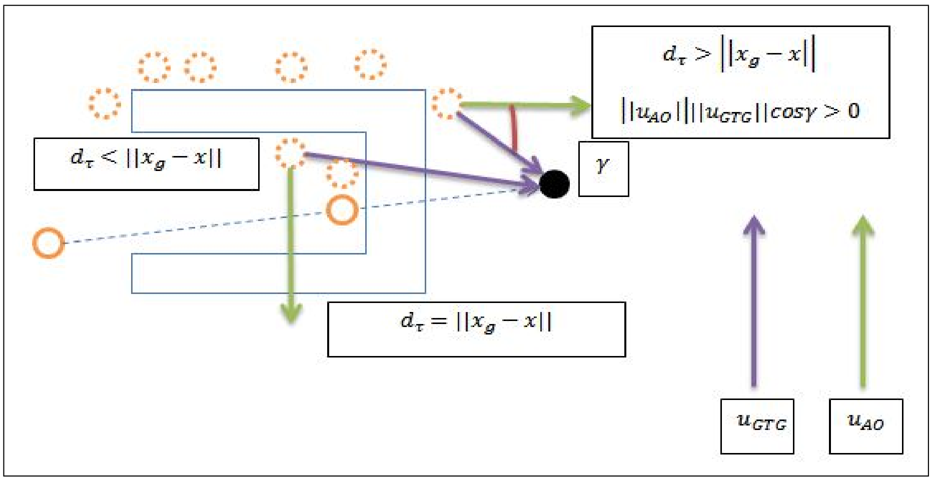


Figure 2.6: When to stop following walls.

to previous paragraph it follows the boundary clockwise with reference to OA control signal. Following two rules have to be simultaneously satisfied to stop following walls:

1. Progress made towards the goal by predefined margin.

2. The inner product between GTG and OA control signals is positive.

Concretely,

$$\begin{cases} ||\mathbf{x_g} - \mathbf{x}|| < d_\tau \\ \mathbf{u}_{GTG} \cdot \mathbf{u}_{OA} > 0 \end{cases} \tag{2.5}$$

Two arbitrary positions of the mobile robot's trajectory is being analyzed in the fig. 2.6. For first position, it is seen that robot has not made enough progress towards the goal. Let $\mathbf{x}_g$ denote coordinates of the goal and $\mathbf{x}$ of robot position and d_τ is distance of the robot from the goal when it starts to follow wall. At the considered instant, the distance of goal from robot is greater than d_τ, hence, as per rule 1, robot does not follow the wall though the inner product between goal and avoid obstacle control signals is positive. At second arbitrary considered instant, one can observe the distance between robot and goal is less than d_τ and it has a clear shot at the goal (rule 2 is satisfied) thus at this position robot stops following walls and switches to GTG behavior.

2.2 Behaviors in action- Arbitration Mechanism

There are two arbitration Mechanisms, namely:

- Blending.
- Hard switching.

Both approaches have merits in different situations. Hard switching results in bumpy ride but performance is guaranteed whereas blending results in smooth ride but reaching the goal may not be possible in some situations.

2.2.1 Blending

Blending is employed by combing GTG and OA behavior as shown in fig. 2.7. Let $\mathbf{u}_{GTG}$ be the velocity vector directing a robot towards the goal and $\mathbf{u}_{OA}$ be the velocity vector directing it away from obstacle. Now, the resultant vector, due to blending, is given by:

$$\dot{\mathbf{x}} = \sigma(d)\mathbf{u}_{GTG} + (1 - \sigma(d))\mathbf{u}_{OA} \tag{2.6}$$

This arbitration mechanism directs a robot to move with a velocity, which is equal to the resultant velocity due to OA and GTG behaviors, where $\sigma(d)$ is blending function, with $\sigma \in$ [0,1].

2.2.1.1 Drawbacks of Blending

While employing blending, there is a component of GTG behavior, which is opposing the OA behavior and there may be a situation that robot collides with the obstacle. If goal is close to the obstacle then it may also happen that robot never reaches the goal, while trying to avoid obstacle.

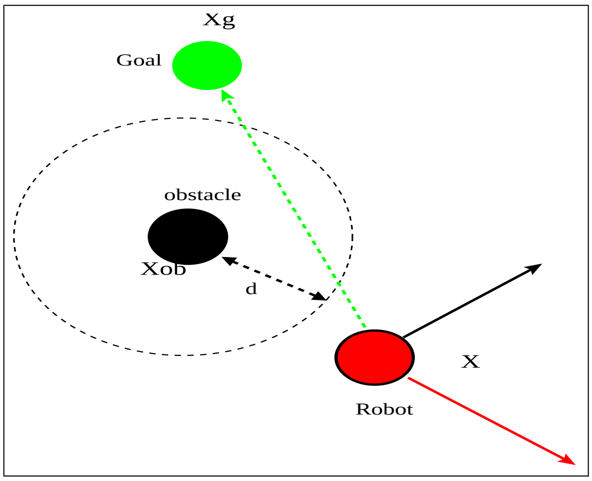


Figure 2.7: Blending

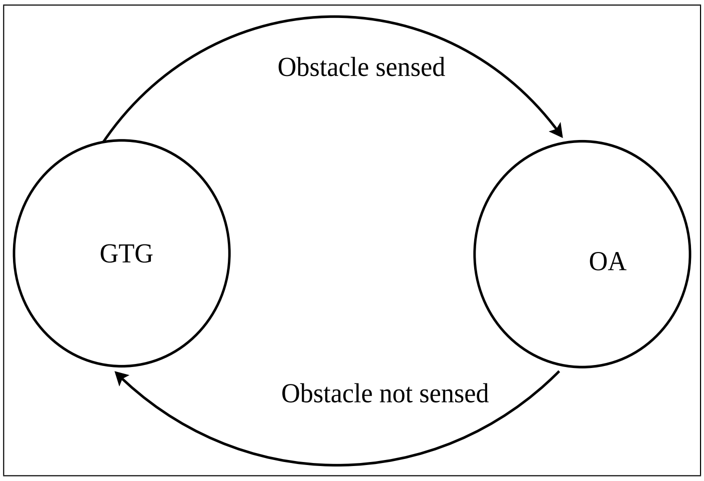


Figure 2.8: Hybrid System depicting switching

2.2.2 Switching

Switching mechanism is a technique where either GTG or OA, but not both at once, is used at a particular instant unlike in the case of blending. Switching leads to developing a hybrid system as shown in fig.2.8 .

2.2.2.1 Drawback of Switching

Consider a convex obstacle as shown in fig.2.9. The robot tries to reach the goal while avoiding collision with the intermittent obstacles. Since the robot is at a safe distance from the obstacle, the goal oriented controller will drive the robot towards the goal till

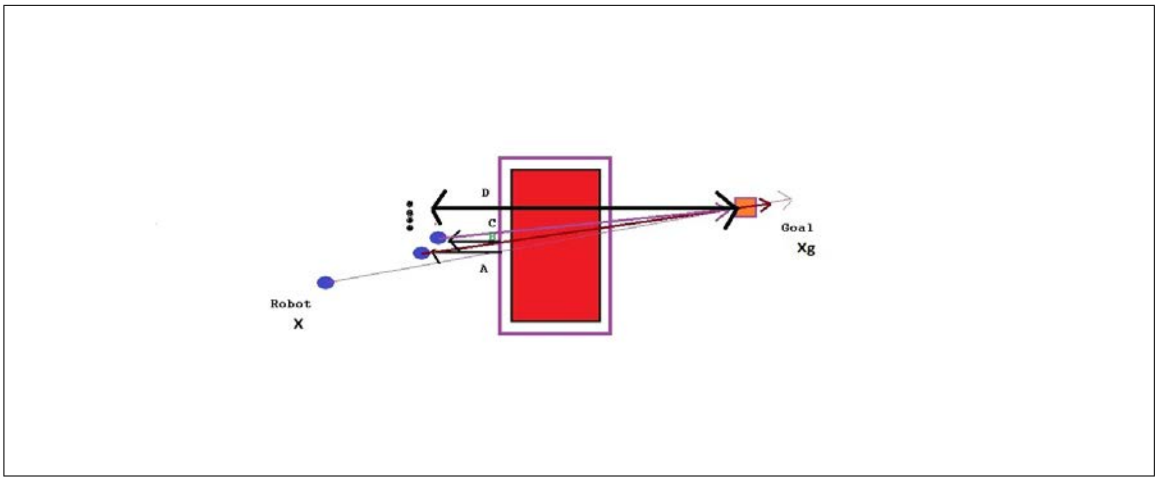


Figure 2.9: Hard switching in convex world.

point A and then the OA behavior comes into picture which will direct the robot strictly away from the obstacle as shown in the figure. The 'GTG' again comes into play and drive the robot to point B and the sequence continues till D, where the Goal oriented and OA control signals are antagonistic to each other and this results in perpetual oscillation of the robot along the line passing through D.

CHAPTER 3

NULL SPACE BASED BEHAVIORAL CONTROL

Chapter 2, discussed the main aspects of behavioral control viz, Goal Oriented, Obstacle Avoidance and Boundary Follow Behaviors. Previously explained methods have some limit, in handling the combinatorial possibilities of priority and sub priority tasks, some fail when number of behaviors are large and complex in nature because their formalization becomes very tedious, while some cause troubles when dealing with conflicting tasks and some leads to underused system but a predictable output. It concludes and motivates for a need of a behavior which is better than the previous explained methods.

Therefore there is a genuine need for a better approach, which is fulfilled by Null Space based behavioral approach. Null Space Behavior (NSB) is a competitive-cooperative approach, it accomplishes global task by prioritizing the local conflicting subtasks. In NSB framework, lower priority task is projected on to the null space of next conflicting higher priority task till the highest priority task is achieved.

In this chapter, section 3.1 highlights the mathematical framework model required for NSB, its systematic approach and explanations. Section 3.2 and 3.3 discusses about prioritizing tasks, formulation of task functions, generation of appropriate control signal. The intuitiveness of NSB method in terms of its geometrical interpretation and its limitations are presented for different conditions and combinations in section 3.3, 3.4.

3.1 Mathematical Framework

It is known that, in real world scenario, single robot is not of much use when it comes to accomplishment of several tasks at the same time. Therefore mathematical framework needed for a MRS is highlighted and carried here with the NSB approach, which first handles the task composition [19], then a relative priority is assigned to the individual task functions by resorting to the task-priority inverse kinematics [21] [20] for redundant manipulators. It is different from other methods in a way the output of individual behaviors are combined to compose the final behavior. Mathematical framework of NSB, will make it more clear.

The elementary behaviors are represented by means of a task function:

$$\sigma_i = f\left(\mathbf{x}_i\right) \tag{3.1}$$

where $\sigma_i \in \mathbb{R}^m$ denotes the task to be controlled for the i^{th} robot. Eq. 3.1 is called direct kinematics equation, which is used to compute value of task function based on given parameters. Considering the neighboring robots static, and σ_i to be differentiable,

$$\dot{\sigma_i} = \mathbf{J}_i(\mathbf{x}_i)\dot{\mathbf{x}}_i, \tag{3.2}$$

where $\mathbf{J}_i \in \mathbb{R}^{m\mathrm{X}2}$ denotes configuration dependent task Jacobian matrix for i^{th} robot. By inverting the locally linear mapping in eq. 3.2, motion references $\mathbf{x}_{d_i}(t)$ for the i^{th} robot starting from desired value $\sigma_{d_i}(t)$ can be obtained

$$\dot{\mathbf{x}}_{d_i} = \mathbf{v}_{d_i} = \mathbf{J}_i^{\dagger}\dot{\sigma}_{d_i}, \tag{3.3}$$

where $\mathbf{J}_i^{\dagger} = \mathbf{J}_i^T\left(\mathbf{J}_i\mathbf{J}_i^T\right)^{-1}$ denotes the penrose pseudo inverse of $\mathbf{J}_i$. Discrete time integration of robot's reference velocity results in numerical drift of reconstructed position of robot. Hence Closed Loop Inverse Kinematics version of algorithm is used to counteract undesired numerical drift to yield

$$\dot{\mathbf{x}}_{d_i} = \mathbf{v}_{d_i} = \mathbf{J}_i^{\dagger}\left(\dot{\sigma}_{d_i} + \mathbf{\Lambda}\tilde{\sigma}_i\right), \tag{3.4}$$

where $\mathbf{\Lambda}$ is a positive definite matrix and

$$\tilde{\sigma}_i = \sigma_{d_i} - \sigma_i. \tag{3.5}$$

For multiple behaviors, a priority index is assigned to each behavior denoted by q ($q = 1$

mean a task with highest priority); the control signal associated with the q^{th} priority task, for i^{th} robot, is given by

$$\mathbf{v}_i^{(q)} = \mathbf{J}_i^{\dagger(q)} \left(\dot{\sigma}_{d_i}^{(q)} + \mathbf{\Lambda}^{(q)} \tilde{\sigma}_i^{(q)} \right). \tag{3.6}$$

With reference to [22][23], for two behaviors

$$\mathbf{v}_{d_i} = \mathbf{v}_i^{(1)} + \left(\mathbf{I} - \mathbf{J}_i^{\dagger(1)} \mathbf{J}_i^{(1)} \right) \cdot \mathbf{v}_i^{(2)} \tag{3.7}$$

Therefore, the motion of i^{th} robot is governed in accordance to eq. 3.7.

The above formulations have interesting geometrical interpretation as discussed in section 3.3. The NSB fulfills a task with highest-priority always at nonsingular configuration. The lower-priority task, subtask are fulfilled in a subspace only where they do not conflict with higher priority tasks. Meaning, every individual task reaches a sub-optimal condition which optimize the task respecting constraints imposed by highest-priority task. Now, the task function $\sigma_i^{(q)}$ for *OA* with priority index 1 and *GTG* behavior-with priority index 2 maintained and continued to end.

3.2 Formulation of Task Functions

The global task to be achieved by multi robot system is divided into elementary subtasks of different priority. Each task is modeled as a task function with a predefined priority. Goal oriented and Obstacle avoidance task functions are illustrated in subsequent sections.

3.2.1 Goal Oriented Task Function

This behavior comes to picture when a robot moves from a point to its target location, known as goal. This task function yield a value, based upon which the closed loop inverse kinematic relation drives robot with a velocity proportional to the displacement vector pointing from its position towards goal.

Let ${\sigma^{(2)}}_i$ represent the task function to be controlled for i^{th} robot to reach the goal:

$$ {\sigma^{(2)}}_i = \mathbf{x}_i \text{ and } {\sigma^{(2)}}_{d_i} = \mathbf{x}_g, \tag{3.8} $$

where $\mathbf{x}_g \in \mathbb{R}^2$ denote the position of goal. The associated Jacobian and it's penrose pseudo inverse is given by

$$ \mathbf{J}_i^{(2)} = \mathbf{I} = \mathbf{J}_i^{\dagger(2)} \tag{3.9} $$

With reference to eq. 3.4 and eq. 3.5

$$ \mathbf{v}_i^{(2)} = \mathbf{K}_{GTG} \cdot (\mathbf{x}_g - \mathbf{x}_i) = \mathbf{u}_{GTG_i}, \tag{3.10} $$

where $\mathbf{K}_{GTG}$ is a positive definite matrix.

3.2.2 Obstacle Avoidance Task Function

Each robot needs to avoid both environmental obstacle and other robots. When an obstacle is present in robot advancing direction, the task function describes a driving velocity aligned to robot-obstacle direction. This task function incorporates a disk abstraction safe distance *d* around $\mathbf{x_{ob}}$ and only tasks with priority index lower than obstacle avoidance, if any, are allowed to produce motion components tangent to circle of radius *d*.

Let ${\sigma^{(1)}}_i$ represent task function to be controlled for i^{th} robot to maintain a safe distance d from obstacle :

$$ {\sigma^{(1)}}_i = ||\mathbf{x}_i - \mathbf{x}_{ob}|| \text{ and }, {\sigma^{(1)}}_{d_i} = d, \tag{3.11} $$

where $\mathbf{x}_{ob} \in \mathbb{R}^2$ denote coordinates of obstacle. The associated Jacobian and it's penrose pseudo inverse is given by

$$ \mathbf{J}_i^{(1)} = \left[\frac{\mathbf{x}_i - \mathbf{x}_{ob}}{||\mathbf{x}_i - \mathbf{x}_0||} \right]^T = \hat{\mathbf{r}}_i^T \tag{3.12} $$

$$\mathbf{J}_i^{\dagger(1)} = \hat{\mathbf{r}}_i. \tag{3.13}$$

With reference to eq. 3.11, 3.12, 3.13, 3.4, and 3.5

$$\mathbf{v}_i^{(1)} = \mathbf{J}_i^{\dagger(1)} k_{OA} \left(d - ||\mathbf{x}_i - \mathbf{x}_{ob}||\right) = \mathbf{u}_{OA_i}, \tag{3.14}$$

where $k_{OA} \in \mathbb{R}^+$. The null space projector matrix associated with $\mathbf{J}_i^1$ is given by

$$\mathbf{N}_{1_i} = \mathbf{I} - \hat{\mathbf{r}}_i \cdot \hat{\mathbf{r}}_i^T \tag{3.15}$$

The OA function is developed for point obstacles or the obstacles which can be approximated as a circle. It is not a necessary assumption because obstacles may be represented by straight lines or concave curves. Hence appropriately changing and signifying point $\mathbf{x}_{ob}$, representing coordinates of obstacle closest to robot at a given instant.

3.3 Geometrical Interpretation and Generating Appropriate Control Signal

Consider a 2D plane and let $\mathbf{v}_1$ and $\mathbf{v}_2$ be corresponding velocities for task 1 and task 2 respectively as shown in fig. 3.1.

According to section 3.1 task 2 is a lower priority task following task 1, therefore velocity corresponding to task 2 is projected into null space of task 1 Jacobian ; then, velocity component affecting primary task are cut off and remaining component is added to the velocity $\mathbf{v}_1$ corresponding to task 1. In this case the orthogonality between two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ is the compatibility condition among these two tasks, i.e., orthogonal vector of $\mathbf{v}_1$ will be null space for $\mathbf{v}_1$ (in this case) and $\mathbf{v}_2$ is projected onto it and vice-versa if the order of priority is reversed.

It must be noticed that, in NSB, a full-dimensional highest-priority task would subsume lower priority tasks. If the Jacobian matrix is full rank, its null space would be

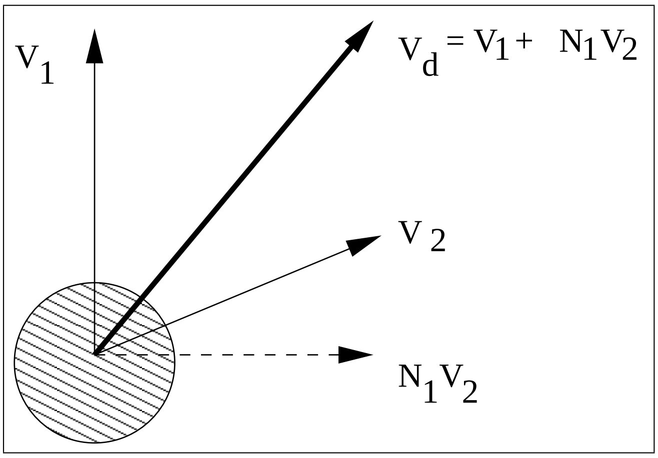


Figure 3.1: Geometric Interpretation of NSB.

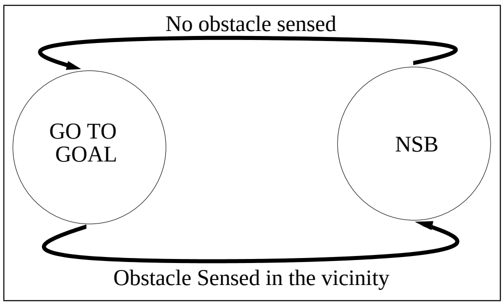


Figure 3.2: Hybrid system to accomplish navigation objective.

empty and thus whole vector $\mathbf{v}_2$ would be filtered out. Therefore NSB control always fulfils the highest priority task, whereas the lower-priority tasks are fulfiled only in a sub-space where they do not conflict with the ones having higher priority.

The appropriate control signal generated from the two task functions defined in section 3.2.1 and section 3.2.2 is given by

$$\mathbf{v}_{d_i} = \mathbf{u}_{OA_i} + \mathbf{N}_{1_i} \cdot \mathbf{u}_{GTG_i} \tag{3.16}$$

Here GTG being task of lower priority, is projected on the column space of null space projector matrix of Jacobian associated with OA. Finally, OA- higher priority behavior is added to obtain control signal. This is explained as shown in fig. 3.1 .

3.4 Limitations of Combining Goal Oriented and Obstacle Avoidance Behaviors in NSB framework

This section intuitively motivates the need of new control strategy, due to failure of conventional NSB based control strategy, for single mobile in an unknown environment consisting of rectangular obstacles.

In an environment, containing only circular obstacles, combining *GTG* and *OA* behaviors, in NSB, guarantees accomplishment of navigation objectives i.e., reaching goal and avoiding obstacle [24]. In case of rectangular obstacles as shown in fig. 4.2, there exist a *point of complete confliction* $[x_{ob_1} - d \quad y_g]^T$, where two behaviors are anti-parallel and hence projection of lower priority task (*GTG*) on to the null space of higher priority task (*OA*) results in a zero vector; therefore robot is, driven purely by the *OA* behavior, normal to the surface i.e. away from the PCC; since, robot is now at a distance greater than d, *GTG* behavior comes into picture and robot is driven towards PCC. Thus the robot perpetually oscillate in neighborhood of *point of complete confliction*(PCC). Highlighting the same is represented pictorially in fig. 3.3. Let robot be represented by a circle, with $\mathbf{p} = [x \ y]^T \in \mathbb{R}^2$ as it's position and $\mathbf{p}_g = [x_g \ y_g]^T \in \mathbb{R}^2$ are position coordinates of the goal represented by a hexagon. By controlling motion of robot, in accordance to hybrid system depicted in fig. 4.1, robot initially heads towards goal (trajectory of robot is depicted in black color) till it reaches near an obstacle. Let the robot maintain a safe distance d from obstacle, thus, NSB is activated as soon as the distance of the robot from obstacle tends to fall below d (Obstacle Avoidance behavior is depicted in red). It can be observed that at PCC, the robot exhibits type-1 Zeno phenomenon. At PCC, *OA* behavior is not shown for the sake of clarity; also the trajectory at PCC consists of two anti parallel black arrows, which portray expected perpetual oscillation of robot in neighborhood of PCC.

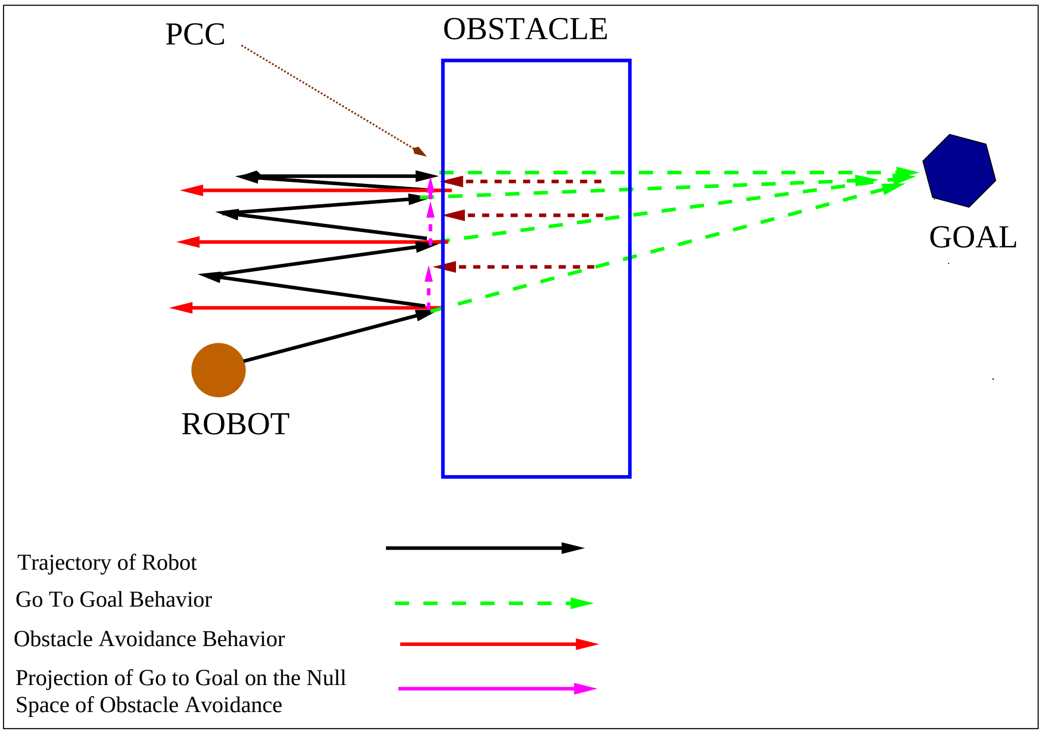


Figure 3.3: Combining go to goal and obstacle avoidance behaviors in NSB frame-work.

CHAPTER 4

FLOCKING AND CONSENSUS PROBLEM

This chapter utilizes the mathematical tools, discussed in chapter 3, to develop a modified null space based control strategy to address flocking and consensus problem. It mathematically formulates statement of problem the dissertation strives to workout. The failure of conventional control strategy- combining GTG and OA behaviors in NSB framework- to accomplish navigation objective of reaching the goal in presence of rectangular obstacles. This chapter provides concrete proof that a mobile robot exhibits type-1 Zeno phenomenon along a line, where GTG and OA behaviors, in presence of rectangular obstacles, are anti-parallel to each other and hence robot is unable to reach the goal. Due to the failure of existing control strategy, one or more mobile robots of MRS are unable to reach neighborhood of Rendezvous point. To evade exhibition of type-1 Zeno phenomenon by a mobile robot, the FW behavior, discussed in section 2.1.3 is exploited, to propose a modified null space based control strategy as in section 4.3.

4.1 Objective of Flocking and Consensus Problem

Consensus problem invokes domain of control theory to generate control signals, to facilitate the agreement of all robots of MRS at a point known as consensus or Rendezvous point as system drives towards steady state. In addition to reaching the consensus point, control strategy should prevent collision of each robot in the MRS with an obstacle or neighboring robot. Since, agreement of all robots at consensus point is not possible unless they collide with each other, which would be undesired; so objective would be to generate appropriate control signals to facilitate flocking of robots in the neighborhood of consensus point.

After briefly discussing issues, which will be addressed in this dissertation, mathematical formulation of statement of problem, which shall help proposing a new control strategy is presented in section 4.3. Let $\mathbf{x} = [\mathbf{x}_1 \;\; \mathbf{x}_2 \cdots \mathbf{x}_N]^T$ is state vector for a N robot MRS where $\mathbf{x}_i \in \mathbb{R}^2$ denote position of i^{th} robot. Let $\mathbf{u}_i$ denote control signal driving i^{th} robot from an arbitrary initial position to the consensus point $\mathbf{c} \in \mathbb{R}^2$. Therefore, the control law governing dynamics of each robot is given by:

$$\dot{\mathbf{x}}_i = \mathbf{u}_i = k_{GTG} \left(\mathbf{c} - \mathbf{x}_i \right), \tag{4.1}$$

where $k_{GTG} \in \mathbb{R}^+$. The MRS can thus be modeled as a Linear Time Invariant (LTI) system described by:

$$\dot{\mathbf{x}} = k_{GTG} \left(\mathbf{A}\mathbf{x} + \mathbf{b} \right), \tag{4.2}$$

where,

$$\mathbf{A} \in \mathbb{R}^{2N \text{ X } 2N} = \begin{bmatrix} -1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ \vdots & & & & \\ 0 & \cdots & 0 & -1 & 0 \\ 0 & 0 & 0 & \cdots & -1 \end{bmatrix} \text{ and } \mathbf{b} \in \mathbb{R}^{2N} = \begin{bmatrix} \mathbf{c} \\ \mathbf{c} \\ \vdots \\ \mathbf{c} \end{bmatrix}. \tag{4.3}$$

The eigen values associated with system matrix of the aforementioned LTI system are strictly negative (-1 here) and $||\mathbf{b}|| < H < \infty$, therefore the above system satisfies BIBO stability criterion and state converges to a point $\mathbf{c}^*$ given by:

$$\mathbf{c}^* = \lim_{t \to \infty} \mathbf{x}(t), \tag{4.4}$$

where t denotes time. The solution to the LTI system described in eq. 4.2 is given by:

$$\mathbf{x} = k_{GTG} \left(e^{\mathbf{A}t} \cdot \mathbf{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)} \cdot \mathbf{b} \; d\tau \right) \tag{4.5}$$

and at steady state

$$\lim_{t \to \infty} \mathbf{x}(t) = \mathbf{c} = \mathbf{c}^* \tag{4.6}$$

Eq. 4.5 validates choice of control signal driving all robots towards the consensus point, however, it assumes that environment doesn't have obstacles and path of robots can intersect in time which in simple words mean collision with other robots. The generation of appropriate control signals, guiding motion of each robot of MRS, becomes challenging in presence of obstacles in the environment. Additionally, satisfying the constraint:

$$\mathbf{x}_i(t^*) \neq \mathbf{x}_j(t^*) \ \ \forall\, i \neq j \tag{4.7}$$

further makes the generation of control signals, a more difficult task. Note that $t^* \in [t_0,\ t_f]$, where t_0 and t_f refers to time at initial and final state; eq. 4.5 assumes $t_0 = 0$.

The control strategy shall drive robots of MRS such that:

1. All robots flock around the consensus point.
2. None of the robot collides with an obstacle.
3. None of the robot collide with any of robots in its neighborhood. Concretely, it should abide to constraint depicted in eq. 4.7.

4.1.1 Challenges

This subsection highlights the challenges that have to be faced in generating appropriate control signals to address the problem statement formulated in the previous section. The problem statement can be effectively decomposed into sub-problems and each sub-problem will be catered by a behavior and therefore each behavior accomplishes the corresponding task and resolves the sub-problem.

Addressing three navigation objectives, there exists three behaviors; the main challenges will be to :

1. Manage more than one task simultaneously.

2. Resolving conflicting tasks or behaviors.

The challenges basically point to a situation of correct "decision making" in choosing one of the behaviors from three available. Also, for instance, consider a situation where a robot encounters an obstacle and a neighboring robot in it's proximity while heading towards goal the question arises- what should robot do? Should it avoid the obstacle, nearest neighbor or should it continue heading towards the goal. Such a situation, where sub-tasks are in confliction of each other are known as *conflicting situations* and corresponding tasks are termed as *conflicting tasks*.

Chapter 3 discussed Null space based behavioral control, wherein conflicting behaviors can be effectively managed by assigning relative priority to each subtask.

4.2 Failure of Conventional Control Strategy

NSB framework elucidated in chapter 3 very effectively manages conflicting behaviors that may arise during navigation. A control strategy, proposed by *Antonelli et. al.*,[24] involved combining GTG, OA and another behavior which would avoid nearest neighboring robot in decentralized NSB framework to accomplish navigation objectives, mentioned in section 4.1, in an unknown environment consisting only circular obstacles.

First, a mathematical proof, claiming the failure of control strategy, proposed in [23] for a single mobile robot in an environment consisting rectangular obstacles is presented. Since the strategy proposed in [24] is indeed an extension of [23], wherein additional lattice formation behavior was included (to avoid nearest neighbor and form alpha lattice structure. Note, the dissertation does not aim at forming an alpha lattice geometry), individual robots of MRS, encountering rectangular obstacles *may not* be able to find a way to the goal.

Proposition 1. *A mobile robot, initialized at* $\mathbf{p}_0 = [x_0 \; y_0]^T$*, heading towards the goal* $\mathbf{p}_g = [x_g \; y_g]^T$*, in an environment containing rectangular obstacle shall exhibit type-1 Zeno phenomenon if line joining* $\mathbf{p}_0$ *and* $\mathbf{p}_g$ *intersects any two parallel edges of the rectangle.*

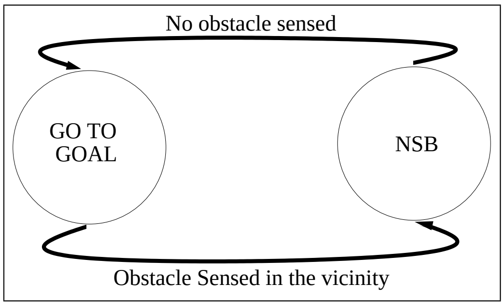


Figure 4.1: Hybrid system to accomplish navigation objective.

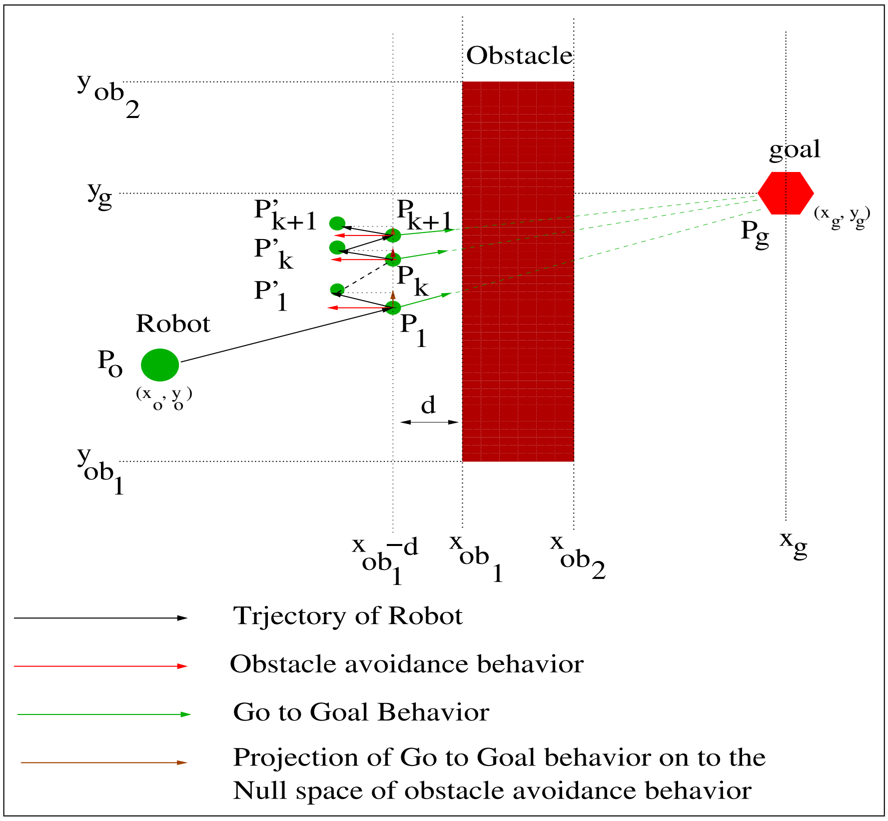


Figure 4.2: Depicting *Point of Complete Confliction.*
(PCC)

Proof. Let $\mathbf{p}_1$ denote the *point of first encounter* as shown in fig. 4.2 , where velocity vector changes it's direction for the first time due to switching of behavior in accordance to fig. 4.1. Note, the state NSB in fig. 4.1, in this section mean an obstacle is sensed by robot and hence it's motion henceforth will be controlled by combining GTG and OA behaviors in NSB framework. Therefore

$$\mathbf{p}_1 = [x_1 \; y_1]^T = \begin{bmatrix} x_{ob_1} - d \\ y_g + m_1(x_{ob_1} - d - x_g) \end{bmatrix} \tag{4.8}$$

where

$$m_1 = \frac{y_g - y_0}{x_g - x_0}$$

d denotes minimum safety distance maintained from obstacle and $x = x_{ob_1}$ is the abscissa of the edge of obstacle under consideration. Let Δt_1 denote the time interval for which dynamics of mobile robot are governed by NSB and it reaches a point

$$\mathbf{p}'_1 = \mathbf{p}_1 + \mathbf{v}_1 \Delta t_1 \tag{4.9}$$

after moving with a $\mathbf{v}_1$ velocity during the time interval Δt_1

$$\mathbf{v}_1 = \begin{bmatrix} -\beta & \alpha(y_g - y_1) \end{bmatrix}^T \text{ and } \begin{cases} \beta = -k_b(x_{ob_1} - d - x_1) \\ k_b \in \mathbb{R}^+ \\ \alpha \in \mathbb{R}^+ \end{cases} .$$

Velocity due to OA behavior will always be constant along negative direction of x-axis: $[-\beta \;\; 0]^T$, at all encounter points and velocity corresponding to GTG behavior projected along null space of OA behavior will always be along y-axis.

$\mathbf{p}'_1$ is assumed to be at a distance which is greater than d and hence robot's dynamics will now be governed according to GTG behavior for an interval Δt_2 after which it reaches second encounter point

$$\mathbf{p}_2 = \mathbf{p}'_1 + \mathbf{v}'_1 \Delta t_2, \tag{4.10}$$

where

$$\mathbf{v}'_1 = \alpha \begin{bmatrix} x_g - x'_1 & y_g - y'_1 \end{bmatrix}^T$$

Both Δt_1 and Δt_2 are small time intervals and hence translation of robot from each $\mathbf{p}_k$ to $\mathbf{p}'_k$ may be assumed to occur in Δt_1 and in a similar way translation from each $\mathbf{p}'_k$

to $\mathbf{p}_{k+1}$ can be assumed to occur in a time duration Δt_2. However, Δt_1 and Δt_2 both are comparable as assumed to be approximately equal to each other

$$\Delta t_1 = \Delta t_2 = \Delta t.$$

Based on presented analysis so far, assume that k^{th} encounter point is known. Suppose the current position of robot as $\mathbf{p}_k = [x_k \;\; y_k]^T$; therefore due to NSB the robot reaches:

$$\mathbf{p}'_k = [x'_k \;\; y'_k]^T = \begin{bmatrix} x_{ob_1} - d - \beta\Delta t \\ y_k + (y_g - y_k)\alpha\Delta t \end{bmatrix}. \tag{4.11}$$

Writing $\mathbf{p}_{k+1}$ as a function of $\mathbf{p}_k$

$$\mathbf{p_{k+1}} = [x_{k+1} \;\; y_{k+1}]^T = \begin{bmatrix} x_{ob_1} - d \\ y_g + m'_k(x_{ob_1} - d - x_g) \end{bmatrix}, \tag{4.12}$$

where

$$m'_k = \frac{y_g - y'_k}{x_g - x'_k} \tag{4.13}$$

Using eq. 4.11, eq. 4.12, eq. 4.13,

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_{ob_1} - d \\ ay_k + b \end{bmatrix}, \tag{4.14}$$

where

$$\begin{aligned} a &= \tfrac{\alpha\Delta t - 1}{M} \\ & \qquad\qquad \text{and } M = x_g - x_{ob_1} + d - \beta\Delta t \\ b &= y_g\left(\tfrac{M+1-\alpha\Delta t}{M}\right) \end{aligned} \tag{4.15}$$

For practical situation, $|a| < 1$ and $|b| < H < \infty$, therefore the system described by eq. 4.14 satisfies BIBO stability criterion and thus

$$\left|\lim_{k\to\infty} y_k\right| < H < \infty. \tag{4.16}$$

and

$$\lim_{k \to \infty} y_k = y^*. \tag{4.17}$$

Computing the steady state of y_k, depicted in eq. 4.14, results

$$\lim_{k \to \infty} y_{k+1} = a \lim_{k \to \infty} y_k + \lim_{k \to \infty} b. \tag{4.18}$$

Using eq. 4.17, eq. 4.18 reduces to

$$y^* = \frac{b}{1-a} = y_g \tag{4.19}$$

thus,

$$\lim_{k \to \infty} \mathbf{p}_k = \begin{bmatrix} x_{ob_1} - d \\ y_g \end{bmatrix} \tag{4.20}$$

Now, compute $\lim\limits_{k \to \infty} \mathbf{p}'_k$ and hence

$$\mathbf{p}'_{k+1} = [x'_{k+1} \;\; y'_{k+1}]^T = \begin{bmatrix} x_{k+1} - \beta \Delta t \\ y_{k+1} + \alpha(y_g - y_{k+1})\Delta t \end{bmatrix} \tag{4.21}$$

using eq. 4.20:

$$\lim_{k \to \infty} \mathbf{p}'_{k+1} = \begin{bmatrix} x_{ob_1} - d - \beta \Delta t \\ y_g \end{bmatrix} \tag{4.22}$$

From eq. 4.20 and eq. 4.22, it may be theoretically concluded that robot exhibits type-1 Zeno phenomenon. Another subtlety hidden in the proof: according to chosen parameters, it may so happen that y_k or y'_k becomes greater than y_g for some k but as $k \to \infty$ they finally converge in accordance to eq. 4.20 and eq. 4.22.

The aforementioned Proposition 1 will hold good for a MRS if and only if all points on the trajectory of robot intersects any two parallel edges of the rectangle. It may also happen that few of robots due to action of a behavior-which avoids nearest neighbor-get driven off the vertex of rectangular obstacle and hence find a way to the Rendezvous point. □

4.3 Proposed Control Strategy

The previous section highlighted the setbacks associated with conventional control strategy [24] wherein GTG, OA behaviors along with another behavior which avoids nearest neighbor are combined in presence of rectangular obstacles in NSB framework. In this section, a modified null space based control strategy to drive a MRS towards the neighborhood of consensus point, satisfying navigation objectives, is presented.

In the proceeding analysis, it is assumed that a robot is able to distinguish between an obstacle and a neighboring robot; there are a total of four different modes, in which a robot can operate during navigation:

1. **No obstacle and no other robot sensed**: The robot head towards goal and its dynamics are based on GTG behavior:

$$\mathbf{v_d}_i = \mathbf{u}_{GTG_i}.$$

2. **Obstacle sensed and no other robot sensed**: The robot's dynamics are based on pure FW behavior and direction to follow wall is governed by:

$$\mathbf{v}_{d_i} = \begin{cases} \mathbf{u}^{C}_{FW_i} \text{ if } \mathbf{u}^{C}_{FW_i} \cdot \mathbf{u}_{GTG_i} > 0 \\ \mathbf{u}^{CC}_{FW_i} \text{ if } \mathbf{u}^{CC}_{FW_i} \cdot \mathbf{u}_{GTG_i} > 0 \end{cases}.$$

3. **Obstacle and other robot sensed**: The robot follows wall of the obstacle while avoiding collision with neighboring robots. Let $\mathbf{u}_{RA_{ij}}$ be the behavior that controls i^{th} robot motion preventing it from colliding with j^{th} robot. The j^{th} robot can be considered to be an obstacle and using OA behavior can help i^{th} robot to avoid colliding with j^{th} robot located at $\mathbf{x}_j$. In this section assume i^{th} robot avoids nearest neighbor at $\mathbf{x}_j$. The dynamics are governed by

$$\mathbf{v_d}_i = \mathbf{u}_{FW_i} + \mathbf{N}_{1_i} \cdot \mathbf{u}_{RA_{ij}}$$

Another interesting fact to observe in dynamics of this state: the projection of lower priority task, i.e. avoiding the neighbor, is projected on to the null space of the Jacobian of the task function associated with *Obstacle Avoidance* behavior and not the *Follow Wall* behavior. This basically speeds or slows the following of wall of obstacle by the robot. Also note, once robot starts following boundary of obstacle, it will continue to follow it unless the conditions mentioned in section 2.1.3 are satisfied.

4. **No obstacle sensed but other Robot sensed:** The conventional blending of GTG and OA behaviors in NSB framework is employed and the robot dynamics are governed in accordance to:

$$\mathbf{v_d}_i = \mathbf{u}_{RA_{ij}} + \mathbf{N}_{ij} \cdot \mathbf{u}_{GTG_i},$$

where $\mathbf{N}_{ij}$ is null space projector matrix associated with the Jacobian of the task function of OA behavior with position of obstacle as position of closest neighboring robot.

CHAPTER 5

SIMULATIONS AND RESULTS

This chapter presents simulations ans results validating the proposed theoretical formulations so far. Section 5.1 will portray MATLAB simulation for MRS, whereas section 5.2 will portray simulations performed in Java using MASON library [25]-[27].

5.1 MATLAB simulation

In this section, MATLAB based simulations to validate the mathematical modeling proposed in previous chapters is presented. First, the state response of three robot MRS heading towards the consensus point, $[15 \quad 11]^T$, is simulated as shown in fig. 5.1; the simulation parameters and initial conditions are depicted in table 5.1. Also note that the behavior, which avoids neighboring robot is not included in this simulation.

The centroid is computed by:

$$\mathbf{c} = \frac{1}{N} \cdot \sum_{i=1}^{N} \mathbf{x}_i$$

and control law is governed by the following dynamical equation:

$$\mathbf{u}_i = \mathbf{c} - \mathbf{x}_i.$$

The system can thus be modeled as LTI system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$$

Table 5.1: Simulation Parameters for stability analysis

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3
Position of Robot 1 ($\mathbf{x}_1$)	$[20\ 10]^T$
Position of Robot 2 ($\mathbf{x}_2$)	$[12\ 19]^T$
Position of Robot 3 ($\mathbf{x}_3$)	$[13\ 04]^T$

$$\mathbf{A} = \frac{1}{3}\begin{pmatrix} -2 & 0 & 1 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 & 0 & 1 \\ 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 \\ 1 & 0 & 1 & 0 & -2 & 0 \\ 0 & 1 & 0 & 1 & 0 & -2 \end{pmatrix}$$

Also by Diagonalization $\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}$, where

$$\mathbf{S} = \begin{pmatrix} 0.4082 & -0.6272 & 0.2732 & -0.1789 & -0.1673 & 0.5526 \\ -0.5774 & -0.2408 & 0.4669 & 0.2396 & 0.5526 & 0.1673 \\ -0.4082 & 0.4283 & 0.0047 & -0.5626 & -0.1673 & 0.5526 \\ 0.5774 & 0.5055 & 0.2770 & -0.0318 & 0.5526 & 0.1673 \\ 0 & 0.1989 & -0.2779 & 0.7415 & -0.1673 & 0.5526 \\ 0 & -0.2648 & -0.7439 & -0.2078 & 0.5526 & 0.1673 \end{pmatrix} = [\mathbf{S}_1 \cdots \mathbf{S}_6]$$

$$\mathbf{\Lambda} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

To represent initial state as linear combination of eigen vectors S_i, the following system of linear equations have to be solved

$$\mathbf{S} \cdot \mathbf{c} = \mathbf{x}_0, \text{ yielding}$$

$$\mathbf{c} = [8.4621 \;\; 1.3200 \;\; 8.8643 \;\; 0.2707 \;\; 10.7062 \;\; 30.3871]^T = [c_1 \cdots c_6]^T$$

Thus, state of the system, as a function of time will be given by:

$$\mathbf{x}(t) = \sum_{i=1}^{6} c_i \cdot \mathbf{S}_i \cdot e^{\lambda_i t},$$

where $\lambda_i \in eig(\mathbf{A})$. At steady state

$$\lim_{t \to \infty} \mathbf{x}(t) = c_5 \cdot \mathbf{S}_5 + c_6 \cdot \mathbf{S}_6 = [15 \;\; 11 \;\; 15 \;\; 11 \;\; 15 \;\; 11]^T$$

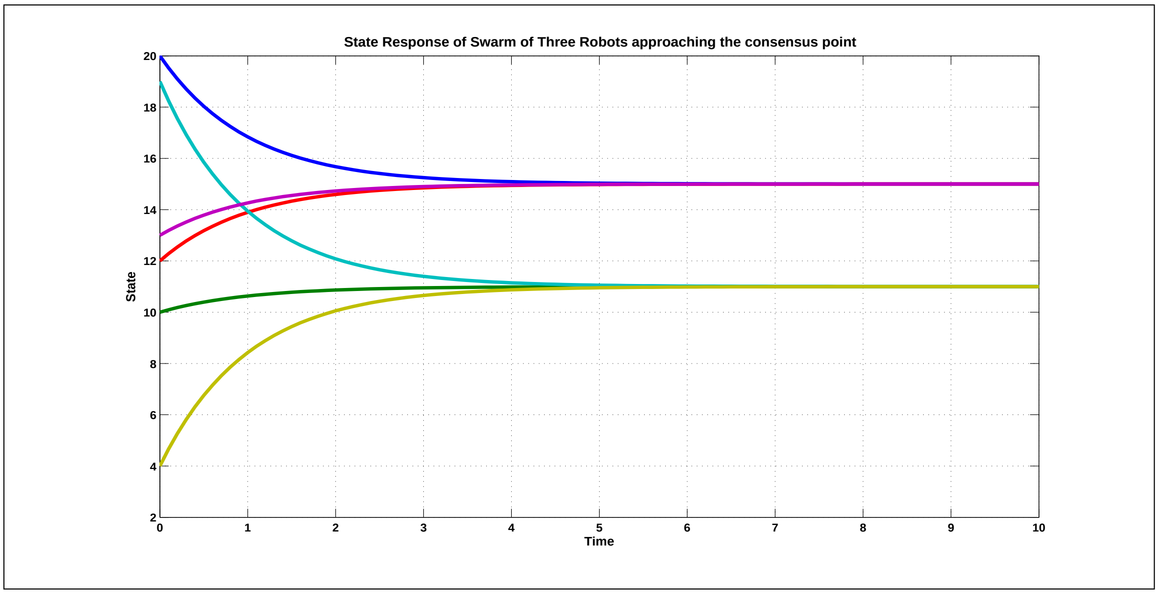


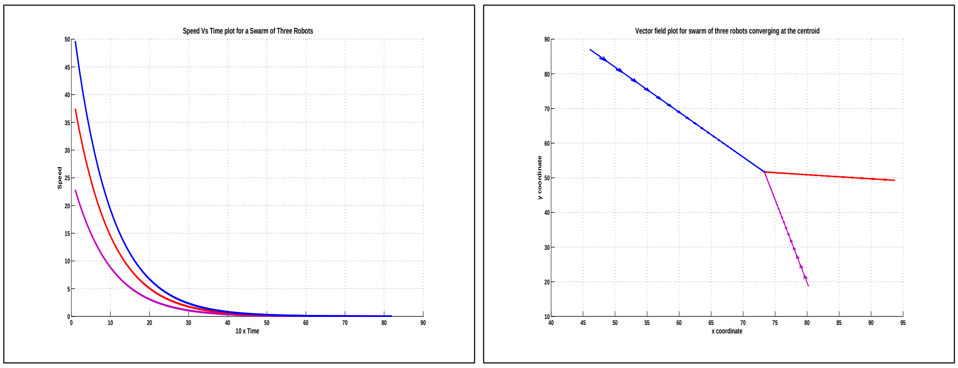
Figure 5.1: State response of 3 robot MRS heading towards centroid.

It is very evident from obtained results depicted in fig. 5.1 and aforementioned analysis that choice of control signal is *good* and hence system is stable and all robots converge at the centroid at steady state.

Secondly, the trajectory of 3,10,100 mobile robots, heading towards centroid is simulated. Simulation parameters can be found in table 5.2 and results are depicted in fig. 5.2, 5.3 and 5.4.

Table 5.2: Simulation Parameters for solving Rendezvous problem

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3,10,100
Robot Position	Randomly Initialized
Grid Size	100 X 100



(a) (b)

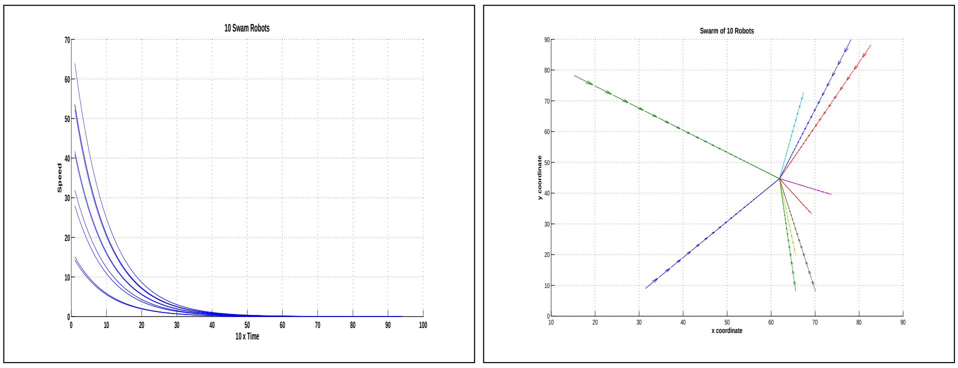
Figure 5.2: Speed vs time and vector field plot for 3 Robots MRS converging to centroid.

Speed vs time characteristic for all robots constituting the MRS is plotted and it is observed that as steady state is approached, speed of all robots reduce. The vector field plot of velocity on 2-D grid is also portrayed in fig. 5.2, 5.3 and 5.4 and one can observe that all the robots meet at Rendezvous point.

Thirdly, trajectory of two mobile robots heading towards the consensus point is simulated, by employing NSB when the obstacle is sensed at a distance less than the safe distance. The simulation parameters can be found in table 5.3 and the result is depicted in fig. 5.5.

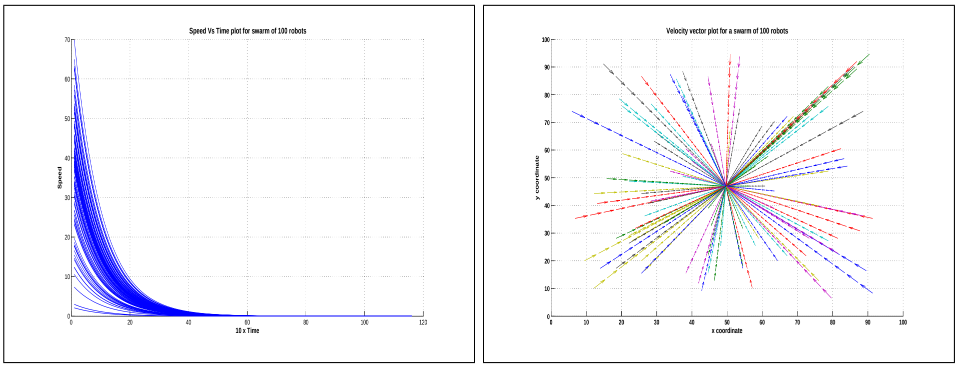
Fourthly, trajectory of three robots initialized at known position, heading towards the consensus point is simulated, in presence of four obstacles. Simulation parameters are depicted in table 5.4 and results are shown in fig. 5.6:

Finally, trajectory of six mobile robots, initialized at known position, heading towards the Rendezvous point is simulated, in presence of no obstacle. The behavior, which pre-



(a) (b)

Figure 5.3: Speed vs time and vector field plot for 10 Robots MRS converging to centroid.



(a) (b)

Figure 5.4: Speed vs time and vector field plot for 100 Robots MRS converging to centroid.

vents collision of one robot with other is employed to obtain the result shown in fig. 5.7. The associated simulation parameters are depicted in table 5.5.

5.2 Java simulations

In this section, the proposed theoretical and mathematical formulations are verified by means of various simulations carried out using MASON library [25]-[27]. The library has also been used previously simulate control algorithms pertaining to MRS [28]-[30]. Snapshot of simulations have been taken up and presented in the dissertation; videos of the simulations can be found in CD attached with the dissertation.

Consider a 100 X 100 grid where robots depicted by small Grey squares are randomly initialized, and their positions are recorded; the objective is to flock in neighborhood

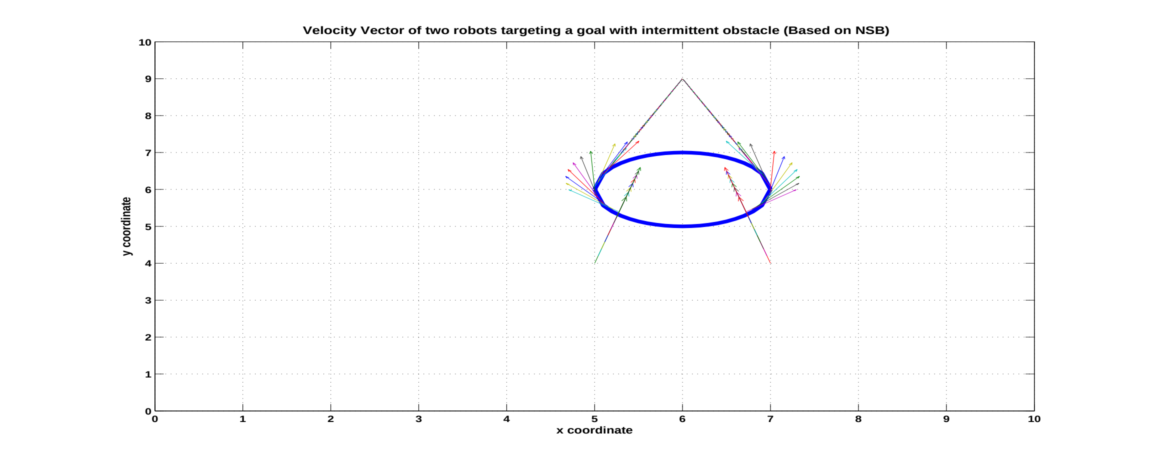


Figure 5.5: Vector field plot for 2 Robots MRS heading towards a goal point with intermittent obstacle in its path.

Table 5.3: Simulation Parameters for solving Rendezvous problem in presence of an obstacle.

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3,10,100
Position of Robot 1 ($\mathbf{x}_1$)	$[5\ 4]^T$
Position of Robot 2 ($\mathbf{x}_2$)	$[7\ 4]^T$
Position of Obstacle ($\mathbf{x}_{ob}$)	$[6\ 6]^T$
Consensus point	$[6\ 9]^T$
Clearing Distance from Obstacle (d)	1

of consensus point depicted by a violet circle. The general parameters common to all simulations performed in Java are depicted in table 5.6.

5.2.1 Elementary behaviors of MRS

Here, simulations of basic behaviors namely GTG, OA and behavior to avoid neighboring robot are presented as shown in fig. 5.8, 5.9 and 5.10 and finally MRS in complex environment containing circular obstacles by including all aforementioned behaviors as shown in fig. 5.13 is simulated. The robots are randomly initialized with goal location fixed at $[10\ 10]^T$ for majority of simulations of subsection. For simulation corresponding

Table 5.4: Simulation Parameters for solving Rendezvous problem for multi robots in presence of multiple obstacles

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3
Position of Robot 1 ($\mathbf{x}_1$)	$[0\ 4]^T$
Position of Robot 2 ($\mathbf{x}_2$)	$[8\ 4]^T$
Position of Robot 3 ($\mathbf{x}_3$)	$[9\ 7]^T$
Position of Obstacle 1	$[2\ 5]^T$
Position of Obstacle 2	$[4\ 7]^T$
Position of Obstacle 3	$[6\ 6]^T$
Position of Obstacle 4	$[8\ 8]^T$
Rendezvous point	$[5\ 9]^T$
Go to Goal Proportional gain	$0.4\mathbf{I}$
Safe distance	1
Obstacle Avoidance Proportional gain	1

to fig. 5.11, only single obstacle is considered and behavior to avoid neighboring robot is employed whereas for simulation presented in fig. 5.12 behavior to avoid neighboring robot is not included.

5.2.2 Follow wall behavior to avoid type-1 Zeno phenomenon

Initially **Proposition 1** is validated by allowing a single mobile robot to navigate in an environment containing square shaped obstacles followed by a similar simulation, which will extend the developed notion to MRS. Second, *Follow Wall* behavior for a single mobile robot is simulated to avoid exhibition of type-1 Zeno phenomenon followed by MRS. In case of circular obstacles, if line joining initial position of the robot and goal passes through center of circle, the robot will get struck. The simulation depicting this situation will also be presented; but the probability of having the three points collinear is very bleak.

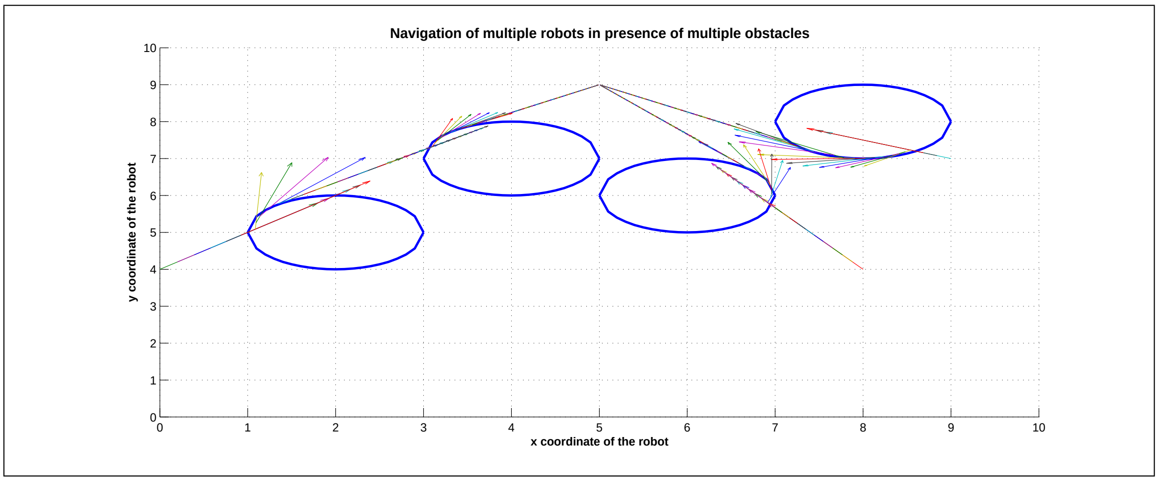


Figure 5.6: Vector field plot for MRS heading towards a goal point with intermittent obstacles in there paths.

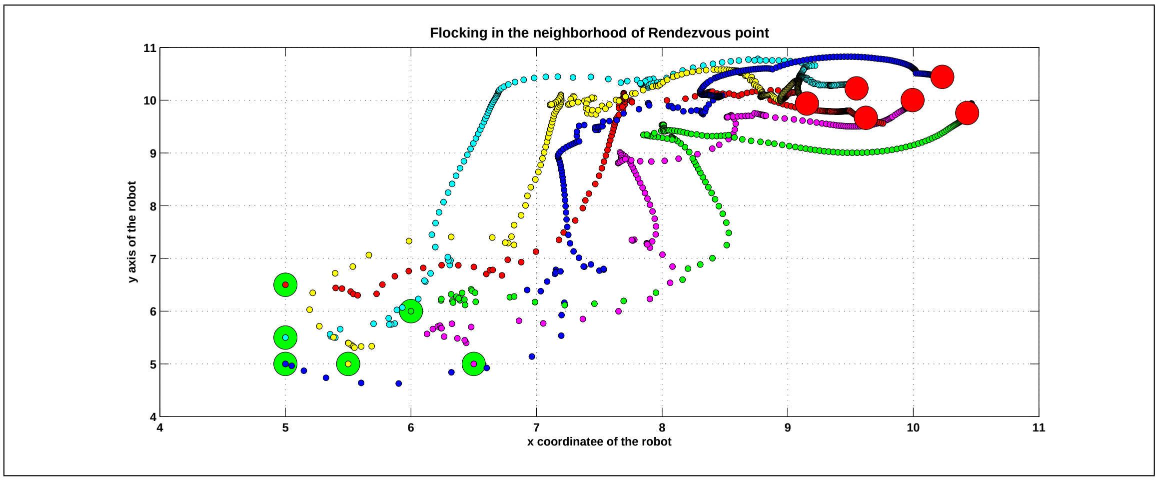


Figure 5.7: Flocking of MRS

5.2.2.1 Observations

The below mentioned observations were made while simulating a single mobile robot and MRS in an environment consisting square shaped and circular obstacle:

1. On simulating the trajectory of a single mobile robot as shown in fig. 5.14 initialized at $[40\ \ 80]^T$, heading towards the goal $[40\ \ 20]^T$, by combining GTG and OA behaviors in NSB framework,in presence of a circular shaped obstacle, it is observed that the robot gets struck and perpetually oscillates in the neighborhood of point of complete confliction, thereby unable to reach the goal (but does not collide

Table 5.5: Simulation Parameters for driving a six robot MRS, without colliding with each other, in the neighborhood of consensus point.

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	6
Position of Robot 1 ($\mathbf{x}_1$)	$[5.5\ 5]^T$
Position of Robot 2 ($\mathbf{x}_2$)	$[6.5\ 5]^T$
Position of Robot 3 ($\mathbf{x}_3$)	$[5\ 5.5]^T$
Position of Robot 4 ($\mathbf{x}_4$)	$[5\ 6.5]^T$
Position of Robot 5 ($\mathbf{x}_5$)	$[6\ 6]^T$
Position of Robot 6 ($\mathbf{x}_6$)	$[5\ 5]^T$
Rendezvous point	$[10\ 10]^T$
Go to Goal Proportional gain	0.8 **I**
Safe distance	1
Obstacle Avoidance Proportional gain	1.2

Table 5.6: General Simulation Parameters

Sampling time	0.01 units
Safe distance from obstacle	2 units
Grid size	100 X 100
Representation of Robot	Grey
Representation of Obstacle	Light green
Representation of Rendezvous point	Violet

with the obstacle).

2. On simulating trajectory of a single mobile robot as shown in fig. 5.15 initialized at $[50\ \ 80]^T$ towards goal $[40\ \ 20]^T$ by combining GTG and OA behaviors in NSB framework based on hybrid system depicted in fig. 4.1 in presence of a square shaped obstacle. It was observed that the robot gets struck and perpetually oscillates in the neighborhood of point of complete confliction thereby unable to reach goal (but does not collide with the obstacle).

3. A MRS consisting of 10 mobile robots randomly initialized heading towards Ren-

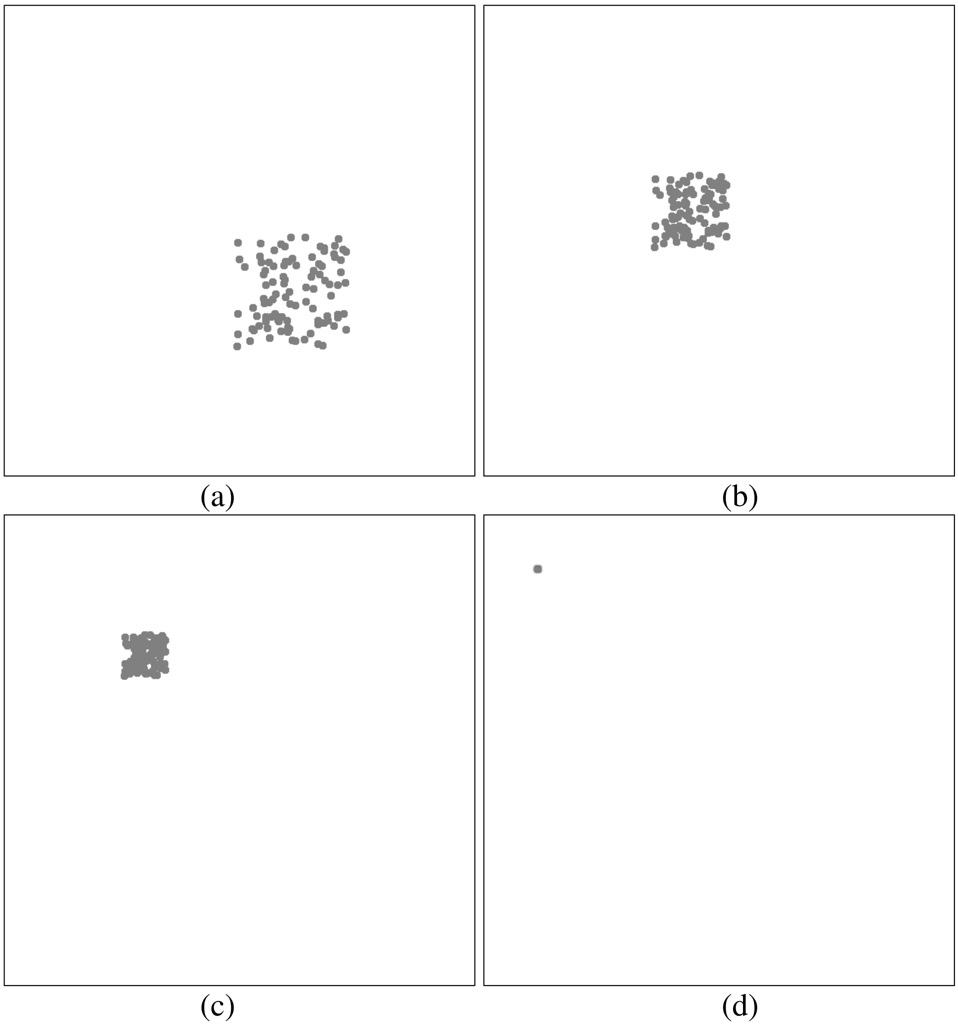


Figure 5.8: Solving consensus problem : (a), (b), (c), (d) depict the positions of the robot during simulation.

dezvous point $[40\ \ 10]^T$ in an environment consisting of square shaped obstacle is simulated. The dynamics of robot is governed by combining OA, GTG and behavior associated with avoiding nearest neighbor in NSB framework [24]; snapshots of obtained results are depicted in fig. 5.16. It was observed that four out of ten robots reach goal without colliding with each other or the obstacle and remaining six get struck (but they do not collide with their neighbors or obstacle). Also, out of total four robots reaching goal, one robot takes more time than other three.

4. *Follow Wall* behavior is simulated as shown in fig. 5.17 in presence of a circular shaped obstacle, for a single mobile robot initialized at $[40\ 80]^T$, which is heading

towards the goal $[40\ \ 20]^T$. It is observed that the robot, on reaching close to the obstacle follows it's boundary till it gets a clear shot at the goal and finally reaches the goal.

5. Finally, *Follow Wall* behavior is simulated as shown in fig. 5.18, in presence of a square shaped obstacle for a single mobile robot initialized at $[42\ \ 80]^T$ which is heading towards the goal $[40\ \ 20]^T$. It is observed that robot on reaching close to obstacle follow it's boundary till it gets a clear shot at goal and finally reaches the goal.

6. On simulating a MRS consisting of ten randomly initialized mobile robots, heading towards rendezvous point $[40\ 10]^T$ using the proposed control strategy described in section 4.3, It was observed that all the robots successfully reach the Rendezvous point, while avoiding collisions among themselves or square shaped obstacle. Snapshots of simulation can be found in fig.5.19.

5.2.2.2 Inference

This section is devoted to analysis of observed results and comparing them with expected theoretical formulations developed in this work. The following are inferences from aforementioned observations:

1. It is very evident from initial position of robot and coordinates of goal that a line joining these two points contain center of circle and thus combining GTG and OA behaviors in NSB framework: with reference to **Proposition 1**, robot is expected to exhibit type-1 Zeno phenomenon and thereby perpetually oscillate and get struck in neighborhood of point of complete confliction.

2. It is very evident from initial position of robot and coordinates of goal that a line joining these two points would definitely intersect two horizontal edges of obstacle

and thus combining GTG and OA behaviors in NSB framework with reference to **Proposition 1** robot is expected to exhibit type-1 Zeno phenomenon and thereby perpetually oscillate and get struck in the neighborhood of point of complete confliction. The observed results are in complete agreement to presented proposition.

3. In case of multiple robots apart from reaching goal and avoiding intermittent obstacle a robot has to avoid collisions with it's neighbors and thus path of a robot towards the Rendezvous point. It is not a straight line even in the absence of obstacles if it's neighbor is close to it. Due to specified constraint it may so happen that few of the robot's, near the vertex of the square, may escape the obstacle in such a way that they no more abide to the conditions mentioned in **Proposition 1** and hence find a path to goal. In presented simulation, three of robots very quickly avoid vertex of obstacles, prevent a collision with a neighboring robot in it's vicinity but one of the robot avoids vertex of obstacle after some delay; but majority of the robots get struck and thereby do not reach goal.

4. To avoid exhibition of type-1 Zeno phenomenon, a robot follows boundary of obstacle till it has a clear shot at goal and has made enough progress towards it. In this way, a robot can clearly pass through the neighborhood of point of complete confliction and hence reach the goal. The observed results completely abide to the theoretical formulations.

5. With reference to **Proposition 1**, a single mobile robot cannot reach the goal if the line joining it's initial position and coordinates of goal intersect any two parallel sides of the square; to avoid the exhibition of type-1 Zeno phenomenon, a robot follows boundary of obstacle till it has a clear shot at the goal and has made enough progress towards it. In this way, a robot can clearly pass through neighborhood of point of complete confliction and hence reach the goal. The observed results completely abide to the theoretical formulations.

6. The proposed control strategy, with reference to section 4.3, takes objectives all

into consideration i.e. reaching the goal while avoiding collision with obstacles and neighboring robots. The dynamics of a single robot in the MRS can be modeled using a hybrid system consisting of four states or modes, which are described in detail in section 4.3. The observed results are in complete concurrence with the proposed theoretical formulations.

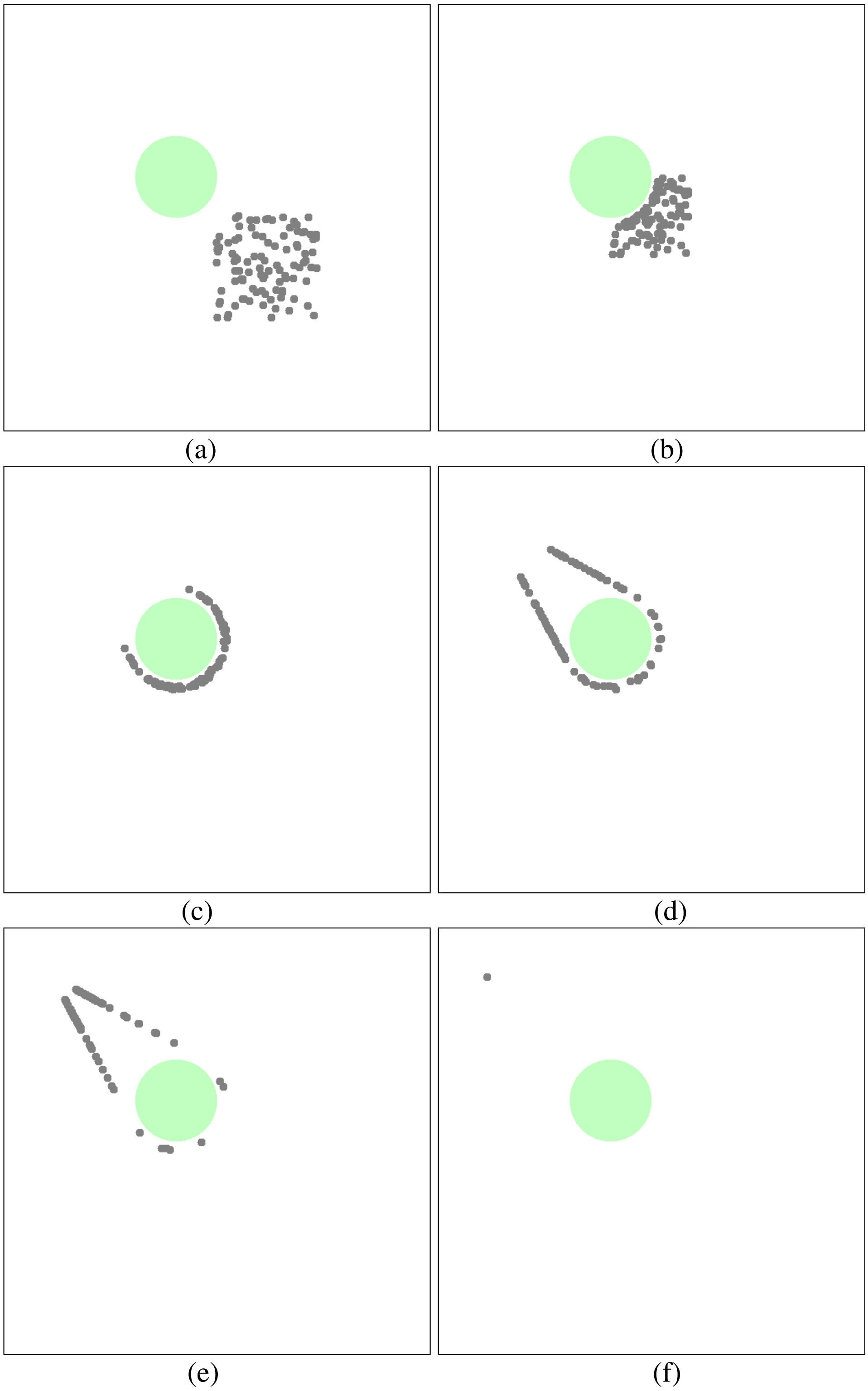
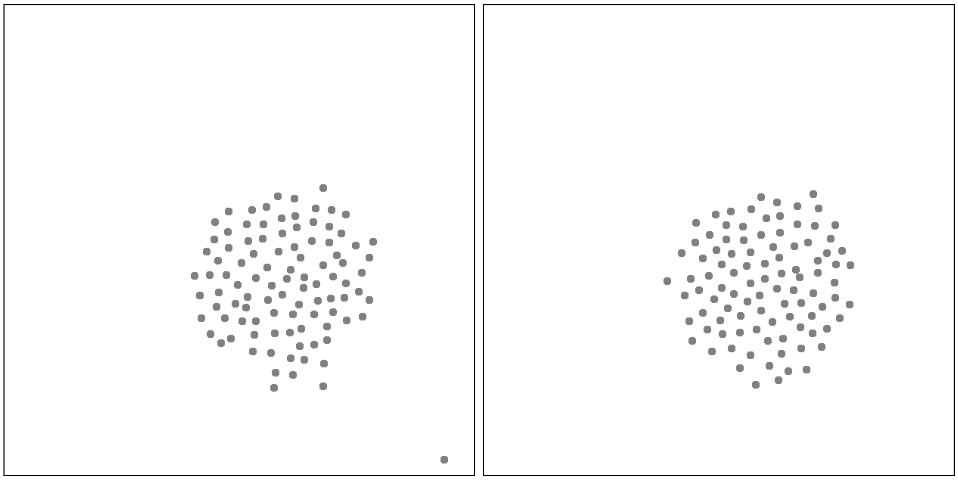


Figure 5.9: Solving consensus problem in presence of circular obstacle: (a), (b), (c), (d),(e),(f) depict the positions of the robot during simulation.



(a) (b)

Figure 5.10: Avoiding Neighboring robots: (a), (b) depict the positions of the robot during simulation.

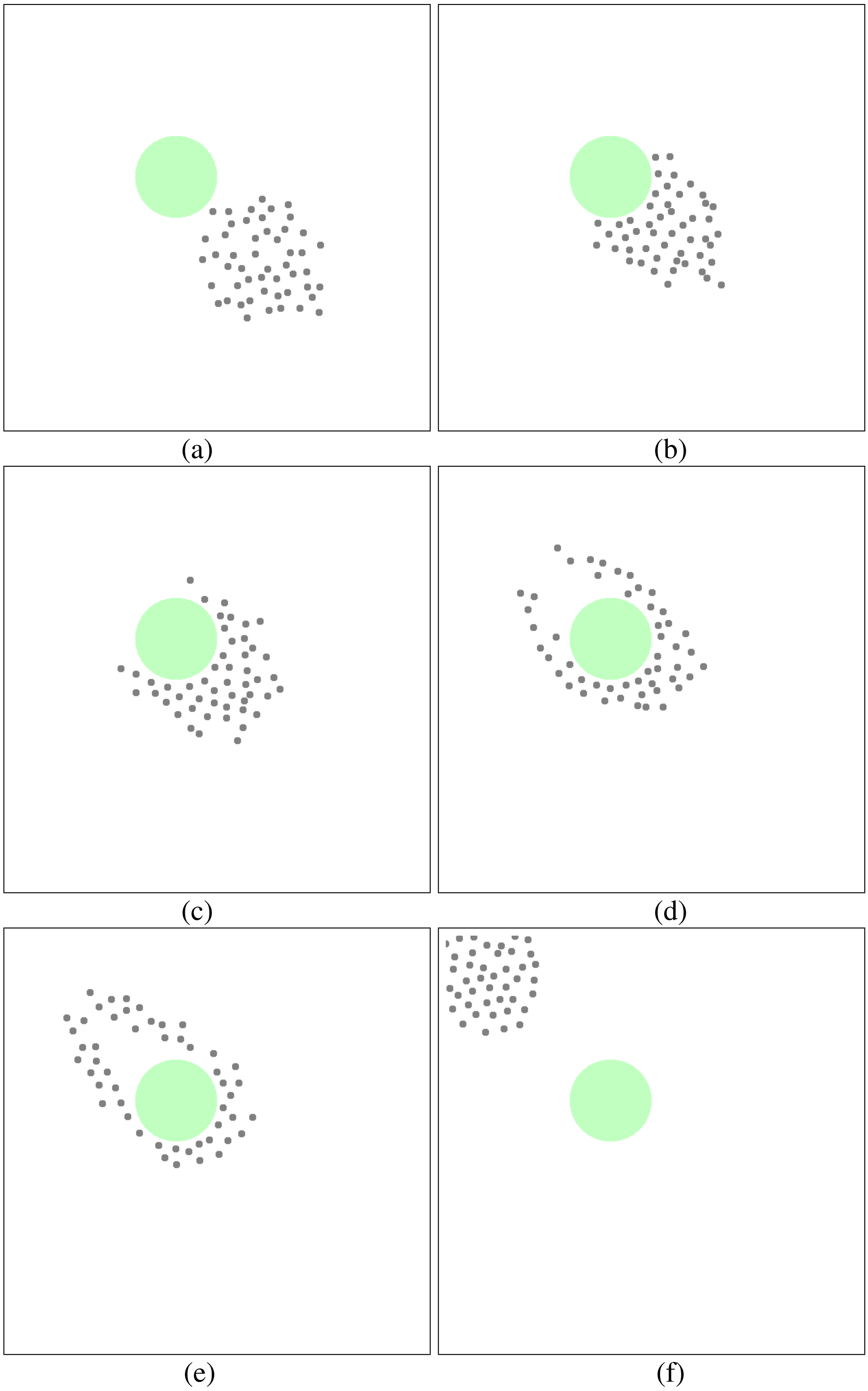
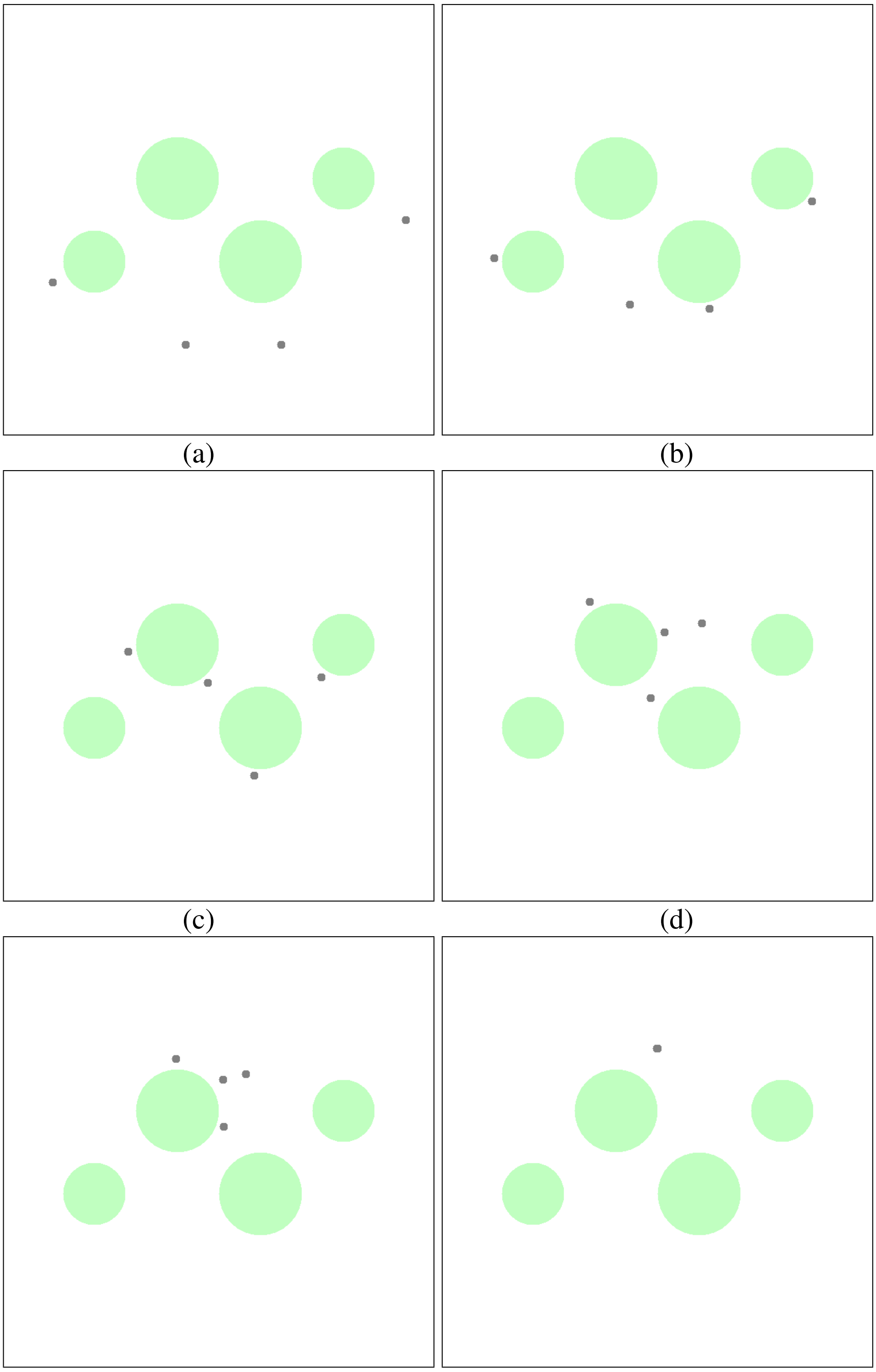
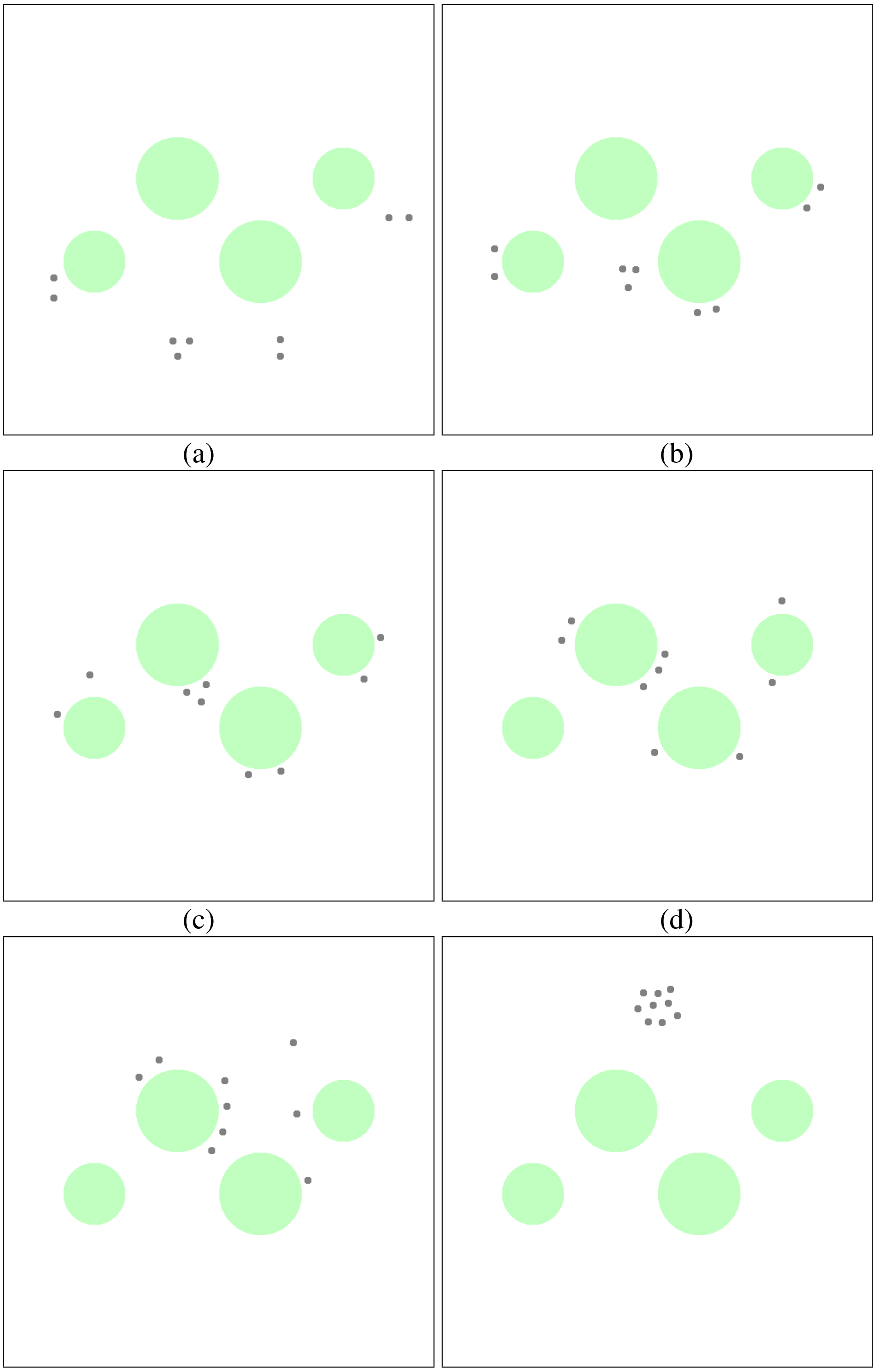


Figure 5.11: Conventional control strategy [24] in presence of circular Obstacle: (a), (b), (c), (d),(e),(f) depict the positions of the robot during simulation.



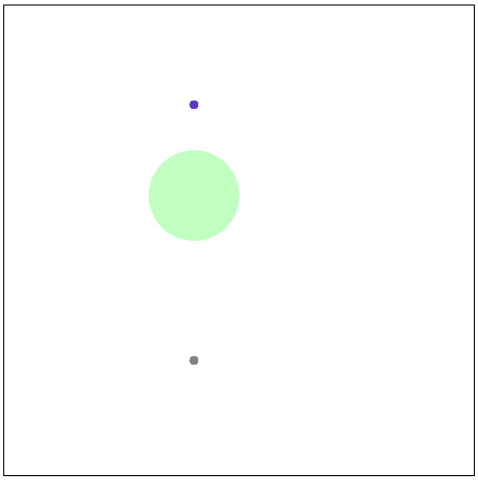
(a) (b) (c) (d) (e) (f)

Figure 5.12: Solving consensus problem in presence of Multiple circular obstacles: (a), (b), (c), (d), (e),(f) depict the positions of the robot during simulation.

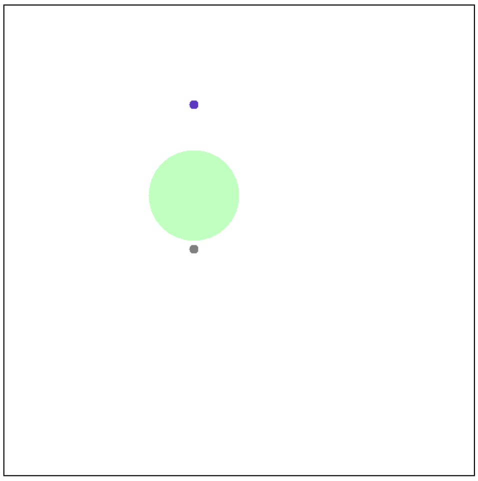


(a) (b) (c) (d) (e) (f)

Figure 5.13: Conventional control strategy [24] in presence of multiple circular Obstacles: (a), (b), (c), (d), (e), (f) depict the positions of the robot during simulation.



(a)



(b)

Figure 5.14: Failure of single mobile robot to reach the goal in presence of circular shaped obstacle: (a), (b) depict the positions of the robot during simulation.

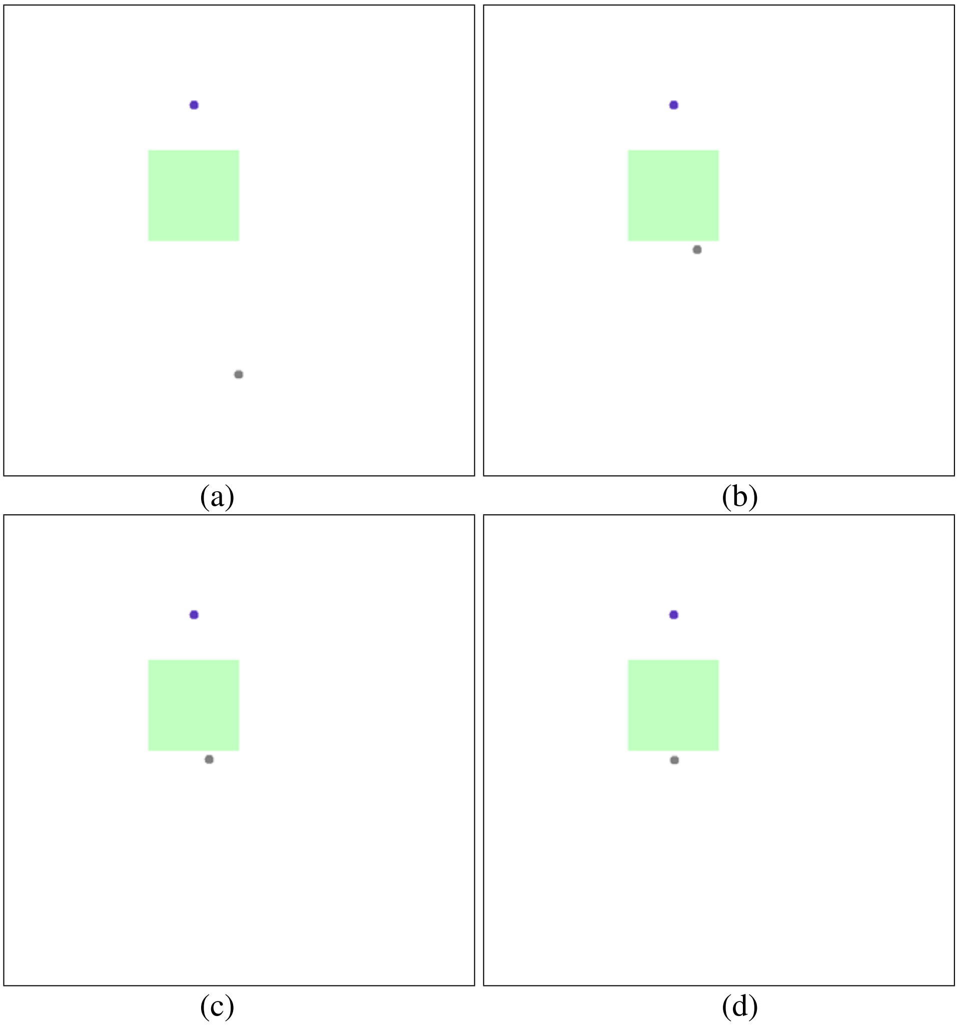


Figure 5.15: Failure of single mobile robot to reach the goal in presence of a square shaped obstacle: (a), (b), (c), (d) depict the positions of the robot during simulation.

5.2.3 Comparative analysis

Control strategy, proposed in section 4.3 has been compared with the one proposed in [24]. To compare the control strategies, simulation parameters depicted in table 5.6 are used with 15 robots initialized at known position coordinates, depicted in table 5.8, with two circular and two square shaped obstacles. Snapshots of the obtained results are por-

Table 5.7: Salient features of the proposed control strategy in comparison to existing

Control strategy [24]	**Proposed control strategy**
No collision with neighboring robots	No collision with neighboring robots
No collision with any obstacle	No collision with any obstacle
Avoid circular obstacles	Avoid circular obstacles
No guarantee of avoiding-rectangular obstacles	Guaranteed avoidance of-rectangular obstacles

trayed in fig. 5.20 and 5.21. It can be observed from the obtained results that for both control strategies, the mobile robots do not collide among themselves or with any obstacle. The robots successfully avoid circular obstacles but main difference between two control strategies is observed when robot encounters a square shaped obstacle. It can be observed that four robots, controlled using the strategy developed [24] get struck and are unable to reach near Rendezvous point but the proposed control strategy ensures each and every robot of MRS flock in neighborhood of Rendezvous point. If robot abides to the condition laid down in **Proposition 1** at all points in its trajectory towards the goal $[40\ 90]^T$ then it is unable to avoid obstacle and gets struck, if it's dynamics are controlled using control strategy formulated in [24]. By employing control strategy described in section 4.3, robot follows walls of the square or circular obstacle while avoiding collision with it's neighbors and flock about the Rendezvous point. The salient features of the proposed control strategy are summarized in table 5.7.

Table 5.8: Initial Position of Robots

Robot	Position
$\mathbf{x}_{01}$	$[90\ 15]^T$
$\mathbf{x}_{02}$	$[90\ 17]^T$
$\mathbf{x}_{03}$	$[92\ 15]^T$
$\mathbf{x}_{04}$	$[92\ 17]^T$
$\mathbf{x}_{05}$	$[35\ 15]^T$
$\mathbf{x}_{06}$	$[37\ 15]^T$
$\mathbf{x}_{07}$	$[35\ 17]^T$
$\mathbf{x}_{08}$	$[37\ 17]^T$
$\mathbf{x}_{09}$	$[10\ 10]^T$
$\mathbf{x}_{10}$	$[10\ 12]^T$
$\mathbf{x}_{11}$	$[12\ 12]^T$
$\mathbf{x}_{12}$	$[02\ 75]^T$
$\mathbf{x}_{13}$	$[04\ 75]^T$
$\mathbf{x}_{14}$	$[02\ 77]^T$
$\mathbf{x}_{15}$	$[04\ 77]^T$

Figure 5.16: Failure of MRS to reach the rendezvous point in presence of square shaped obstacle: (a), (b), (c), (d), (e), (f), (g),(h)depict the positions of the robot during simulation.

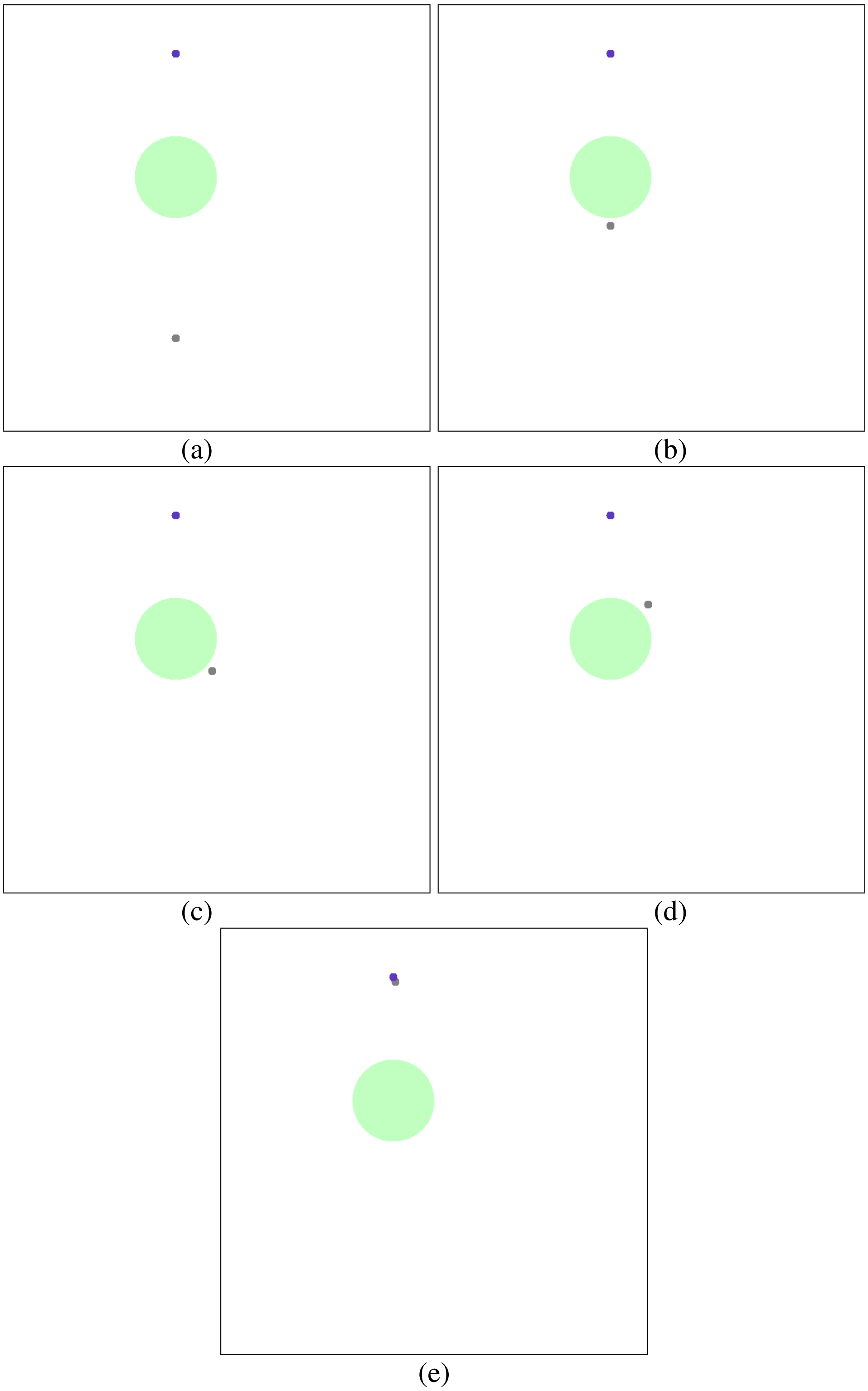


Figure 5.17: Failure of single mobile robot to reach the goal in presence of a square shaped obstacle: (a), (b), (c), (d),(e) depict the positions of the robot during simulation.

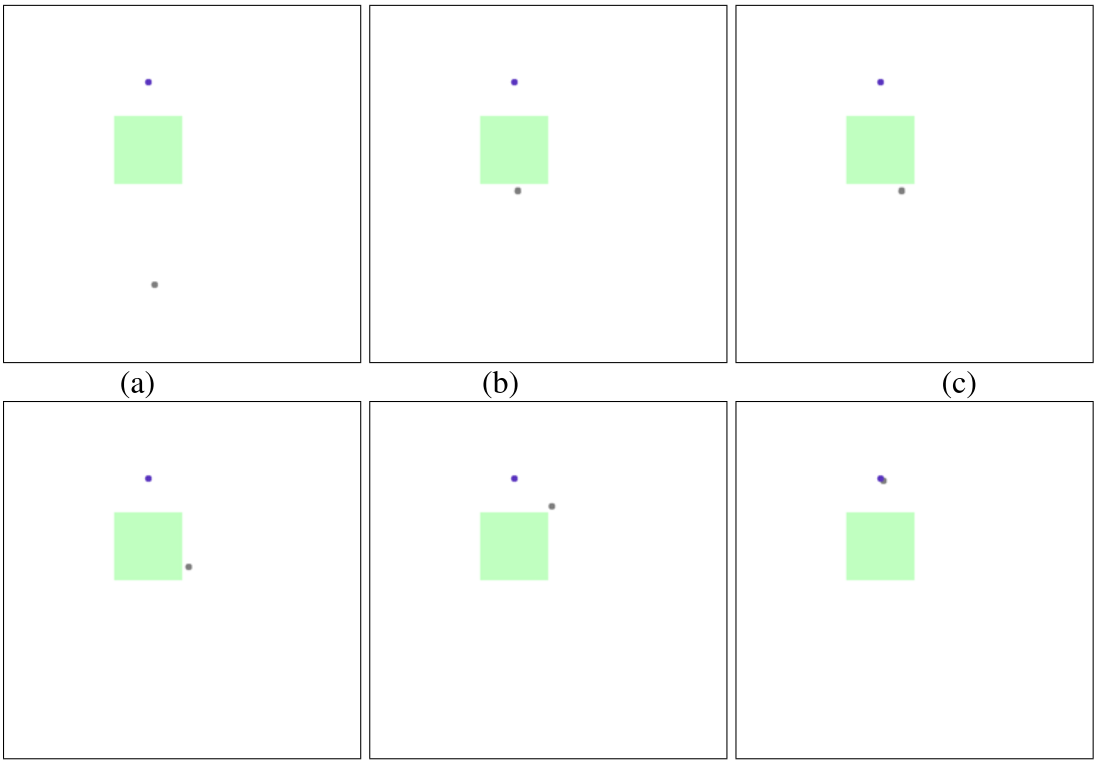

(a) (b) (c)
(d) (e) (f)

Figure 5.18: Simulation of follow wall behavior, for single mobile robot, to avoid the square shaped obstacle: (a), (b), (c), (d), (e), (f) depict the positions of the robot during simulation.

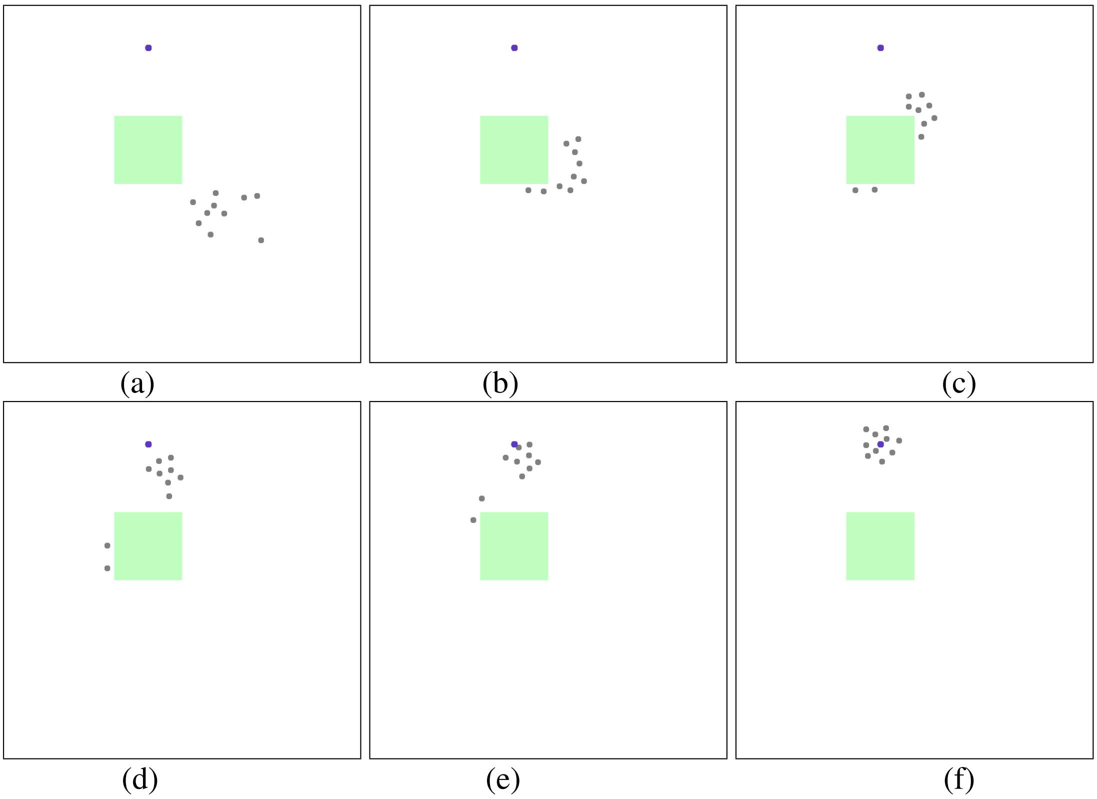


Figure 5.19: Simulation of the proposed control strategy to reach Rendezvous point in presence of square shaped obstacle: (a), (b), (c), (d), (e), (f) depict the positions of the robots constituting the MRS during simulation.

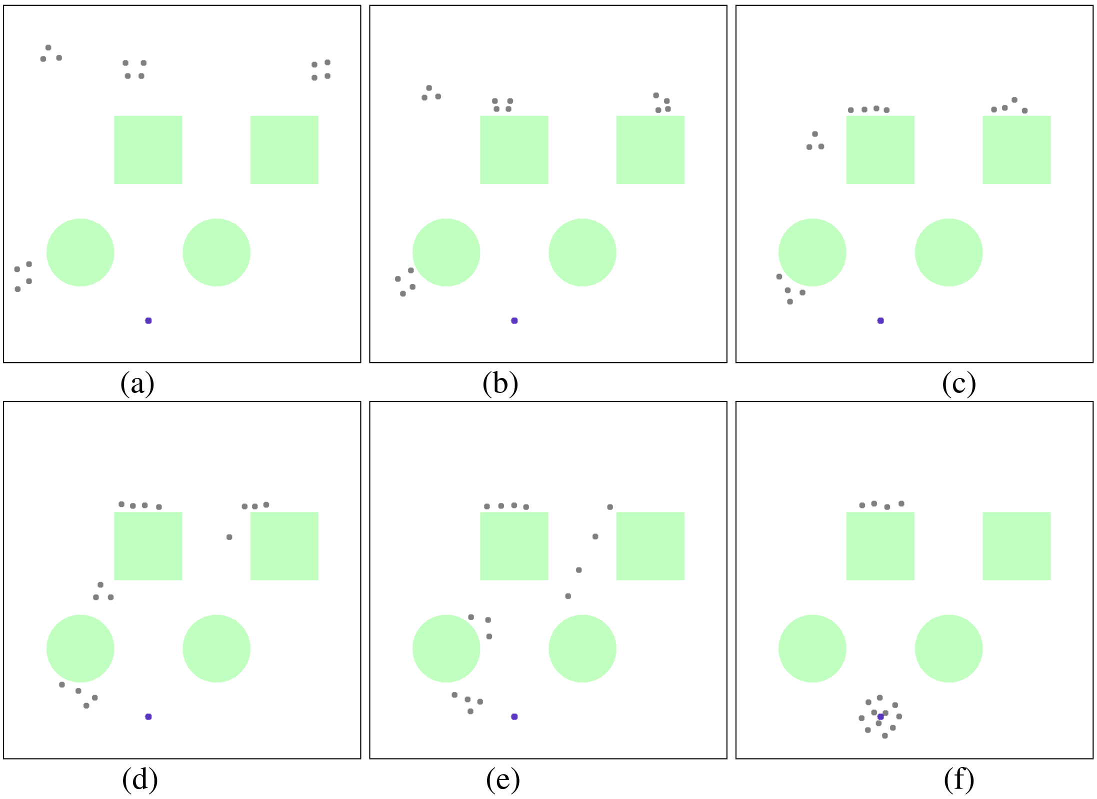


Figure 5.20: Navigation of multiple robots based on control strategy developed in (a), (b), (c), (d), (e), (f) depict the positions of the robots constituting the MRS during simulation.

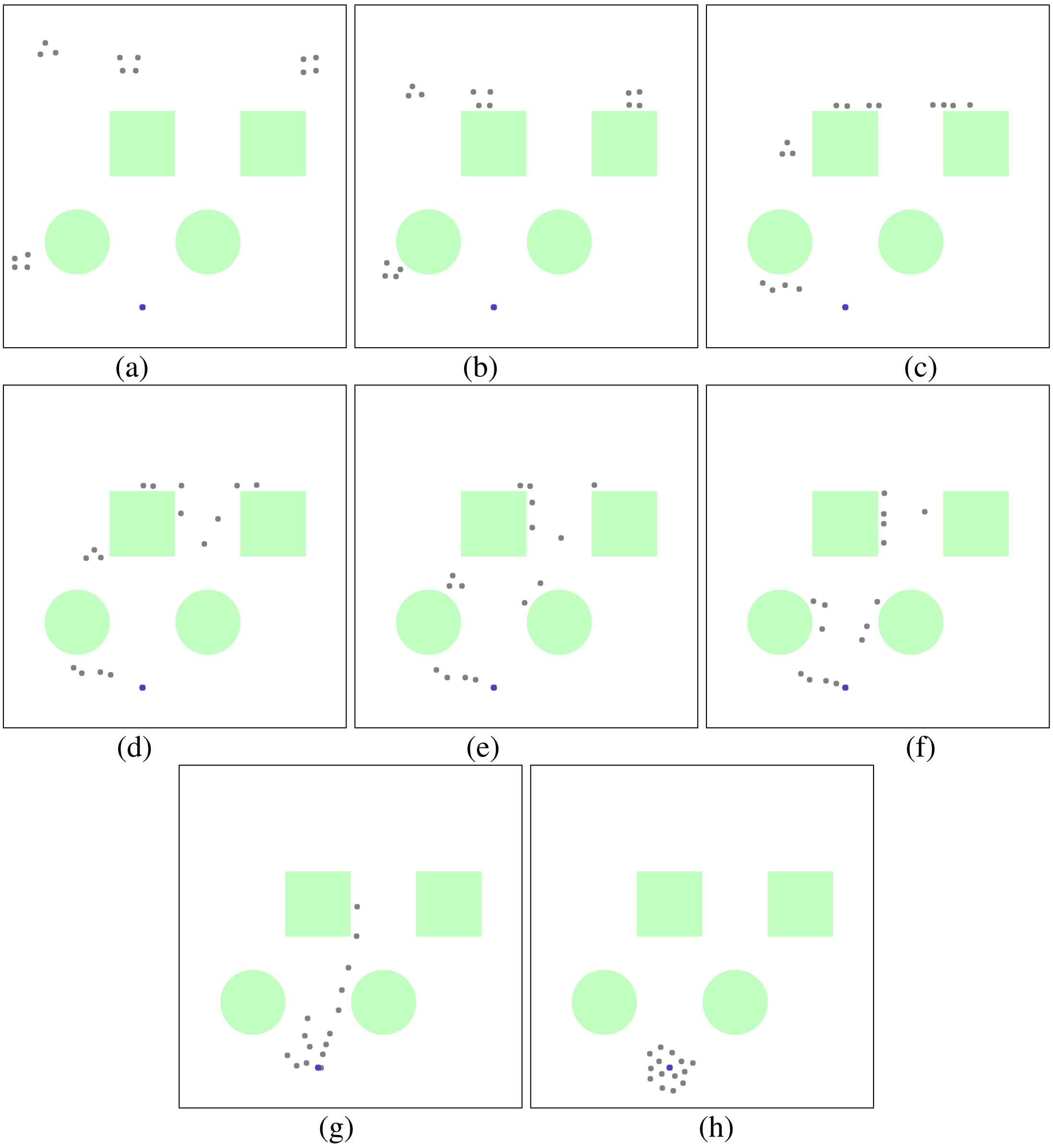


Figure 5.21: Simulation of the proposed control strategy to reach Rendezvous point in presence of square shaped obstacle: (a), (b), (c), (d), (e), (f), (g), (h) depict the positions of mobile robots of MRS during the simulation.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

Combining GTG and OA behaviors, in NSB framework, in presence of rectangular obstacles does not guarantee accomplishment of navigation objective of reaching the goal as the robot exhibits type-1 Zeno phenomenon, which has been proved both mathematically and by means of simulations. Proposed control strategy, modified NSB approach, based on *Follow Wall behavior*, was found to be a promising solution for avoiding PCC and reaching the Rendezvous point, while avoiding collisions with intermittent rectangular or circular obstacles and neighboring robots. The results obtained though simulations testify the formulated strategy and proposition.

6.2 Future Work

While executing this project, another theory which claims the failure of combining GTG and OA behaviors in NSB framework, in presence of concave obstacles was intuitively formulated. The claim was simulated using MASON library and found that the robot is struck along a line where the OA and GTG behaviors are in complete confliction of each other but unfortunately this dissertation is not able to provide a concrete proof of it. The simulation results for a single mobile robot is shown in fig. 6.1. For a MRS, simulation results are depicted in fig. 6.2. The proposed control strategy, presented in section 4.3 can be exploited in this case too; snapshots of the simulation for a single robot and MRS are shown in fig. 6.3 and 6.4. We, think that this problem could be addressed in the future.

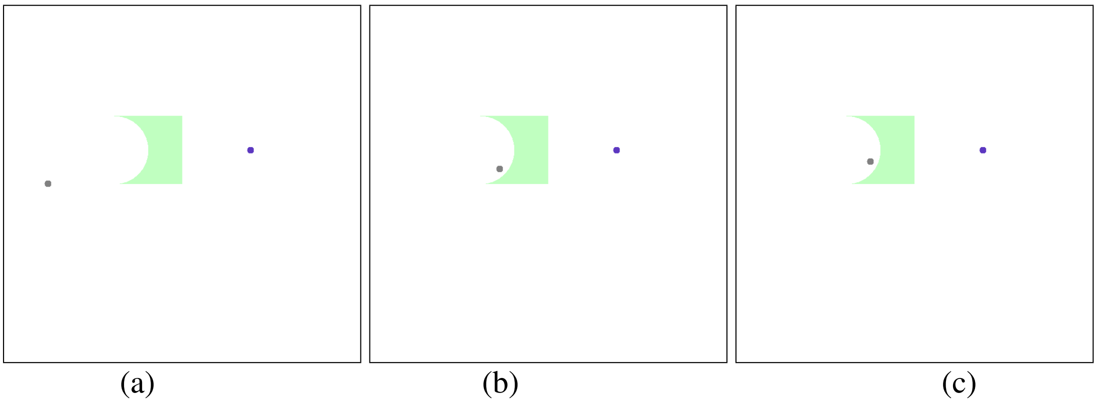


Figure 6.1: Simulation of the conventional NSB control strategy to reach the goal in presence of concave obstacle: (a), (b), (c) depict the positions of mobile robot during the simulation.

6.3 Publication

Aman Sharma, Kurapati Naga Ganesh, Bolloju Pranay, Tejkaran Charan, Matam Manjunath, Ravi Prasad K. Jagannath, Patrice Wira, A.V. Narasimhadhan, "A Modified Null Space Based Strategy to Avoid Straight-Edged Obstacles and Solve Consensus Problem for Multi Robot System.", *Unpublished.*

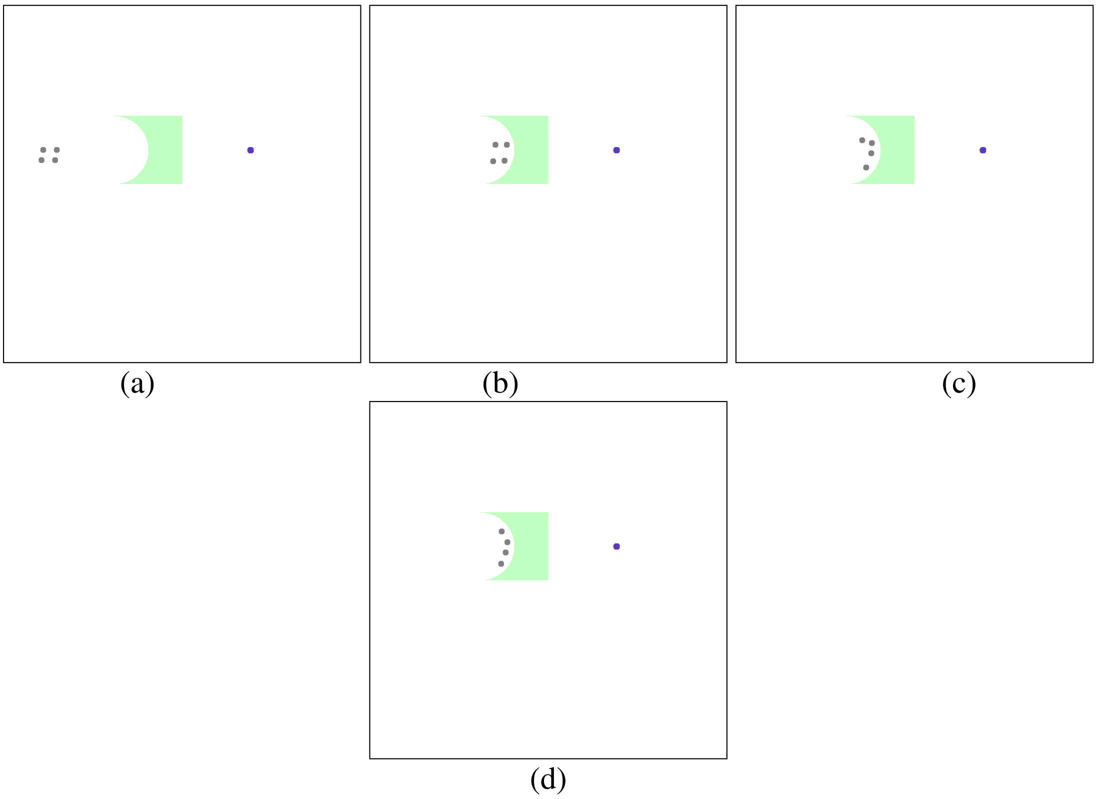


Figure 6.2: Simulation of the conventional NSB control strategy to reach Rendezvous point in presence of concave obstacle: (a), (b), (c), (d) depict the positions of mobile robots of MRS during the simulation.

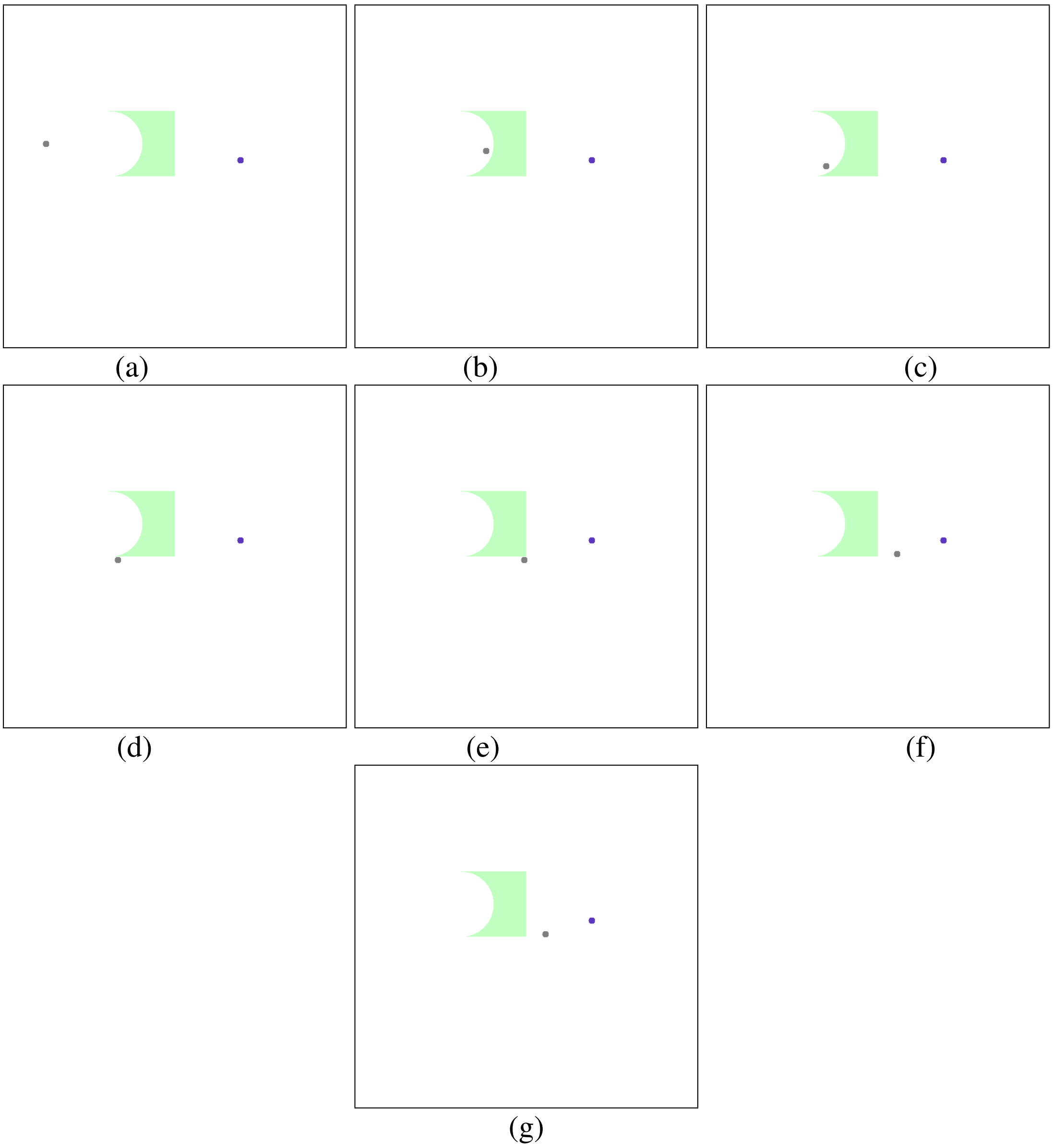


Figure 6.3: Simulation of FW behavior to reach the goal in presence of concave obstacle: (a), (b), (c), (d), (e), (f), (g) depict the positions of mobile robot during the simulation.

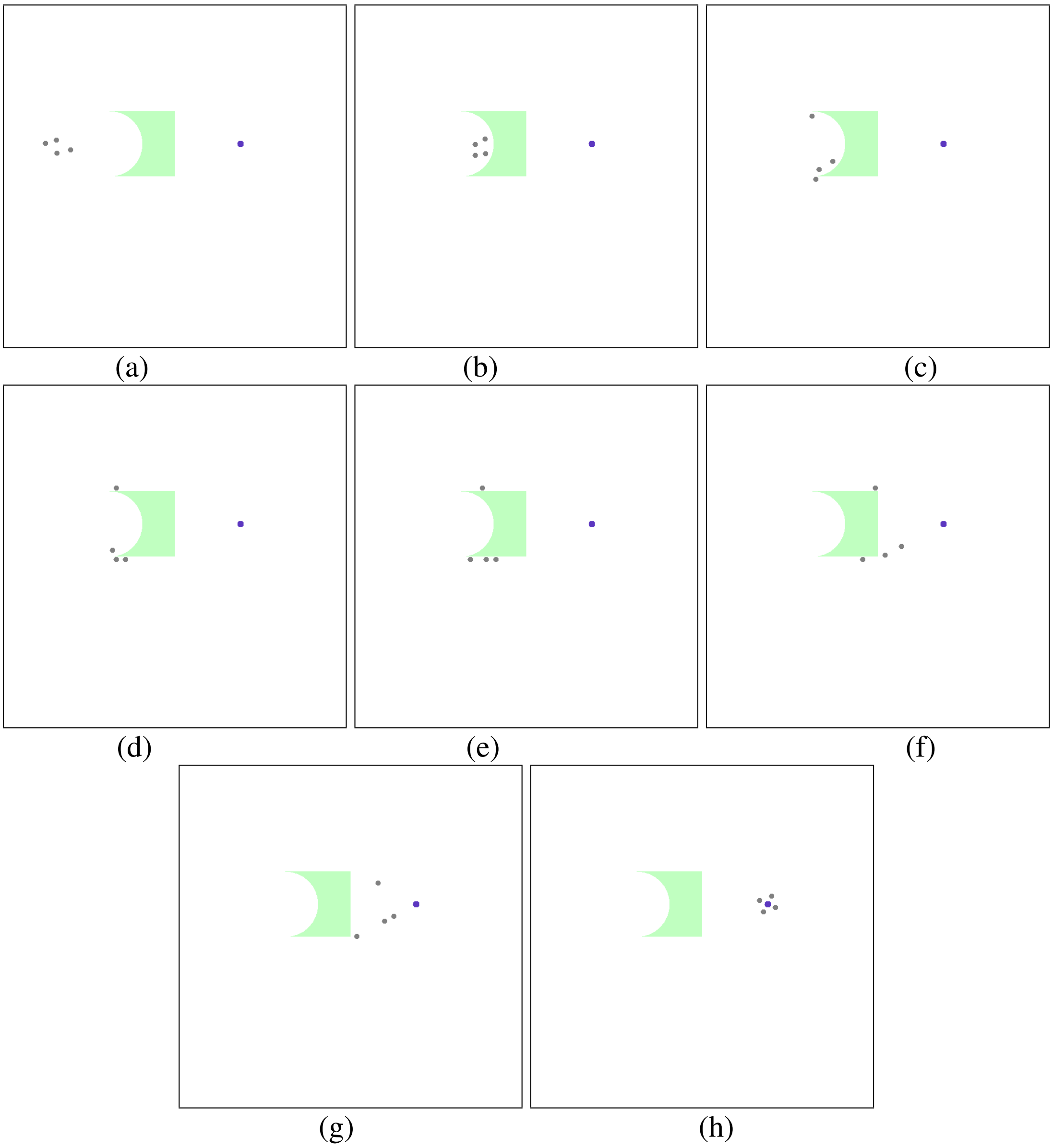


Figure 6.4: Simulation of the proposed control strategy to reach Rendezvous point in presence of concave obstacle: (a), (b), (c), (d), (e), (f), (g),(h) depict the positions of mobile robots of MRS during the simulation.

REFERENCES

[1] Gaurav S. Sukhatme, James F. Montgomery and Richard T. Vaughan., "Experiments with Cooperative Aerial-Ground Robots.", *In Robot Teams: From Diversity to Polymorphism, AK Peters.*, pp. 345–367, 2001.

[2] Reid Simmons, Sanjiv Singh, David Hershberger, Josue Ramos and Trey Smith., "First Results in the Coordination of Heterogeneous Robots for Large-scale Assembly.", *In Proceedings of the International Symposium on Experimental Robotics (ISER).*, 2000.

[3] Milutinovic D., Lima P., "Modeling and Optimal Centralized Control of a Large-Size Robotic Population.", *Robotics, IEEE Transactions on.*, vol. 22, no. 6, pp. 1280–1285, Dec. 2006.

[4] Khoshnevis B., Bekey G., "Centralized Sensing and Control of Multiple Mobile Robots.", *In* 22^{nd} *ICC&IE Conference.*, pp. 503–506, 1998.

[5] Baldassarre G., Trianni Vito, Bonani M., Mondada F., Dorigo M., Nolfi S., "Self-Organized Coordinated Motion in Groups of Physically Connected Robots.", *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on.*, vol. 37, no. 1, pp. 224–239, Feb. 2007.

[6] Maja J Mataric., "Issues and Approaches in Design of Collective Autonomous Agents.", *Robotics and Autonomous Systems.*, vol. 16, no. 2–4, pp. 321-331, Dec. 1995.

[7] J. McLurkin. "Stupid robot tricks: Behavior-based distributed algorithm library for programming swarms of robots.", *Master's thesis, Massachusetts Institute of Technology, Cambridge.*, 2004.

[8] Butler Z.J., Rizzi A.A., Hollis R.L., "Cooperative Coverage of Rectilinear Environments.", *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on.*, vol. 3, pp. 2722–2727, 2000.

[9] Israel A. Wagner, Michael Lindenbaum and Alfred M. Bruckstein. "MAC vs. PC: Determinism and Randomness as Complementary Approaches to Robotic Exploration of Continuous Unknown Domains.", *The International Journal of Robotics Research.*, vol. 19, no. 1, pp. 12–31, 2000.

[10] Craig W. Reynolds., "Flocks, herds, and schools: A distributed behavioral model.", *Proceeding SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques.*, pp. 25–34, 1987.

[11] L. Iocchi, D. Nardi, and M. Salerno., "Reactivity and deliberation: A survey on multi-robot systems.", *In Balancing Reactivity and Social Deliberation in Multi-Agent Systems, From RoboCup to Real-World Applications (selected papers from the ECAI 2000 Workshop and additional contributions), Springer-Verlag.*, pp. 9–34, London, UK, 2001.

[12] S. Carpin and L. E. Parker., "Cooperative motion coordination amidst dynamic obstacles.", *Distributed Autonomous Robotic Systems.*, pp. 145–154, 2002.

[13] M.J. Mataric., "Behavior-based control: Examples from navigation, learning, and group behavior.", *Journal of Experimental and Theoretical Artificial Intelligence.*, vol. 9 no. 2–3, pp. 323–336, 1997.

[14] S. Premvuti and S. Yuta., "Consideration on the cooperation of multiple autonomous mobile robots.", *Intelligent Robots and Systems '90. 'Towards a New Frontier of Applications', Proceedings. IROS '90. IEEE International Workshop on.*, vol. 1, pp. 59–63, 1990.

[15] C. F. Touzet., "Robot awareness in cooperative mobile robot learning.", *Auton. Robots.*, vol. 8, no. 1, pp. 87âĂŞ97, 2000.

[16] Murphy R.R., Christine Lisetti, Russ Tardif, Liam Irish, Aaron Gage., "Emotion-based control of cooperating heterogeneous mobile robots.", *Robotics and Automation, IEEE Transactions on.*, vol. 18, no. 5, pp. 744–757, 2002.

[17] D. Jung and A. Zelinsky., "Grounded symbolic communication between heterogeneous cooperating robots.", *Autonomous Robots.*, vol. 8, pp. 269–292, 2000.

[18] Kitano H., "RoboCup Rescue: a grand challenge for multi-agent systems.", *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on.*, pp. 5–12, 2001.

[19] B.E. Bishop. "On the use of redundant manipulator techniques for control of platoons of cooperating robotic vehicles.", *IEEE Transactions on Systems, Man and Cybernetics.*, vol. 33, no. 5, pp. 608–615, Sept. 2003.

[20] A.A. Maciejewski. "Numerical filtering for the operation of robotic manipulators through kinematically singular configurations.", *Journal of Robotic Systems*, vol. 5, no. 6, pp. 527–552, 1998.

[21] Y. Nakamura, H. Hanafusa, and T. Yoshikawa. "Task-priority based redundancy control of robot manipulators.", *The International Journal Robotics Research.*, vol. 6, no. 2, pp. 3–15, 1987.

[22] Chiaverini, S., “Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators,” *Robotics and Automation, IEEE Transactions on* , vol.13, no.3, pp.398–410, Jun 1997.

[23] Antonelli, G.; Arrichiello, F.; Chiaverini, S., “The null-space-based behavioral control for mobile robots”, *Computational Intelligence in*

Robotics and Automation, 2005. CIRA 2005. Proceedings. 2005 IEEE International Symposium on., pp.15–20, 27–30 June 2005

[24] Gianluca Antonelli, Filippo Arrichiello, Stefano Chiaverini, "Flocking for multi-robot systems via the Null-Space-based Behavioral control.", *Swarm Intell.*, vol. 4, pp. 37–56, 2010.

[25] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan, "A Multi-Agent Simulation Environment.", *In Simulation: Transactions of the society for Modeling and Simulation International.*, vol.82, no.7, pp.517–527, 2005.

[26] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, and Keith Sullivan. "A New Multi- Agent Simulation Toolkit.", *Proceedings of the 2004 SwarmFest Workshop.*, 2004.

[27] Sean Luke, Gabriel Catalin Balan, Liviu Panait, Claudio Cioffi-Revilla, and Sean Paus. "A Java Multi-Agent Simulation Library 2003.", *Proceedings of the Agent 2003 Confer- ence.*, 2003.

[28] Sean Luke, Keith Sullivan, Gabriel Catalin Balan, and Liviu Panait., "Tunably Decen- tralized Algorithms for Cooperative Target Observation.", *Technical Report GMU-CS-TR- 2004-1, Department of Computer Science, George Mason University.*, 2004.

[29] Liviu Panait and Sean Luke., "Ant Foraging Revisited", *Proceedings of the Ninth In- ternational Conference on the Simulation and Synthesis of Living Systems (ALIFE9).*, 2004.

[30] Liviu Panait and Sean Luke., "Learning Foraging Behaviors.", *Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALIFE9)*, 2004.