Collaborative Optimal Reciprocal Collision Avoidance (CORCA)

By

Shehryar Ali Khan

NUST201260408MSMME62112F

Supervised by

Dr. Yasar Ayaz

A Report Submitted to the
Department of Robotics and Artificial Intelligence
in Partial Fulfillment of the Requirements for the Degree of
Masters in
ROBOTICS AND INTELLIGENT MACHINE ENGINEERING

School of Mechanical and Manufacturing Engineering (SMME) at
National University of Sciences and Technology
(NUST),Islamabad, Pakistan

August, 2016

Copyright©2015 by NUST

All rights reserved. Reproduction in whole, in part or in any form requires the prior written permission of Mr. Shehryar Ali Khan or designated representative.

Dedicated to my parents who deserve the credit to all my success.

Certificate of Approval

It is certified that the Thesis work titled "Collaborative Optimal Reciprocal Collision Avoidance (CORCA)", is carried out by **Shehryar Ali Khan** under the supervision of **Dr. Yasar Ayaz**, at National University of Sciences and Technology (NUST), Islamabad. It is fully adequate, in scope and in quality, as a thesis for the degree of Masters in Robotics and Intelligent Machine Engineering.

Thesis Advisor
Dr. Yasar Ayaz
Associate Professor NUST, Islamabad

Examiner
Dr. Umar Ansari
Asst Professor NUST, Islamabad

Examiner
Dr. Muhammad Naveed
Asst Professor NUST, Islamabad

Examiner
Sara Babar
Lecturer NUST, Islamabad

Head of Department (R and AI)
Dr.Yasar Ayaz

Pricipal (SMME)
Dr.Abdul Ghafoor

Declaration

It is declared that the work entitled "Collaborative Optimal Reciprocal Collision Avoidance (CORCA)" presented in this Thesis is an original piece of my own work, except where otherwise acknowledged in text and references. This work has not been submitted in any form for another degree or diploma at any university or other institution for tertiary education and shall not be submitted by me in future for obtaining any degree from this or any other University or Institution.

Shehryar Ali Khan
NUST201260408MSMME62112F

Acknowledgments

Especial thanks to all the faculty of SMME in general and Robotics Department in particular, who have guided me whenever needed, and without whom my research would not have been possible.

I am also thankful to my supervisor Dr. Yasar Ayaz, who took special interest in my work and guided me towards success.

I also thank my colleagues who helped me in understanding and troubleshooting the problems I faced, especially Habib Ahmed, Umair Iqbal, Usama Siraj, Faizan Ahmed and Huzaifa Qamar.

Abstract

Mobile robots are the future of modern day vehicles and their collision is one of the major concerns and challenges which is a very active field of research these days. As the number of vehicles increasing, the complexity and challenges are also growing exponentially. There is an imminent need of a strategy or algorithm to mitigate their problems. In that regard, the research presented in this thesis is one of possible solutions that eliminates the drawback of existing collision avoidance algorithms known as *Deadlock*. The proposed algorithm is named as Velocity-based Collaborative Optimal Reciprocal Collision Avoidance (V-CORCA) which is extensively analyzed using holonomic robots in a computer simulation. A brief comparison among other versions of the same techniques such as Optimal Reciprocal Collision Avoidance (ORCA) and Collaborative Optimal Reciprocal Collision Avoidance (CORCA), is also presented for performance evaluation. The simulation results reveal that unlike ORCA, V-CORCA was able to prevent deadlocks, while at the same time, reducing the overall halt time for single and multiple robots within an eight robots-based multi-robot system. By analyzing the simulation results of a single and (eight robots-based) multi-robot system , it is concluded that V-CORCA is not only efficient in reducing the over all halt time but also eradicates the possibility of occurring a *Deadlock*.

Keywords: Collision avoidance, mobile robots, multi-robot systems, ORCA, CORCA.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

With advent of industrialization, automobiles have become as essential part of daily life. With the dramatical increase of populations, the number automobiles have also increased drastically, so their problems along with them. Automobiles are usually driven by humans who take decisions over the course to guide the vehicle to its destination. Driving a vehicle is one of the cumbersome jobs that everyone encounters in his daily life. As the number of vehicles increased exponentially, the probability of occurring an accident have increased beyond dangerous levels.

Numerous methods have been adopted to avoid the accidents such as traffic rules and regulations, road guides and adequate road width for vehicles to maneuver but these are not panacea. These measures are effective for a short time. Almost every vehicle is driven by human, therefore, most of the accidents happen due to human errors instead of uncertain electrical or mechanical breakdown of vehicles.

All these issues give rise to an inevitable need of robots taking control of the vehicles, eliminating the human errors at all. Robots are less likely to make mistakes as they are programmed to do a specific task using a sophisticated algorithms. With introduction of flying cars [5], the situation have even more complicate and there is an utmost need of unmanned vehicles to avoid collision/accidents. Collision avoidance system is one of fundamental key safety feature of modern day automobiles.

Nevertheless, there are unmanned vehicles which used to be considered as futuristic, are now coming forth in real life. A lot of research is being conducted to on numerous problems such as collision avoidance [6] [7] [8], vehicle's aerodynamics [9] [10], flying cars [11] etc., all these to meet the fundamental need of transportation in a person's daily life and to develop sophisticated robots for defence and warlike situation for combat and rescue missions.

The manned vehicles in which the robots take control and drive the vehicle to its destination, is the ultimate solution to avoid collision in ever growing complexity of maneuvering. Robots follow a set rules defined the algorithm that governs its whole motions and enables the robots to take decision where it is necessary in order to successfully achieve the target or reach the destination. Before diving deep into underlying strategies and algorithms, there are few terms that are needed to be familiar with.

1.1 Collision Avoidance System

A system of control strategies and algorithm that senses the environment and based on sensed data, develops an approach or road map to maneuver it to its destination without any collision or accident. This is broad concept or more generalized term to represent a system that detects and prevents a collision.

1.2 Reciprocal Collision Avoidance System

A system in which different moving entities or robots make decision to avoid collision and to navigate through a common plane without communicating with each other.

1.3 Optimal Reciprocal Collision Avoidance System

A system in which different moving entities or robots develops an optimal road map or path to their destination and on their course, if then encounter with other moving or stationary entities, they make an optimal decision without communicating with each other in order to avoid collision. This technique is far superior in achieving the goal and complexity as compared to Reciprocal Collision Avoidance technique.

As the autonomous robots become more and more popular, the need of an algorithm that not only prevents the robots from colliding but also makes sure that all the robots reach the destination in a finite time even if they are sharing the same limited space to maneuver. There are countless applications and advantages of multi-agent systems such as monitoring, detection, leading rescue or recovery missions, structural assembly and military surveillance [12] [13] [14] etc. which have made them very popular and sound choice for deployment. If there fewer robots located at far distance in a shared space, the probability of occurring a collision can be eradicated by deploying global path planing techniques.

Chapter 2
Literature Survey

In this chapter, the different collision avoidance techniques such as ORCA, CORCA and Velocity-based CORCA (V-CORCA) are described briefly. The Optimal Reciprocal Collision Avoidance (ORCA) is the fundamental technique from which other collision avoidance techniques are derived. The derivation of advanced techniques is presented for better understanding the abstract of this research. Keeping in view the vital theoretical details, the performance of these algorithm can be assessed.

ORCA is founded on velocity-based algorithm, which changes the velocity of robot if there is probable chance of collision, instead a new velocity is adopted which will not only avoid collision but also keeps the robot near to the optimum velocity for reaching its target in time [15].

While Collaborative Optimal Reciprocal Collision Avoidance (CORCA) utilizes the information of instantaneous position of the robot for collision prediction and avoidance by maintaining the fundamental laws of ORCA. Likewise, V-CORCA utilizes the velocity information of the robot for collision prediction and prevention. It is of prime importance that the vital point of this discussion to eliminate the limitation of ORCA i.e. *Deadlock* [15].

2.1 Optimal Reciprocal Collision Avoidance (ORCA)

ORCA was first designed to overcome the problem of reciprocal dances, however in [16], this issue was also resolved by other techniques known as Reciprocal Velocity Obstacle (HRVO). It happens when two robots that are likely to collide take measures to avoid collision but the attempts made, are in the same direction as described by the algorithm to be the preferred side to maneuver. It not only makes the robot to take a longer path but also leaves it astray from its proposed trajectories based on the velocities [17].

Let assume a scenario in which the plane is disk-shaped and there are k holonomic robots which have radius r_k, position and velocity of p_k in plane $\Re^2$ and $v_k \in \Re^2$

respectively. It is the desire of each robot, determined by the algorithm, to reach its goal g_k by optimally choosing a velocity v_k in order to avoid collision which should also approximate to the proposed velocities $v_k^{prefer} \in \Re^2$ in a finite time τ. For better understanding, an exampling of two robots is illustrated in Figure 2.1 where r_A, r_B are their radii, p_A, p_B are their positions and v_A, v_B are their velocities respectively.

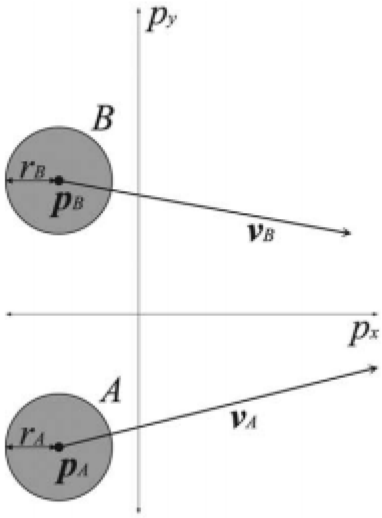

Fig. 2.1: Two Robots moving with v_A and v_B

The relative velocities and the distance between two robots are computed which yields a cone of collision space. Any obstacle within this cone is subject to collision. This phenomena is illustrated in Figure 2.2. The main objective of ORCA is to find a possible safe velocity that could be adopted by the robot in order to avoid collision from a dynamic or stationary objects and it also assumed that the velocity of robot during collision avoidance time window τ remains constant.

This is achieved by calculating the half-plane velocities of each object that has the potential to collide. For a perfect collision free performance, the information of dimensions, position and velocities of different entities present in the surroundings should be well known in order to determine an appropriate course of action [18].

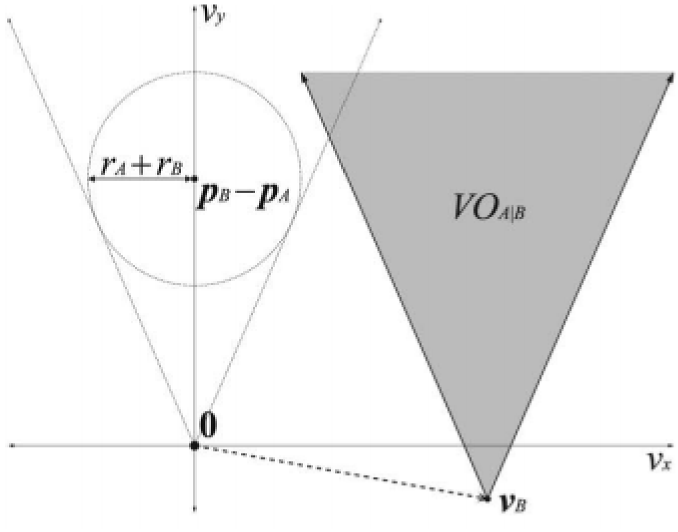

Fig. 2.2: Two Robots moving with v_A and v_B

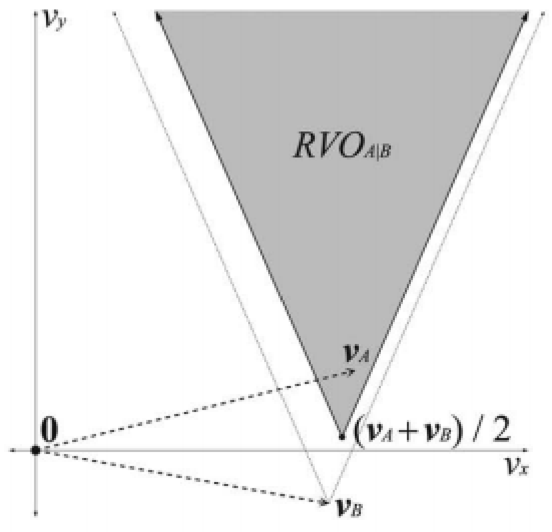

Fig. 2.3: Two Robots moving with v_A and v_B

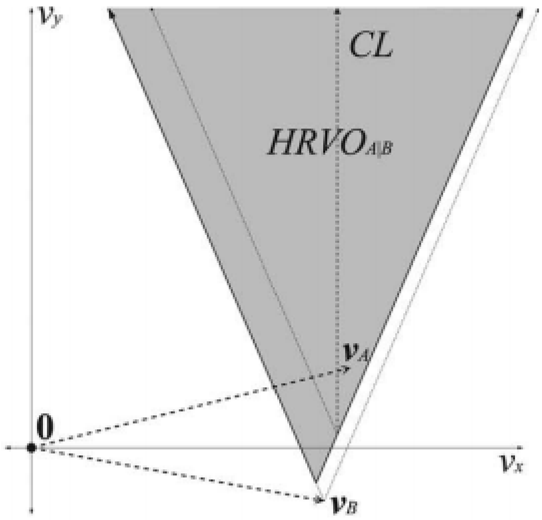

Fig. 2.4: Two Robots moving with v_A and v_B

For calculation of half-plane velocity of holonomic robot A with respect to robot B, the velocity of obstacle must be know which will form a set of allowed velocities in a given time τ.

$$u = (argmin_{v \in X} ||\nu - Y||) - Y \tag{2.1}$$

$$X = TVO^{\tau}_{A|B} \tag{2.2}$$

$$Y = (\nu_A - \nu_B) \tag{2.3}$$

$$ORCA^{\tau}_{A|B} = \{\nu | \left(\nu - (\nu_A + 0.5\nu_B)\right).n \geq 0\} \tag{2.4}$$

In equation 2.1, u represents the relative velocity vector for two entities $(v_A - v_B)$ to the nearest point of boundary set by the robot B for robot A. This boundary is known as Truncated Velocity Obstacle (TVO). Each robot or entity performs half of its duty in order to avoid collision. This gives the total velocity adjustment to be the half of u within the allocated time τ. The ORCA plane for robot A with respect to robot B is represented by equation 2.4.

From the pragmatic point of view, these techniques such as RVO, HRVO and ORCA require perfect information of the surroundings which imposes serious issues of cost and complexity, making aforementioned technique far from practically realizable [17] [19]. Also, there is a severe problem with ORCA which is *Deadlock*. This research work present a novel technique to prevent this *Dead-*

lock. Using simulation two different scenarios are presented in Figure 2.5 and 2.6 which show that even though the algorithm is quite efficient in collision avoidance however it gets worst when *Deadlock* situation arises.

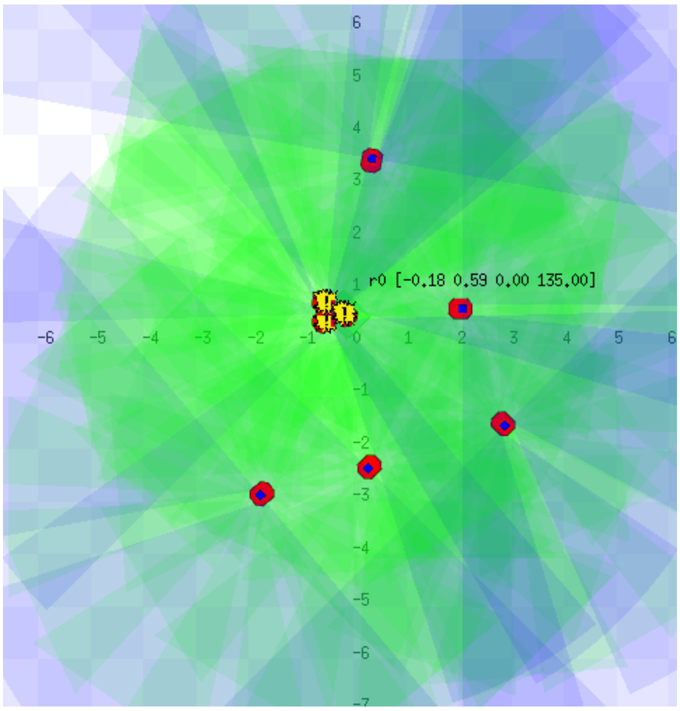

Fig. 2.5: Three Robots in Deadlock

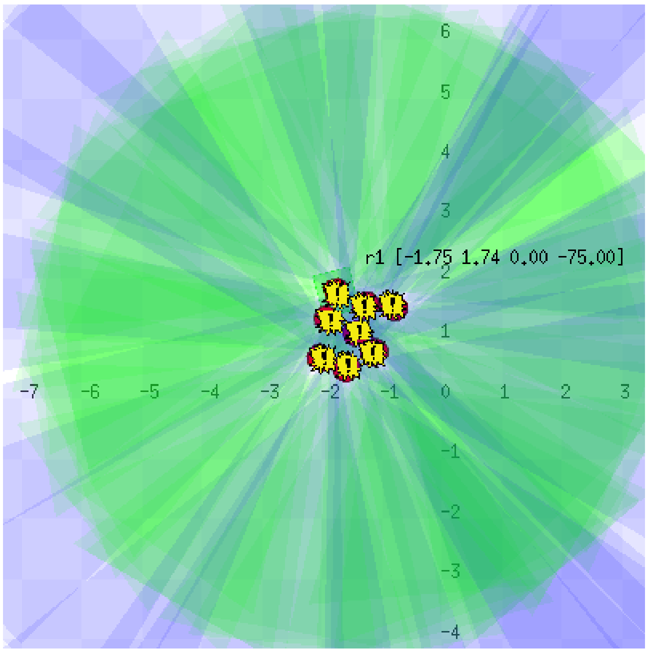

Fig. 2.6: All Eight Robots in Deadlock

An extensive study on collision avoidance is available in the literature but none of them address the problems of collision avoidance with zero deadlock scenario. [1][2] addresses the problem of collision avoidance assuming that the obstacles are static or moving slowly than the robot but in real life the robot may have to face with the fast-moving entities. If the fast-moving entities that also take decision intelligently to avoid collision which means they can change their velocity on the way to their goal, are present in the surroundings of the robot, this technique will simply fail to avoid collision. This technique might work if the velocities of other robots are known and this approach is presented as Reciprocal Velocity Obstacle (RVO) in the literature [1][3].

2.2 An Analysis of the Reciprocal Robots Collision Avoidance

In this paper, Aurel Fratu and Danut Ilea discussed the problem of collision briefly and proposed a solution that guarantees that there will be no collision. Each robot takes its decision independently without communication with the other robot. It is assumed that robots have the ability of perfectly sensing the environment that contains obstacles or other moving entities. The robots have circular shapes and moves only in two-dimensional space. The motion of the robot can be in any direction in order to avoid collision. The algorithm dictates that the robot must sense its environment and take decision based on the observation. If there is an obstacle or any other moving entity, its velocity is measured and possibility of collision is calculated. If the collision is imminent, the robot must change its velocity in order to avoid collision. This approach is studied well in complex simulations and the results were astonishingly successful. The simulation results of two robots that have different goals proved that no collision occurred and the goal of individual robot was successfully achieved.

2.2.1 Collision Detection:

There are so many technique available in the literature for collision detection but the most frequently used technique is Bounding Volume Hierarchies (BVHs). The visibility reducing algorithm as defined in [4] is used which in general can be used in any environment to detect the collisions. The complexity of input model and the output of the problem, both determines the performance of the system. The already available algorithm in the literature requires high memory of the computing device, therefore it is not recommended for large and complex problems. The presented algorithm tries to achieve high performance, generalization, large problems handling and approach to realistic scenarios. Potentially Collision Set: The collision is predicted based on the criteria that if the obstacle or the moving entity comes in the close vicinity of the robot or about to overlap it. If the collision is imminent, that object is added to potentially collision set or PCS. In this paper, potentially visible set or PVS is used for collision detection. If an object is about to collide in the image space, it is added to the PVS. Initially, all the objects are considered to be a part of PCS. But using reduction method, the objects are removed from PCSC, if the objects do not seem to collide.

Most of the time, in search of the most optimal path, the robot gets stuck or fail to find the solution to avoid deadlock.

Chapter 3
Problem Statement

Robots face a fundamental issue of collision when moving in an environment that contains obstacles or other moving robots or entities. Generally, when a robot is moving in an environment, it senses its surroundings and acts accordingly. It follows a certain set of rules to find its path in the environment keeping in view the observation of its surroundings where there are obstacles or other moving entities. These set of rules or instruction are modeled in order to achieve the destination by finding the shortest possible path.

Despite of the fact that the most efficient algorithms that have been developed that help the robot to navigate and find the most optimal path to its destination but they still get stuck in a *Deadlock* situation. This situation happens when two or more robots come in the vicinity of each other and take the same decision independently resulting in a collision which stops them to move any further and the robot has no other way to get out of the situation.

A simulation based on the existing algorithms of the literature is conducted. There are four pair of moving entities, representing the robots, and each entity is moving in opposite direction of its pair, forming a collisional situation.

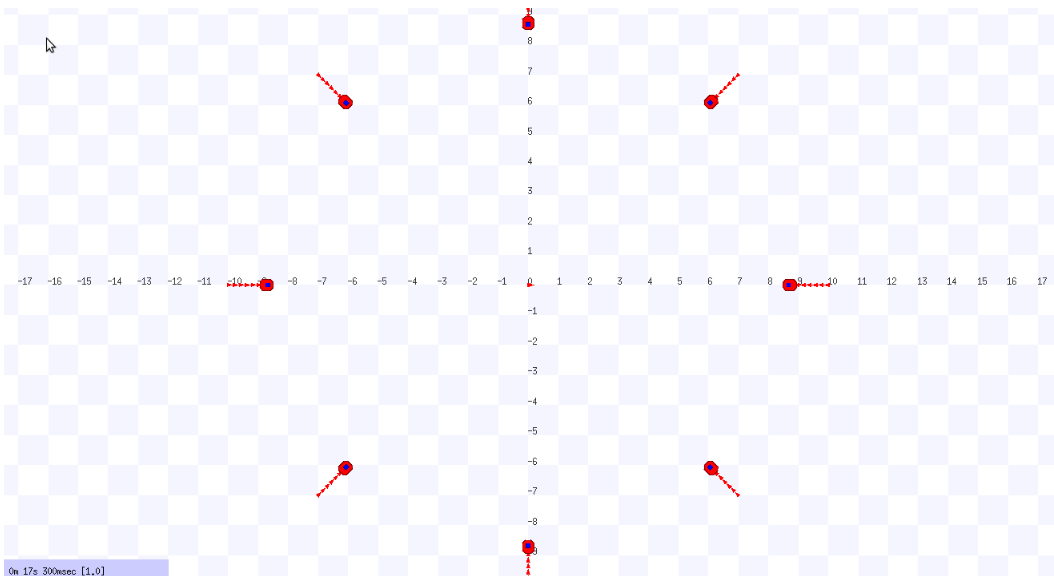

Fig. 3.1: After 17 Seconds from the starting point

The trajectory of individual robots is also traced out for better understanding. After 17 seconds from the starting points as shown in Figure 3.1, the trajectories are smooth and the robots are headed strait to their destination.

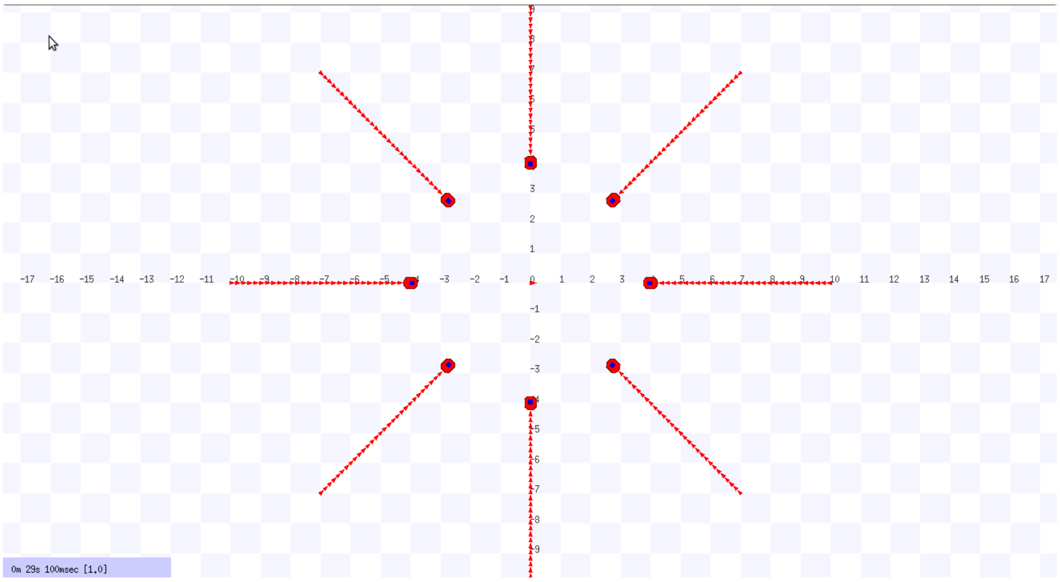

Fig. 3.2: After 29 Seconds from the starting point

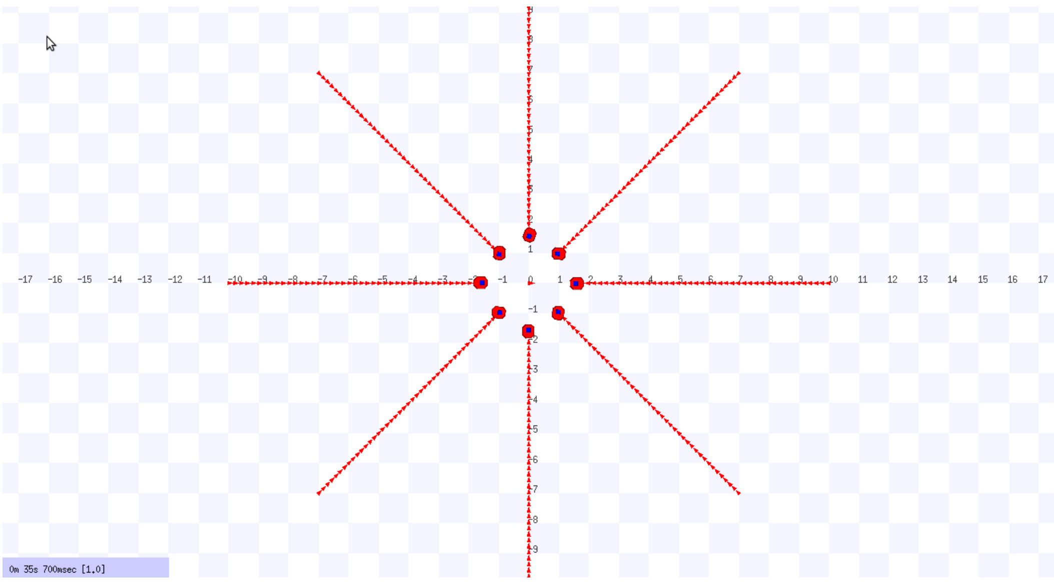

Fig. 3.3: After 35 Seconds from the starting point

After 29 seconds, all the robots are uniformly moving and their trajectories are also traced out which are strait lines showing optimal, most efficient paths to achieve their goals. However, after 35 seconds of starting the simulation, the situation have become even more dramatic. The collision seems imminent and all of the robots are going to collide. Based on the algorithm, it should easily prevented once a robot detects other robots in his vicinity. This is turning point situation, where the effectiveness of algorithm can be evaluated.

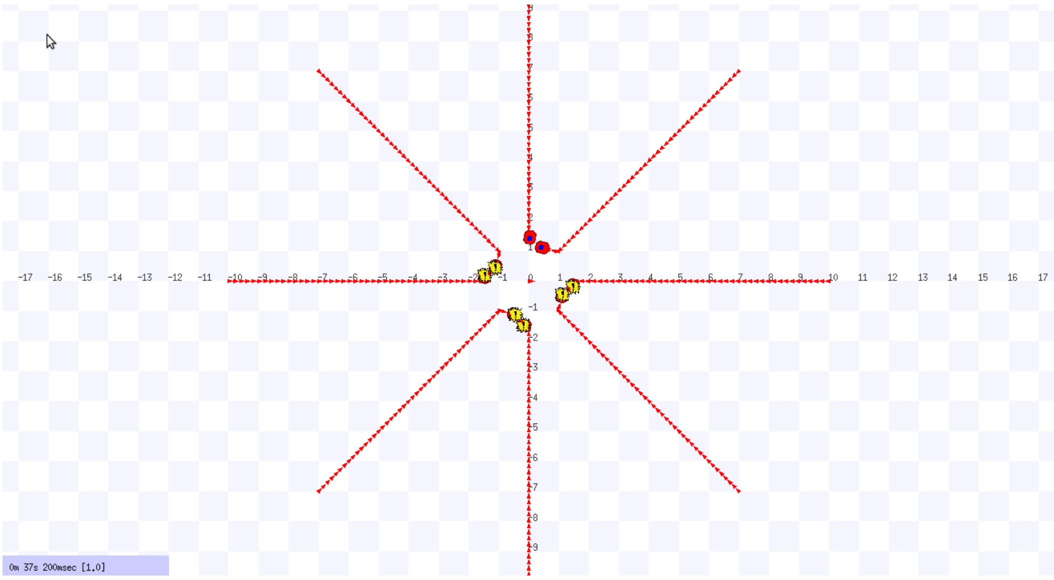

Fig. 3.4: After 37 Seconds, a Deadlock

After 37 seconds of starting the simulation, something unexpected has happened. The algorithm indeed helped the robots to avoid collision but they are not moving rather got stuck in perpetuating situation. Looking deep in the algorithm, it seems that algorithm seem functional but incomplete to counter such immovable situation. This situation is known as *Deadlock*. Despite of most effective algorithms, the whole system may go haywire, if this happens in real life. The results may be catastrophic. The simulation results show the severity of deadlock problem.

Chapter 4
Proposed Solution

Nature is the greatest scientific experiment to learn and find solution. Human also move from one place to another sharing the same space for maneuvering but they do not seem to get stuck in a *Deadlock* situation. The key point is all humans behave the same.

4.1 CORCA & V-CORCA

CORCA stands for Collaborative Optimal Reciprocal Collision Avoidance and V-CORCA stands for Velocity based Collaborative Optimal Reciprocal Collision Avoidance.

The first algorithm proposed in this thesis is the pioneer of its kind based on an approach discussed in [20]. This state of the art technique is named as Collaborative Optimal Reciprocal Collision Avoidance (CORCA). This technique is designed to eliminate the possibility of occurring a collision along with the problem of *Deadlock* in a multi-agent system. *Deadlock* is a situation in which two or more robots come close in each others vicinity such that they are unable to move to reach their target anymore. Figure 3.4 illustrate the phenomena of *Deadlock* among multiple robots.

Nature is the greatest teacher. Following this quote, the CORCA is also based on a natural phenomena of humans finding their way in a crowded situation. Hence, there are few governing rules rule based on the traffic rules and regulations which ensure a smooth collision and deadlock free maneuvering of robots. The robot make suitable decisions on their way to their target. These decisions are based on preferences such as in this case the right of the robot will serve as a preferred option to avoid collision. Meanwhile, Potentially Colliding Sets (PCS) or Potentially Colliding Regions (PCR) are formed to maintain an adequate distance among the robots [21]. The Figure 4.1 and 4.2 illustrates the overlapping PCS in a simulation of eight robots.

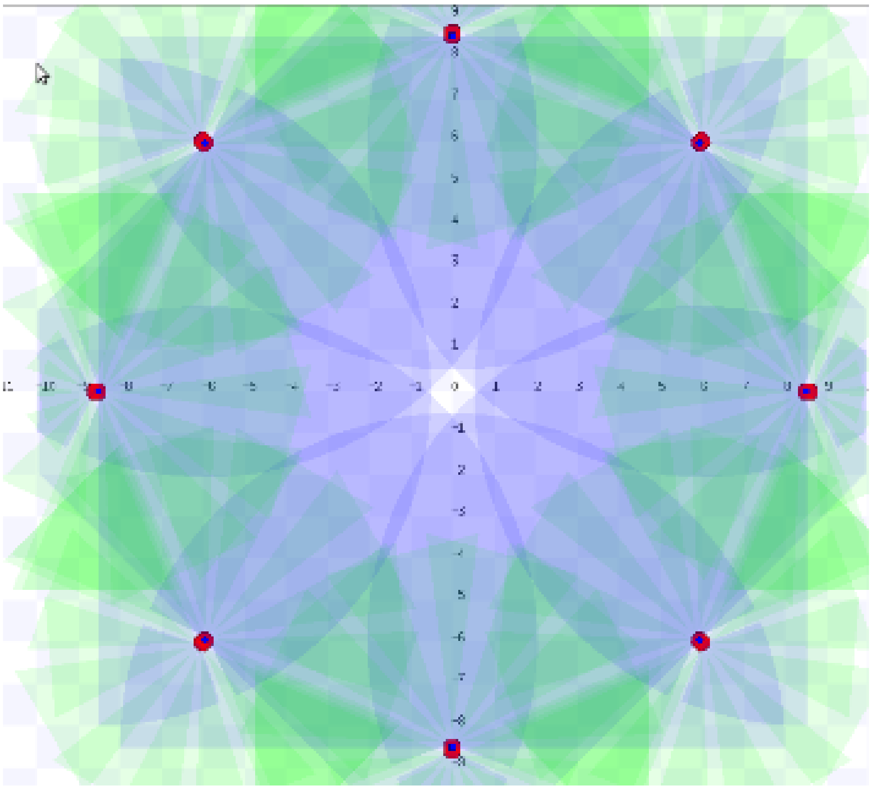

Fig. 4.1: All Robots Start from their Initial Positions

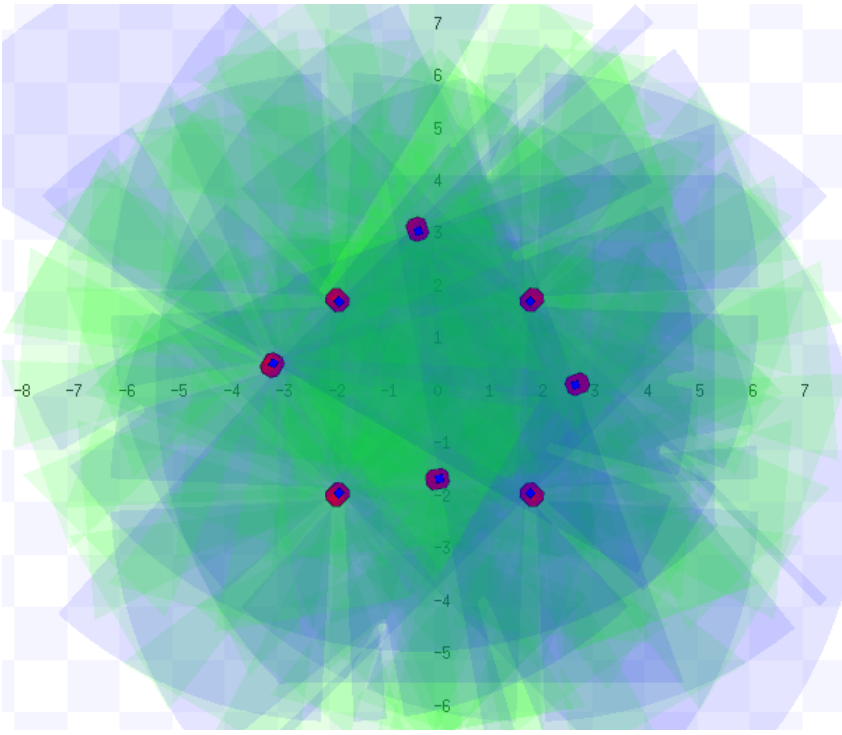

Fig. 4.2: Robot moving in Diverging PCS

If the PCS of each robot is not overlapping with others, the robots can make decision based on traffic-rules to avoid collision ignoring the complete set of PCS. These robots are highlighted in the simulation. In Figure 4.1, there are eight holonomic robots and the initial position of each robot is shown. The field of view and PCS of each robot is represented as blue and green respectively. The PCS of each robot is same as they are equidistant from each other in a circular shape. On the other hand, the Figure 4.2 shows that as the robots move towards their goals, the robots in PCS are highlighted indicating that they are now taking decisions based on the algorithm and information retrieved from the sensors in order to avoid collision. In short, the Figures 4.1 and 4.2 illustrates the basic underlying mechanism of robot for decision making and avoiding collision without communicating with each other.

The sensors feed the information about the surroundings such as the velocity and position of other robots in the same plane. In response, the robots take suitable decision based on traffic rules to avoid possible collision. If somehow, the robots gets too close to each which is minimum allowed distance between the robots the an appropriate decision is made to change its course. This change in direction will not only prevent the collision but the *Deadlock* well in time.

$V \leftarrow \{V_{init}\}$
$\omega \leftarrow \phi = 0$
$D \leftarrow \{D_{MinDistance}\}$
$T \leftarrow \{T_{init}\}$
for $j \leftarrow 0$ **to** N **do**
    $O_j \leftarrow PCS(j)$
    for $i \leftarrow 0$ **to** N **do**
        $D_{robot} \leftarrow SonarProxy(i)$
        $T \leftarrow TrafficRulePriority(i)$
        if $D_{robot} < D$ **Then**
            $V \leftarrow \{V_{turn}\}$
            $\omega \leftarrow \phi = TurnSpeed$
        else
            $V \leftarrow \{V_{init}\}$
            $\omega \leftarrow \phi = 0$
            $R \leftarrow (V, \omega, T)$
return R

Table 4.1: CORCA Algorithm

V-ORCA is derived from ORCA but inspired from traffic rules that humans follow in their daily life to avoid collisions. The key difference between ORCA and V-ORCA the use of information for decision making. ORCA utilized the position information of the robots and other robots in the surroundings in order to take appropriate action for smooth maneuvering. On the other hand, V-ORCA uses the velocity information of the robot itself and the robots of the surroundings. This comparison of velocities enables the robot to take decision well in time to avoid possible collision. In both the case i.e. ORCA and V-ORCA, the decision is made based on PCS set. Using this PCS set, the robot differentiate between the colliding robots and those which far away sufficiently.

Figure 4.3 illustrates few scenarios in which traffic rules steer the robot to avoid collision and *Deadlock* situations. In the first part, there is an imminent chance that the two robots A & B will collide, therefore, the traffic rules based algorithm performs its job and guides the robots. The robot A will take its preferred side which is right side but the robot B will wait until the robot A is passed from its course.

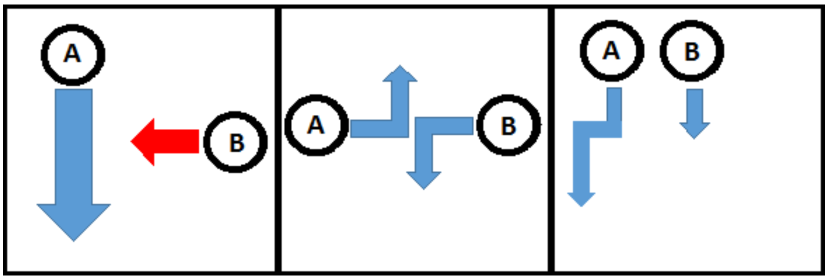

Fig. 4.3: Traffic Rules Representation

In the second part, the Figure 4.3 illustrates a scenario in which two robots are moving toward each other and the collision seems imminent. To avoid this collision, the two robots will sense the situation first then based on the algorithm steer to their non-preferred side which mean both the robot will steer to their left sides. In this way, both the robots will continue to their target without colliding with each other. In the last part, when two or more robot which are traveling in the same direction somehow comes close to each other. The algorithm dictates that if this situation happens then the robots should change their lane in order to maintain a significant distance. In the present case, the robot a will change in direction to the preferred side (which is its right side). If there was a third robot C on the other side of robot B, then that robot C will have to change its lane away from robot B. These traffic-rules based algorithm ensures that there will be not even a remote for the robot to collide or get stuck to indefinite time.

In figure 4 given below, after initialization of velocities, location, orientation and distance from other robots, the PCS is repeatedly calculated.

The collision avoidance is achieved using PCS. This PCS is updated continuously based on the information retrieved from the sensors. The overall algorithm that controls and dictates the course of action is briefly described in Table 4.2 in the form of pseudo code.

$V \leftarrow \{V_{init}\}$
$\omega \leftarrow \phi = 0$
$D \leftarrow \{D_{MinDistance}\}$
$T \leftarrow \{T_{init}\}$
for $j \leftarrow 0$ **to** N **do**
    $O_j \leftarrow PCS(j)$
    for $i \leftarrow 0$ **to** N **do**
        $D_{robot} \leftarrow SonarProxy(i)$
        $(V_{min}, V_{max}) \leftarrow AvoidCollision(i, D_{robot}, V)$
        $T \leftarrow TrafficRulePriority(i)$
        if $V_{min} < V_{robot} < V_{max}$ **Then**
            $V \leftarrow \{V_{turn}\}$
            $\omega \leftarrow \phi = TurnSpeed$
        else
            $V \leftarrow \{V_{init}\}$
            $\omega \leftarrow \phi = 0$
            $R \leftarrow (V, \omega, T)$
return R

Table 4.2: V-CORCA Algorithm

Chapter 5
Simulation & Results

To determining the action of robot at a certain point of its track, the designer usually apply feedback control strategies. The stability of these control algorithm is of prime concern. The facility where these control algorithms can be implemented and tested is quite expensive and the designers cannot take risk to actually test their algorithm in the practical environment. Therefore, a computer simulation is the best option to test the algorithms in which a virtual robot can interact with the simulated environment and takes decision based on the set of instructions given in the algorithm.

5.1 Simulation

Two different observation are performed for three different collision avoidance techniques are simulated and analyzed. These observation are compiled based on different velocity of robots. During first observation, the velocity of robot is chosen to be $0.35m/s$ and $0.7m/s$ in the second observation. The open source platforms such as Player-Stage is used to conduct the simulations. The environment was develop to mimic the actual real life scenario. The robots were positioned at predetermined locations and the obstacles were deployed in order to assess the performance. The algorithm was programmed that will control and guide the motion of robot. The robot its self is presented as an entity.

Figure 5.1 illustrates the drawback of ORCA. Eight robots are simulated of which two get stuck in deadlock situation. The ORCA is working but the robots could not find a possible solution to get out this situation. On the other hand the results of CORCA are shown in Figure 5.2. There is no collision or deadlock. Similarly, the Figure 5.3 illustrates the performance of V-CORCA. It can be seen that there is no collision or deadlock. However the robots are moving more freely in case of V-ORCA than in case of CORCA. It is due to time efficiency which is inherited by V-CORCA.

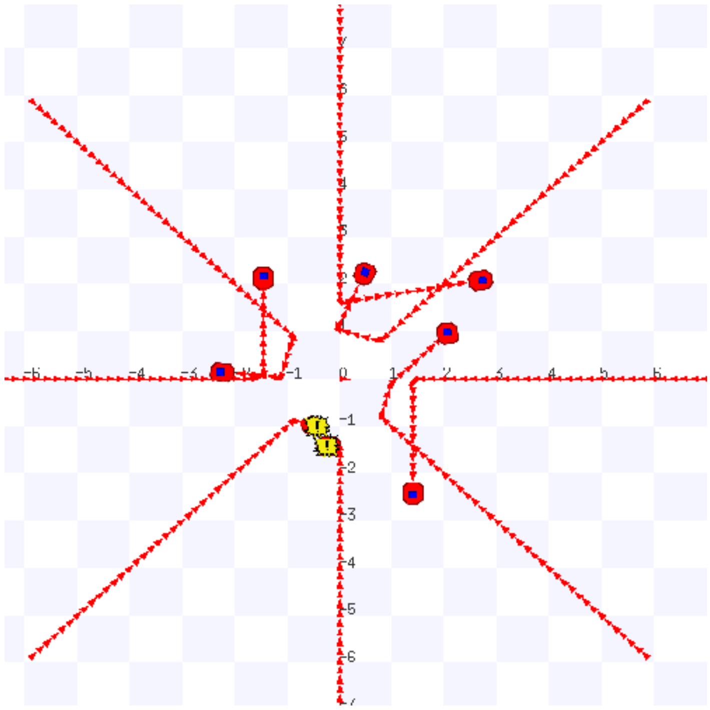

Fig. 5.1: Deadlock in case of ORCA with 0.35 m/s velocity of robot

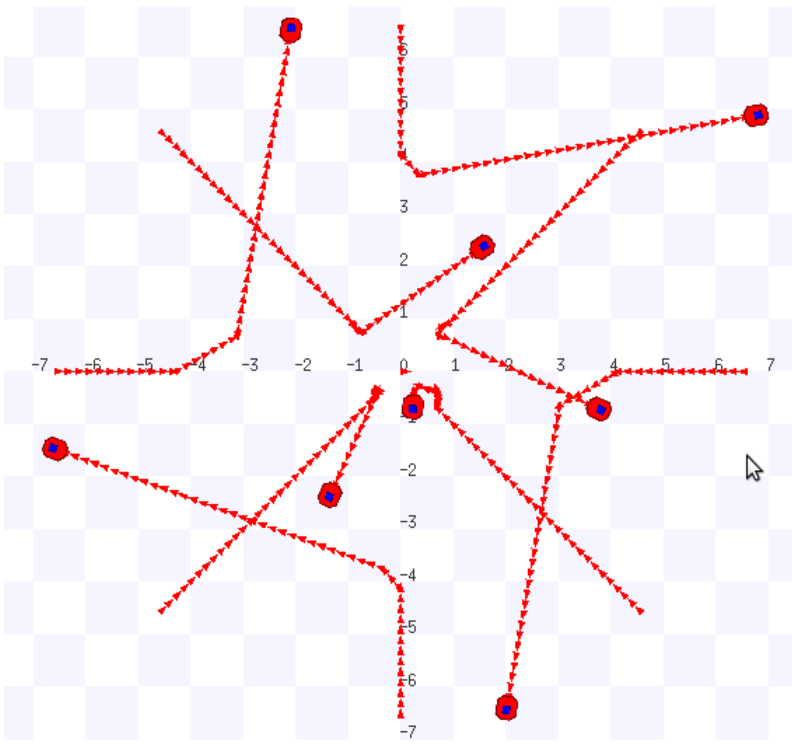

Fig. 5.2: CORCA with 0.35 m/s velocity of robot

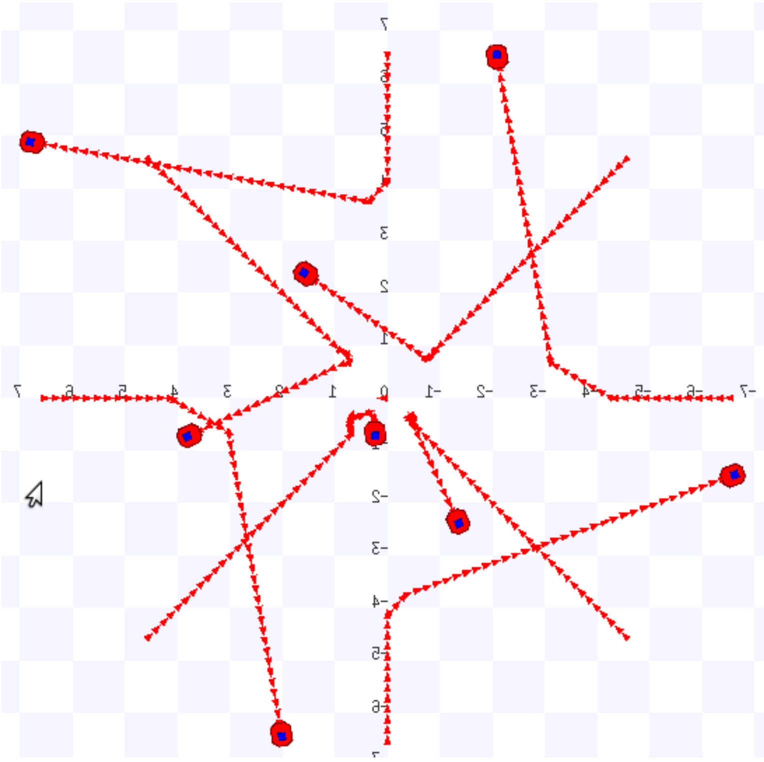

Fig. 5.3: V-CORCA with 0.35 m/s velocity of robot

If the velocity of robots is increased, the collision avoidance behavior also changes drastically. To analyze the impact fo velocity change, lets consider the scenarios in which the robots velocity is selected to be $0.7m/s$ and repeat the same simulation for ORCA, CORCA and V-CORCA. Figure 5.4 shows that with the slight change in speed has change the behavior of robots dramatically. In the present case, six out of eight robots got stuck in deadlock situation. However in the previous case where the velocity of robot was selected to be $0.35m/s$, there were only two of eight robot were subject to deadlock. On the other hand, CORCA and V-CORCA were successful in avoiding collision and deadlocks. These phenomenas can be seen in Figures 5.5 and 5.6.

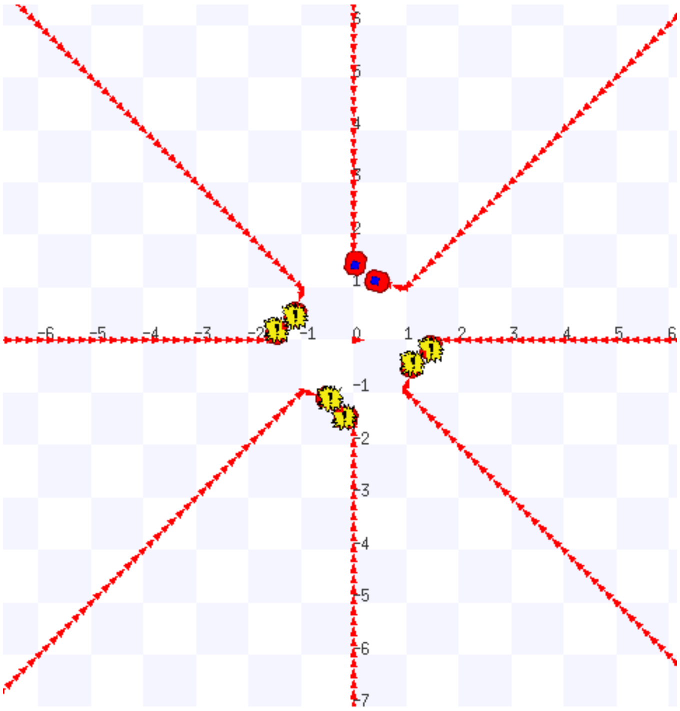

Fig. 5.4: Deadlock in case of ORCA with 0.7 m/s velocity of robot

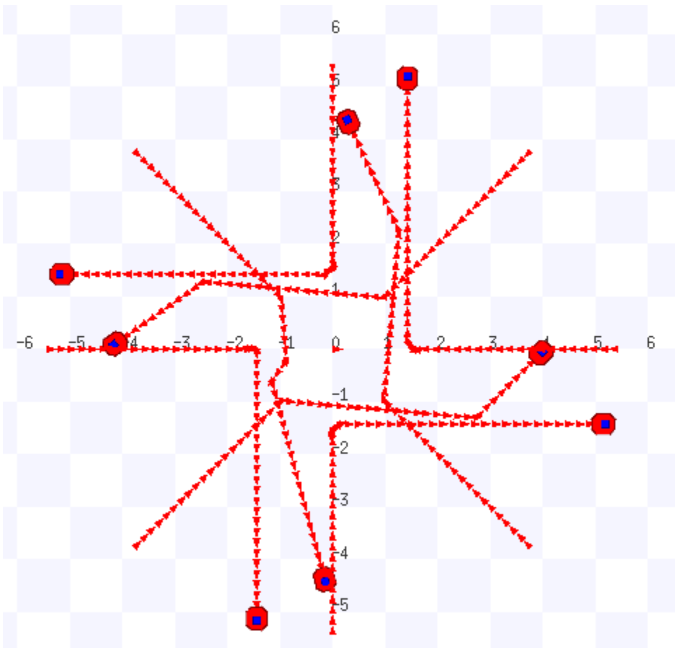

Fig. 5.5: CORCA with 0.7 m/s velocity of robot

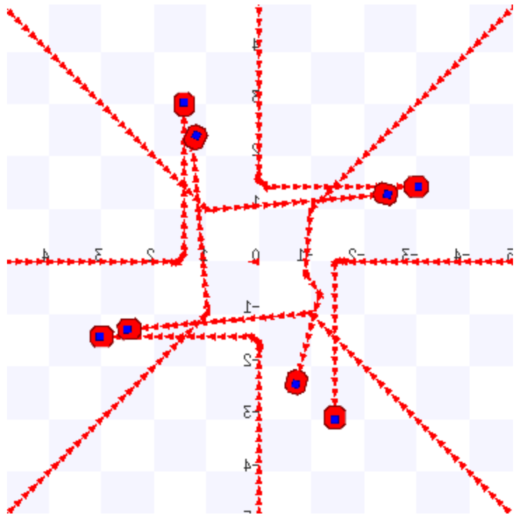

Fig. 5.6: V-CORCA with 0.7 m/s velocity of robot

5.2 Consolidated Simulation Results

The consolidated simulation results of ORCA, CORCA and V-CORCA for different selected velocities are presented briefly in this section. There are several parameters that can be considered for performance evaluation but few of them have significant impact. Those parameters are listed in Table 5.1 which act as a foundation for examining the outcomes of each collision avoidance technique.

Parameter	**Abbreviation**
N	number of observations
Tsimulation (hr)	time per simulation
Vmin (m/s)	minimum velocity
Vmax (m/s)	maximum velocity
Vaverage (m/s)	average velocity
Nonehalt	number of instances when a single robot halts
Nnhalt	number of instances when multiple robots halt
Ndeadlock	number of deadlocks for two or more robots

Table 5.1: Performance Evaluation Parameters

By carefully analyzing the observations given in Table 5.2, it can be deduced that total of 120 samples are taken which mean that the simulation was repeated 120 times for each collision avoidance technique and each time it took almost an hour to complete. Other parameters such minimum and maximum velocities are kept the same for each technique in order to achieve a better comparison. It can also be observed that the average velocity of robots in case of ORCA and V-ORCA is slightly higher than that of CORCA. It implies that in case of CORCA the robots tend to slow down more often as compared to ORCA or V-ORCA. The major cause of this behavior is due to the inherit design if individual collision avoidance technique. ORCA and V-CORCA are velocity based techniques, however CORCA is position based technique. Similarly, the Nonehalt and Nnhalt parameters dictates the efficiency of each collision avoidance technique. Nonehalt parameters shows that the robots are lacking the decision making ability, therefore, they have to slow down in order to avoid collision. The most efficient technique was V-CORCA which had the least number of instances but there are still more than 7,000 instances which is significant enough for further research.

Parameter	**ORCA**	**CORCA**	**V-CORCA**
N	120	120	120
Tsimulation (hr)	1	1	1
Vmin (m/s)	0.2	0.2	0.2
Vmax (m/s)	0.7	0.7	0.7
Vaverage (m/s)	0.5	0.45	0.5
Nonehalt	7,936	8,092	7,033
Nnhalt	774	786	622
Ndeadlock	252	0	0

Table 5.2: Observations for ORCA, CORCA and V-CORCA

Likewise, in case of Nnhalt instances in which two or more robots halt simultaneously are also quite significant for all three collision avoidance techniques. In comparison to ORCA and CORCA, V-CORCA shows the best performance i.e. the average velocity is significant, Nonehalt and Nnhalt are sufficiently low to make a difference. All these observations imply that V-CORCA is so far the best collision avoidance technique. This is also the major accomplishment of this research.

The following Figures 5.7, 5.8, 5.9, 5.10, 5.11 and 5.12 illustrate the motion of robots in the plane coexisting with each other without colliding or getting stuck in *Deadlock* situation.

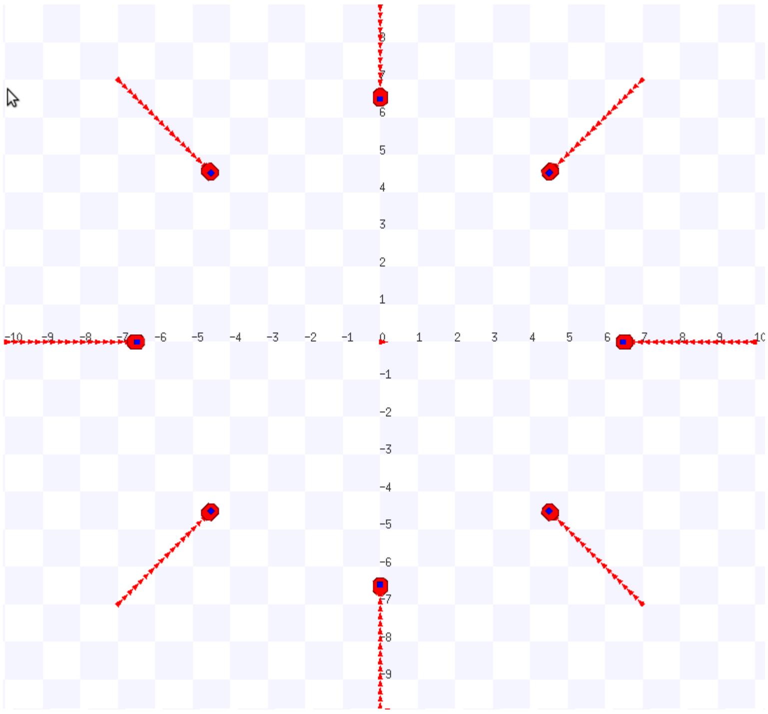

Fig. 5.7: V-CORCA Simulation 1

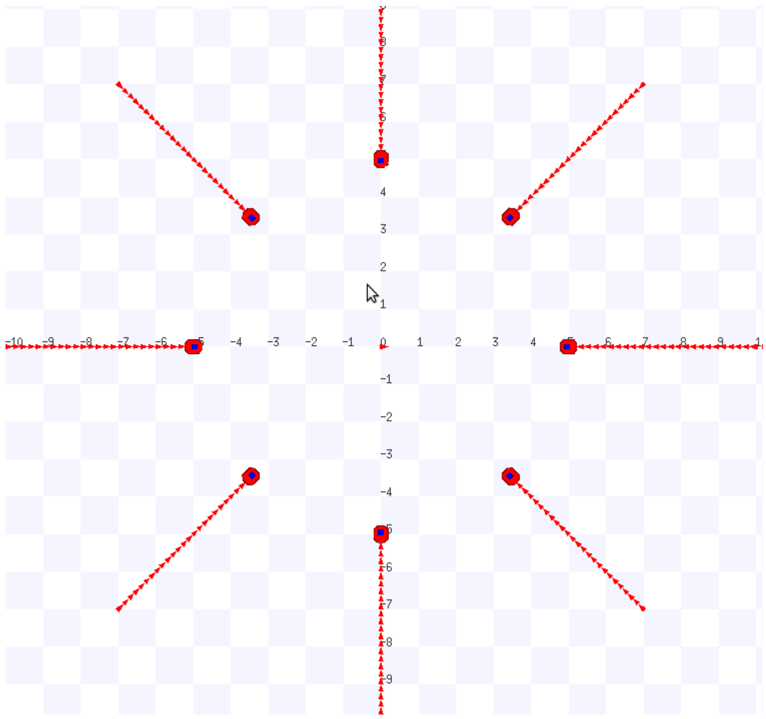

Fig. 5.8: V-CORCA Simulation 2

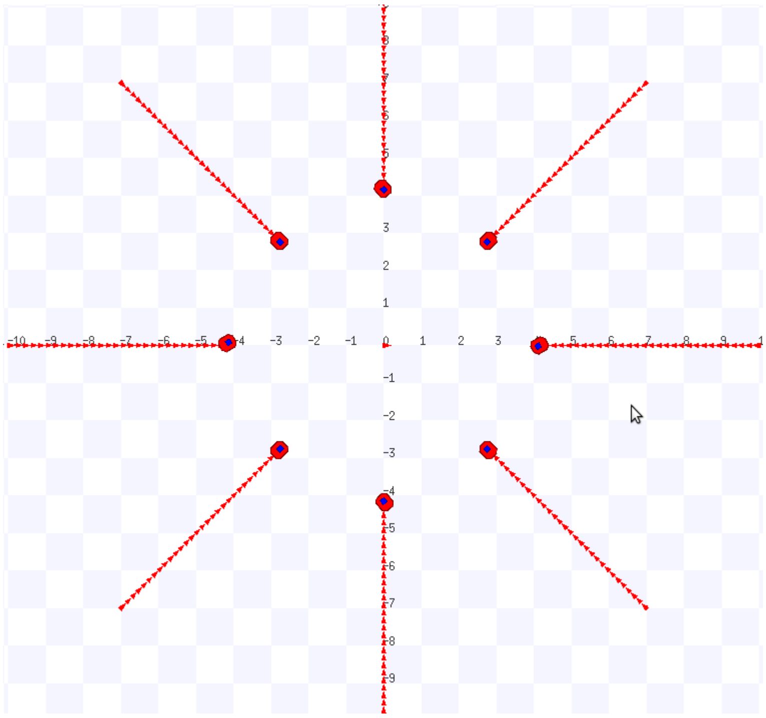

Fig. 5.9: V-CORCA Simulation 3

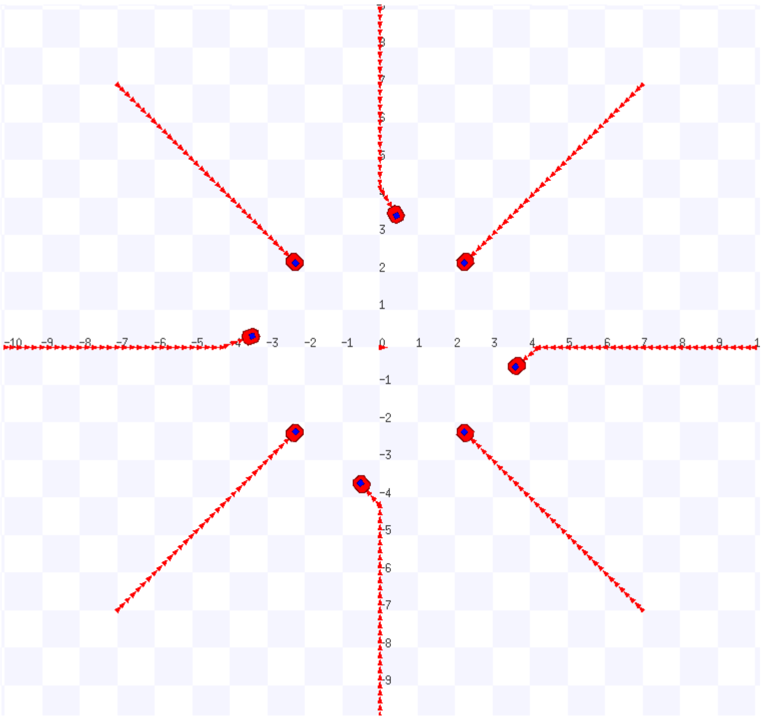

Fig. 5.10: V-CORCA Simulation 4

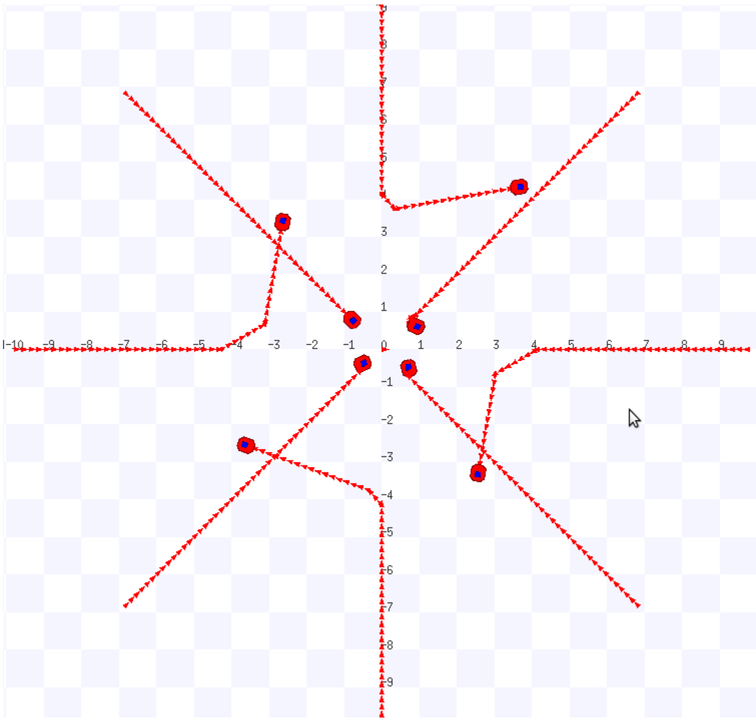

Fig. 5.11: V-CORCA Simulation 5

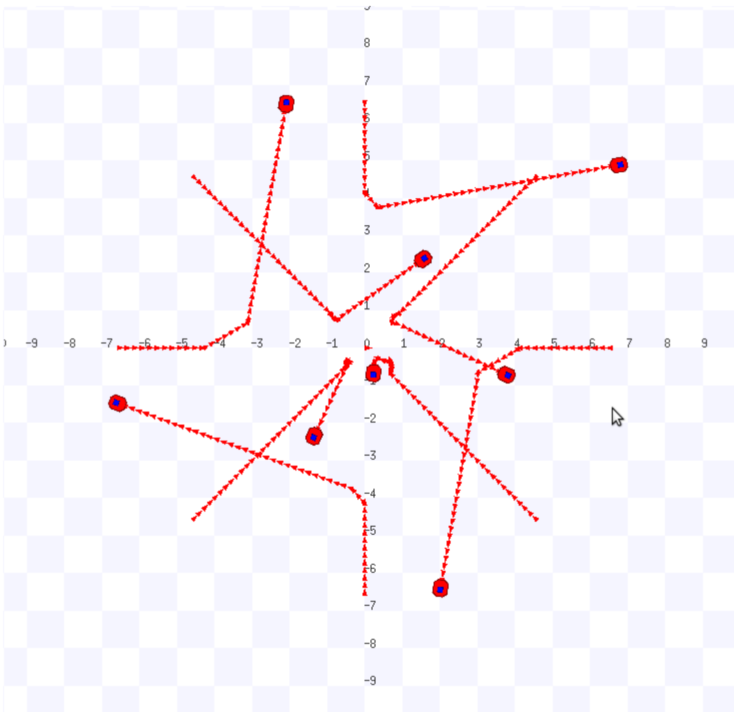

Fig. 5.12: V-CORCA Simulation 6

Chapter 6
Conclusion & Future Work

6.1 Conclusion

The collision avoidance technique such as ORCA is known to be an efficient technique to avoid collision for multi-agent holonomic systems. But this technique fails to cater the problem of deadlock in which the robot cease to move for an indefinite period. That problem is solved by newly proposed technique called CORCA. Simulation results of CORCA have shown that it is an efficient technique to avoid collision and also to cater deadlock scenarios. Since CORCA relies on the information of position of the robots therefore, the decision making mechanism is quite slow. Hence, an other approach is proposed call V-CORCA. In comparison to all other techniques, V-CORCA is found to be best collision avoidance technique that not only caters the problem of deadlock but improves the decision making ability of robot.

6.2 Future Work

- From the simulation results, it has been observed that the robots lack the ability of efficient decision making well in time. The robots have to slow down each time when they making decision to avoid collision or a deadlock. This decrease in velocity making the whole system inefficient in terms of time efficiency and response.

- There is a need of technique that will not only improve the decision making ability of robots but also the save the overall maneuvering time to reach the goal.

- So far, the simulation have been conducted to evaluate the performance of algorithm. There is need of practical application as the pragmatic scenario reveal the deep insights of problems. After which the true performance of algorithms can be determined.

- A swarm based systems which contains thousands or millions of robots can be a good platform for performance evaluation. V-CORCA may also be tested using swarm based systems in future.

References

[1] Fulgenzi, A. Spalanzani, C. Laugier., "Dynamic obstacle avoidance in uncertain environment combining PVOs and occupancy grid." *IEEE International Conference Robot Autom.*, 2007, pp. 1610-1616. 9

[2] Y. Abe, M. Yoshiki., "Collision avoidance method for multiple autonomous mobile agents by implicit cooperation." *IEEE RSJ International Conference Intelligent Robotic System*, 2001, pp. 1207-1212. 9

[3] M. de Berg, O. Cheong, M. van Kreveld, M. Over mars., "Computational Geometry: Algorithms and Applications." *Springer-Verlag*, 2008. 9

[4] S. Guy, J. Chhugani, C. Kim, N. Satish, P. Dubey, M. Lin, D.Manocha., "Highly parallel collision avoidance for multi-agent simulation.", *University of North Carolina at Chapel Hill, Tech. Rep.*, 2009. 9

[5] Kaushik Rajashekara, Qingchun Wang, Kouki Matsuse, "Flying Cars: Challenges and Propulsion Strategies", *IEEE Electrification Magazine*, Volume 4, Issue 1, March 2016, pp. 46-57. 1

[6] Nurbaiti Wahid; Hairi Zamzuri, Mohd Azizi Abdul Rahman; Satoshi Kuroda, Pongsathom Raksincharoensak, "Study on Potential Field based Motion Planning and Control for Automated Vehicle Collision Avoidance Systems", *IEEE International Conference on Mechatronics (ICM)*, 2007, pp. 208-213. 1

[7] Mohamed Abdelaal, Axel Hahn, "NMPC-based trajectory tracking and collision avoidance of unmanned surface vessels with rule-based colregs confinement", *IEEE Conference on Systems, Process and Control (ICSPC)*, 2016, pp. 23-28. 1

[8] Seema Rawat, Zohaib A. Faridi, Praveen Kumar, "Analysis and proposal of a novel approach to collision detection and avoidance between moving objects using artificial intelligence", *International Conference System Modeling & Advancement in Research*, 2016, pp. 135-138. 1

[9] Chai Xue, Wang Ganglin, Wu Zhe, "Flight control system design for the flying-wing aircraft by using intelligent optimal seeking method", *2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010)*, Volume 2, 2010, pp. 429-432. 1

[10] Daizong Li, "Design of a new VTOL UAV by combining cycloidal blades and FanWing propellers" *IEEE Aerospace Conference*, 2013, pp. 1-8. 1

[11] IEEE Spectrum, "The Big Picture: My Other Car Is A Flying Saucer" *IEEE Journals & Magazines*, Volume 44, Issue 11, 2007, pp. 24-25. 1

[12] R. W. Beard, T. McLain, D. Nelson, D. Kingston, & D. Johanson, "Decentralized cooperative aerial surveillance using fixed wing miniature UAVs", *Proceedings of IEEE*, 94(7), 2006, pp. 1306-1324.

[13] A. Ollero, "Aerial robotics cooperative assembly system (ARCAS): First results", *Aerial Physically Acting Robots (AIRPHARO) Workshop*, IROS 2012, Vilamoura, Portugal, October 2012, pp. 7-12.

[14] L. Merino, F. Caballero, J. M. de Dios, I. Maza, & A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement", *Journal of Intelligent and Robotic Systems*, 65(1), 2012, pp. 533-548. 3

[15] O. Khatib, K. Yokoi, O. Brock, K. Chang, A. Casal, "Robots in human environments: Basic autonomous capabilities", *The International Journal of Robotics Research*, 18(7), 1999, pp. 684-696. 3

[16] J. Snape, J. van den Berg, S. J. Guy, & D. Manocha, "Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles", *Proceedings of 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 5917-5922. 3

[17] D. Claes, D. Hennes, K. Tuyls, & W. Meeussen, "Collision avoidance under Bounded Localization Uncertainty", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 1192-1198. 4

[18] D. Hennes, D. Claes, W. Meussen, & K. Tulys, "Multi-robot collision avoidance with localization uncertainty", *Proceedings of 11th International conference on Autonomous Agents and Multi-agent Systems*, Valencia, Spain. June. 2012. 4

[19] A. Levy, C. Keitel, S. Engel, & J. McLurkin, "The Extended Velocity Obstacle and Applying ORCA in Real World", *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, Washington, May 2015, pp. 26-30. 4, 7

[20] S. A. Khan, Y. Ayaz, M. Jamil, S. O. Gilani, M. Naveed, A. H. Qureshi, K. F. Iqbal, "Collaborative Optimal Reciprocal Collision Avoidance for Mobile Robots", *International Journal of Control and Automation*, 8(8), 2015, pp. 203-212. 5

[21] S. Guy, J. Chhugani, C. Kim, N. Satish, P. Dubey, M. Lin, et al., "Clea-Path: Highly parallel collision avoidance for multi-agent simulation", *ACM SIGGRAPH/ Eurographics Symposium on Computer Animation*, 2009, pp. 177-187. 7

14

14