CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Environment-Aware Resilient Multi Robot System in Constrained Conditions

by

Faheem Gulzar

A dissertation submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Engineering
Department of Electrical and Computer Engineering

2025

Environment-Aware Resilient Multi Robot System in Constrained Conditions

By

Faheem Gulzar

(DEE193004)

Dr. Raja Syamsul Azmir Abdullah, Professor
Universiti Putra Malaysia, Malasyia
(Foreign Evaluator 1)

Dr. Shihua Li, Professsor
School of Automation, Southeast University, China
(Foreign Evaluator 2)

Dr. Noor Muhammad Khan
(Research Supervisor)

Dr. Noor Muhammad Khan
(Head, Department of Electrical & Computer Engineering)

Dr. Imtiaz Ahmad Taj
(Dean, Faculty of Engineering)

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
ISLAMABAD

2025

This thesis is dedicated to my parents and siblings.

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD
Expressway, Kahuta Road, Zone-V, Islamabad
Phone:+92-51-111-555-666 Fax: +92-51-4486705
Email: info@cust.edu.pk Website: https://www.cust.edu.pk

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in the dissertation, entitled "**Environment-Aware Resilient Multi Robot System in Constrained Conditions**" was conducted under the supervision of **Dr. Noor Muhammad Khan**. No part of this dissertation has been submitted anywhere else for any other degree. This dissertation is submitted to the **Department of Electrical & Computer Engineering, Capital University of Science and Technology** in partial fulfillment of the requirements for the degree of Doctor in Philosophy in the field of **Electrical Engineering.** The open defence of the dissertation was conducted on **June 26, 2025**.

Student Name : Faheem Gulzar (DEE193004)

The Examination Committee unanimously agrees to award PhD degree in the mentioned field.

Examination Committee :

(a)	External Examiner 1:	Dr. Mukhtar Ullah Professor FAST-NUCES, Islamabad	
(b)	External Examiner 2:	Dr. Laiq Khan Professor CUI, Attock Campus	
(c)	Internal Examiner :	Dr. Fazal ur Rehman Professor CUST, Islamabad	Fazal Rehman

Supervisor Name : Dr. Noor Muhammad Khan, Professor, CUST, Islamabad

Name of HoD : Dr. Noor Muhammad Khan, Professor, CUST, Islamabad

Name of Dean : Dr. Imtiaz Ahmad Taj, Professor, CUST, Islamabad

AUTHOR'S DECLARATION

I, **Faheem Gulzar** (Registration No. **DEE193004**), hereby state that my dissertation titled, "**Environment-Aware Resilient Multi Robot System in Constrained Conditions**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/ world.

At any time, if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my PhD Degree.

(Faheem Gulzar)

Dated: 26 June, 2025

Registration No : DEE193004

PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the dissertation titled "**Environment-Aware Resilient Multi Robot System in Constrained Conditions**" is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete dissertation has been written by me.

I understand the zero-tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled dissertation declare that no portion of my dissertation has been plagiarized and any material used as reference is properly referred/ cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled dissertation even after award of PhD Degree, the University reserves the right to withdraw/ revoke my PhD degree and that HEC and the University have the right to publish my name on the HEC/ University Website on which names of students are placed who submitted plagiarized dissertation.

(**Faheem Gulzar**)

Dated: 26 June, 2025

Registration No : DEE193004

List of Publications

It is certified that following publication(s) have been made out of the research work that has been carried out for this dissertation:-

1. **Faheem Gulzar**, Noor Muhammad Khan, Yasir Awais Butt and Aamer Iqbal Bhatti, "Constraint-oriented Formation Control of Multi-robot System in Leaderless Consensus under Confined Conditions," *Systems Science & Control Engineering*, vol. 12, no. 1, article no. 2436666, 2024.

2. **Faheem Gulzar**, Yasir Awais Butt and Aamer Iqbal Bhatti, "Adaptive Cruise Control for Ground Vehicles using Control Barrier Function under Weather based Surface Conditions," *19th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan*, pp. 573-578, 2022.

(Faheem Gulzar)

Registration No: DEE193004

Acknowledgement

All Praises to ALLAH Almighty, the Most Gracious and the Most Merciful.

First and foremost, I would like to express my profound gratitude to my supervisor, Dr. Noor Muhammad Khan, whose expertise, patience and thoughtful feedback have been invaluable in shaping my research in significant ways and i am grateful to him for the mentorship he has provided.

I express my gratitude to Dr. Aamer Iqbal Bhatti; a source of inspiration in the field of control engineering, Dr. Imtiaz Taj, for their exceptional guidance, steadfast support and inspiring teaching throughout my PhD coursework.

Furthermore, I would like to thank the members of CASPR and ARWiC research teams for their guidance and encouragement.

Above all, I am highly thankful to my father, who taught me patience, devotion and commitment. I am also indebted to my mother, brother and sisters, who have always provided me with motivation and relentless support to pursue higher studies.

(Faheem Gulzar)

Abstract

Multi robot systems have gained prominence in various domains like agriculture, surveillance, security operations, etc., due to their potential to solve complex problems collaboratively. The coordination control of the robots in these systems manifests the crucial research challenges primarily in terms of the resilience in execution of the tasks specifically under an evolving environment scenario. The existing frameworks designed for the tasks completion by the team of robots lack an environment awareness and are robot centric. These frameworks only consider the robots and their specialties required for the execution of assigned multi robot tasks. In practical applications these robot centric frameworks are vulnerable to the dynamically changing environment. The inactive status of the robots due to on-the-fly changes in an environment can seriously hamper the desired resilience of a multi robot framework to execute the assigned tasks in the continued fashion. Under these circumstances, the current frameworks are needed to be made capable to respond to such changing environment through the capability of the environment-aware resilience. The capability of the environment-aware resilience helps to achieve the switching of the assigned tasks among the robots for their continued execution. The prime focus of this research is to achieve the resilience of tasks that is, the accomplishment of assigned tasks by a multi robot system under the dynamically evolving environment in the constrained conditions of robots velocities while avoiding obstacles and inter-robots collisions. The resilience of tasks is useful as these tasks are usually issued once at the initiation of a multi robot based mission to the robots possessing specific specialties. These once-issued or once-assigned tasks are unable to be re-issued/reassigned under the unforeseen environment faced by the robots thus halting the execution of tasks in the midway. During the mission course, the resilient framework is able to issue/withdraw/re-issue the tasks to the robots under the influence of active/inactive/re-active status of the specialties as soon as these specialties get affected by the environment. Therefore in this thesis, a simultaneous mechanism for the tasks issuance/withdrawal/re-issuance coupled with specialties is developed incorporating all such robots that can perform the tasks. The scenario merits corresponding activation/deactivation/re-activation of the specialties connected with

the tasks issuance/withdrawal/re-issuance strategy for an achievement of the resilience of tasks. For this, a novel dynamic task-specialty matrix based mechanism termed as environment module is proposed and incorporated in an optimization technique developed in this thesis. The environment module has environment coefficient vectors each associated with the task issuance strategy and specialties of the robots in the team. The manifestation of the resilient execution of the multiple tasks is carried out through the three different cases viz. (*i*) simultaneous task re-issuance and specialties re-activation, (*ii*) mid-course withdrawal of the tasks and (*iii*) mid-course degradation of the environment coefficients. The optimization technique based decentralized controllers let the robots safely accomplish the respective tasks optimally under constrained conditions in a resilient fashion as showcased in the simulation results presented in this thesis, thus providing an evidence of working of the proposed technique. These results, in terms of the optimization variables' behavior, are found to be in agreement with the existing work. The gradual development of the presented thesis work leads to first exploration of the concept of control barrier function. For this, an application of the control barrier function is sought through solving a specific scenario of an adaptive cruise control in ground vehicles under the weather affected road surfaces and corresponding Matlab© simulation results for the scenario are presented in this thesis. Moreover during the research course, in order to show the multi robot tasks execution using optimization controllers leveraging control barrier functions, the constraint-oriented formation control of a first-order multi robot system is also achieved where the rectangular velocity components of the participating robots are subject to the constraints while avoiding inter-robots collisions. The overall research thus contributes by proposing a dynamically evolving environment-aware framework to perform disparate tasks by the robots in a resilient fashion. Since the proposed research focuses on robots with first-order dynamical models operating in two-dimensional scenarios, a heterogeneous team of robots including wheeled and aerial robots with higher-order dynamics, can be considered for future research.

Contents

List of Figures

List of Tables

Abbreviations

ACC	Adaptive Cruise Control
CBF	Control Barrier Function
CLF	Control Lyapunov Function
JPL	Jet Propulsion Lab
KKT	Karush-Kuhn-Tucker
MRS	Multi Robot System
MRTA	Multi Robot Task Accomplishment
QP	Quadratic Program

Symbols

t	Time
r_i	Robot i
C_i	Specialty matrix of the robot i
c_{ij}	Specialty component of robot i to perform task j
T_j	Task j
$\mathbb{R}$	Real numbers
x	State of the system
$\dot{x}$	Derivative of state of the system
f_v	Drift term
g_v	Actuator term
x_e	Equilibrium point of state x
u	Control input
V	Continuously differentiable function
λ	Tunable parameter for function V
L_{f_v}	Lie derivative in the direction of vector field f_v
L_{g_v}	Lie derivative in the direction of vector field g_v
H_1	Hessian matrix for the adaptive cruise control problem
F_1	Real value vector for the adaptive cruise control problem
A_1	Real matrix of linear inequality constraints of the adaptive cruise control problem
b_1	Real vector of linear inequality constraints of the adaptive cruise control problem
f_1	Control barrier function against ACC safety objective
f_2	Control barrier function against formation control
S	Zero superlevel set of function f
S_1	Zero superlevel set of function f_1

S_2	Zero superlevel set of function f_2
S_3	Zero superlevel set of function o
S_4	Zero superlevel set of a function c_2
α	Extended class K function against f_1
δ	Slack variable
d_{eff}	Effective distance
d_{br}	Braking distance
m	Mass of the ego vehicle
v	Velocity of the ego vehicle
v_o	Velocity of the front vehicle
F_w	Wheel force
F_r	Rolling resistance
z	Distance between vehicles
p	Position of the ego vehicle
c_d	Deceleration factor
c_a	Acceleration factor
g	Acceleration due to gravity
$v_{desired}$	Desired velocity
T_h	Lookahead time
q	State of the robot
γ	Extended class K function against f_2
$\mathbb{R}_-$	Negative real numbers
$\mathbb{R}_+$	Positive real numbers
u_{min}	Velocity vector for minimum rectangular component of velocity
u_{max}	Velocity vector for max rectangular component of velocity
u_{xmax}	Max rectangular component of velocity in x direction
u_{xmin}	Min rectangular component of velocity in x direction
u_{ymax}	Max rectangular component of velocity in y direction
u_{ymin}	Min rectangular component of velocity in y direction
$\hat{u}$	Optimal value for the control input
$\hat{u}_{max}$	Maximum optimal control input
$\hat{u}_{min}$	Minimum optimal control input

$\hat{q}$	Stationary point of state q
L	Lagrange multiplier
J_1	Cost function for ACC problem
J_2	Cost function for formation control
Φ_i	Neighborhood of the robot i
$g(v, e)$	Graph g with vertex v and edge e
N	Total no. of vertices
q_i	Position of the robot i
q_j	Position of the robot j
d_{ij}	Distance between robot i and j
n_r	Total no. of robots in the multi robot system
v	Vertex vector of the graph
v_i	Vertex i
v_j	Vertex j
v_n	Vertex n
e	Edge of the graph
e_{ij}	Edge between vertex i and j
$\deg(v_i)$	Degree vector of vertex i
$\mathrm{Deg}(g)$	Degree matrix of graph g
$A(g)$	Adjacency matrix of graph g
$\mathrm{Lap}(g)$	Laplacian of the graph g
Q	Ensemble state of the robots
eig	Eigenvalues vector
ϑ	Agreed-upon state value for all agents in a multi robot system
u_i	Control input for the robot i
f_i	One-to-one function between robot i and near-by robot j
J_i	Pairwise cost between robot i and near-by jth robot
u_{ix}	Component of control input u to robot i in x axis
u_{iy}	Component of control input u to robot i in y axis
c_1	Control barrier function for collision cost (formation control)
c_{1i}	Pairwise collision avoidance cost of ith robot against c_1
β	Extended class K function against c_1

K_{1ij}	Pairwise super-level set of function c_{1i} w.r.t jth robot
d_c	Minimum threshold of distance between robots i and j
d_g	Diagonal distance between robots i and j
T	Multi-robotic tasks
n_t	Total multi-robotic tasks
d_n	Destination for the task n
σ	Slack variables vector
σ_i	Slack variable for the robot i
$\sigma_{i,n}$	Slack variable for the robot i against the task n
$\sigma_{i,m}$	Slack variable for the robot i against the task m
σ_{max}	Maximum value for the slack variable
f_n	One-to-one function between the robot i and the destination d_n
τ	Scaling factor
ψ	Current task specification
ψ_i	Task specification for robot i
ψ^d	Desired task specification
ψ'	Task re-issuance vector
ψ_n^θ	Task issuance status
ψ^*	Environment-aware current task specification
ψ_i^*	Environment-aware task specification for the robot i
ρ	Extended class K function against function f
η	Extended class K function against function o
k	Extended class K function against function c_2
ζ	Scaling factor
ϕ	Vector of task preference for a multi robot system
ϕ_i	Vector of task preference for robot i
$\phi_{i,n}$	Task preference for the robot i against task n
$\phi_{i,m}$	Task preference for the robot i against task m
ψ_n^θ	Fraction of robots to perform task n
C_i^p	Penrose inverse of C_i
$c_{i,n}$	Specialty of the robot i to perform the task n
$c_{i,m}$	Specialty of the robot i to perform the task m

E	Environment module
e_i	Environment coefficients vector for the robot i
$e_{i,n}$	Environment variable associated with the robot i against the task n
C_i^E	Environment-aware specialty matrix of robot i
c_{ij}^E	Environment-aware specialty component of C_i^E against task j
a	Rate of depreciation or appreciation
n_o	Maximum no. of obstacles
d_o	Minimum safe distance between robot and near-by obstacle
c_2	Control barrier function for collision avoidance
c_{2i}	Pairwise collision avoidance cost against c_2
K_{2ij}	Zero super-level set for CBF c_2
o	Control barrier function for obstacle avoidance
K_{io}	Zero super-level set for CBF o
H_2	Hessian matrix for formation control problem
F_2	Real value vector for formation control problem
A_2	Real matrix for linear inequality constraints of formation control problem
b_2	Real vector for linear inequality constraints of formation control problem
H_3	Hessian matrix for formation control problem with collision avoidance
F_3	Real value vector for collision-free formation control
A_3	Real matrix for linear inequality constraints of collision-free formation control
b_3	Real vector for linear inequality constraints of collision-free formation control
A_c	Real matrix for linear inequality constraints of collision avoidance
b_c	Real vector for linear inequality constraints of collision avoidance
H	Hessian matrix for the resilient problem
F	Real value vector for the resilient problem
A	Real matrix for linear inequality constraints of the resilient problem
b	Real vector for linear inequality constraints of the resilient problem
A_{eq}	Real matrix for linear equality constraints of the resilient problem
b_{eq}	Real vector for linear equality constraints of the resilient problem
w_r	Width of the robot
w_o	Width of the obstacle

Chapter 1

Introduction

During the preceding decade, multi robot systems have left secluded laboratories and are now being robustly arrayed and operated across numerous areas such as warehouse-operations, precision agriculture, search-&-salvage missions and area monitoring-&-survey. The vital motivation that more robots are desirable in these kinds of fields is because there is a great power in numbers. By utilizing a large quantity of robots, redundancy spontaneously comes into the system. If one robot malfunctions, there are quite a number of functioning robots available to endure the requirements for mission tasks: a property defined as swarm-resilience. A broader spatial zone can likewise be covered more resourcefully if more robots are operationally positioned and robots heterogeneous capabilities can be disseminated across a robot team to conduct the operational tasks. These tasks can be performed by specific robots suitable for particular situation and selected out of other robots in team: a property known as swarm-adaptivity. Cooperation, control and communication among robots in team are topics of vital importance for successful accomplishment of tasks. In reaction to these technological and application driven scenarios, various control-&-coordination schemes are offered for unifying the robots in a team to empower the robots to resolve team tasks using locally defined interaction rules. We can say that multi robot system comprises of robots in a team such that each one of these robots is specialized for a specific task and together they work to carry-out a coordinated task.

The multi robot systems can be employed in robotics, automation and artificial intelligence to increase the overall capabilities and efficiency through scalability [1]. As an example, the Maritime and Multi-Agent Autonomy (347N) from the Jet Propulsion Lab (JPL) is at the forefront of swarm autonomy technology development and maturation while working with industry and academia to expand the technology. The Fig. 1.1 shows the schematics of a control architecture for the multi robot command and sensing which enables a team of autonomous surface vessels to collaboratively patrol a harbor [2]. There are certain key aspects

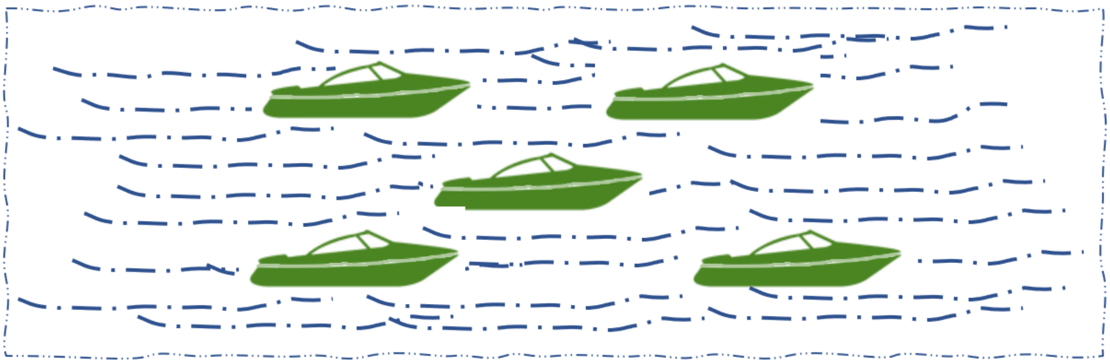


FIGURE 1.1: Schematics of the control architecture for multi robot command and sensing enables a team of autonomous surface vessels to collaboratively patrol a harbor [2].

and considerations associated with multi robot systems such as cooperation, coordination, communication and task allocation. The cooperation in a multi robot system is the ability of the system to collaborate and share information to achieve a common objective [3]. The cooperation can include task allocation, information-sharing and mutual support. Effective communication between robots is essential for coordination which can be achieved through wireless sensor networks or other means and using these sensors robots can inform about their environment to perform collective collaboration [4]. Various mechanisms can be employed to avoid collisions between robots and obstacles in their environment while they move such as employing barrier certificates for safety using control barrier functions as used by Pickem *et al.* [5]. The robots in a multi robot system may have different capabilities or specialized skills empowering them to be resilient to individual robot failures while the system should be able to adapt and continue functioning even if some robots encounter issues [6]. Thus multi robot systems have the capability to influence many applications of real world as mentioned earlier such as search and rescue operations, agriculture, warehouse automation, surveillance and environmental monitoring [7]. The researchers and engineers continually explore new algorithms and technologies to improve the performance of multi robot systems across various domains.

1.1 Motivation and Background

As we know that multi robot systems represent a burgeoning area of research focused on bio-inspired collaborative control methods, where a multitude of robots are coordinated in a distributed and decentralized manner. Thus overall the field serves as a pivotal platform for aspiring researchers to delve into and exchange novel insights, enabling them to explore their novel concepts through both analytical and heuristic approaches [8]. The team of robots in a multi robot system works together to achieve desired common objective across diverse fields and thus recent advancements in multi robot systems have proven themselves more beneficial for their numerous real-world applications [9]. As mentioned earlier, robots in a multi robot system cooperate with each other to attain a common goal but at an individual level these robots are also capable to carry out certain tasks. As this team of robots is gradually moving from research laboratories to unstructured and dynamically evolving environments, new challenges like environment-aware resilience are arising from novel unforeseen environment scenarios. The situation is quite pertinent when these robots are required to assign the tasks in practical scenarios for long-term applications where properties of resilience and safety in terms of obstacles/ inter-robots collisions avoidance are essentially needed.
The resilience and safety based frameworks are mentioned in detail in [6, 10–12] however the task allocation and execution in an environment-aware multi robot system hold the prime importance with desired objectives of achieving resilience. Majority of these researches are only robot centric and dynamically evolving environment has not been accounted-for in the introduced frameworks. Thus the requirement of research in the field of environment-aware multi robot system based task allocation and execution has become a vital problem because of the increasing deployment of multi robot based system in a quite dynamic and unexpected/

unknown outdoor circumstances [13, 14].

The design and development of multi robot based task allocation and execution algorithms involve many intricate challenges that the team of robots is ought to face in a dynamically evolving environment. To prepare robots for operation in an environment, the energy constraint has also to be kept in mind while designing of environment-aware algorithms. For example many diverse practical applications involve robots with very limited energy to work for longer duration of time thus necessitating the design of survivability based algorithms for task allocation and execution [15, 16]. In these circumstances, the heterogeneity of robots in terms of different features has also got the attention of researchers. The heterogeneity of robots is based on numerous types of sensors, relevant-actuators and devices for inter-communication thus inclusion of such heterogeneity in multi robots task allocation and execution algorithm can make the robots to present themselves for wider range of tasks [17–19]. The property of robots' resilience in terms of allocation and execution of tasks can be attained by utilizing the specialties presented by robots such as carrying-capacity, wheels and actuators. Thus heterogeneity can grant another worthwhile quality of a multi robot task allocation and execution framework that is the property of resilience. Resilience can be described as capability of an allocation and execution algorithm to respond to specialty-failures on the robots under non-idealities in the operating environment [20]. Consider an example in which a heterogeneously featured multi robot system comprising of wheeled robots, is primarily assigned the task of transporting things to some known destinations in a field. The wheeled robots have numerous specialties such as some of them are traction enabled to cross muddy areas, some are equipped with camera to surveil and few have ultrasound sensors to detect objects. There can be one scenario in which unpredicted environment changes such as weather conditions like intense rains can turn the ground into muddy place thus preventing the non-traction robots for making more progress towards the attainment of an assigned goal or accomplishment of the task like transporting of certain things to a certain destination. In such a scenario the diversification in the specialties possessed by the robots can be benefited via an environment-aware reassignment of the assigned task that is the task of the transportation of things to the traction enabled wheeled robots. Such a diversification in specialties of robots is termed as heterogeneity. Here it can be seen that in these kinds of scenarios, the issue of allocating tasks to different robots in a team is inseparably connected to the conduct of assigned tasks by the member of robot team. This kind of situation is quite obvious when we need to deploy environment-aware multi robot systems for longer duration where dynamically evolving surrounding scenarios can severely influence the allocation/assignment of a task and its subsequent execution.

This thesis research puts forward a solid framework for multi robot system which incorporates the environment based dynamically evolving changes by making the current frameworks of [6, 16] resilient to frequent changes in environment. Thus the research shifts the existing robot centric frameworks of [6, 16] to environment based framework while keeping in view the safety aspects. Safety constraints like collisions and obstacles avoidance are employed using control barrier functions for preventing the robots from going into unsafe states or sets. Heterogeneity of the robots is utilized for showcasing the suitability that robots have for performing numerous assigned tasks. The newly designed and developed environment-aware resilient framework in this thesis has utilized constraint driven optimization based technique whose ultimate solution at every point in time has produced an environment-aware tasks allocation to robots through a pre-designed tasks-prioritization based scheme. The technique generates control input for each robot in the team and also ensures the accomplishment of the assigned tasks optimally. Current techniques of tasks allocation and execution primarily characterize both robots & assigned tasks with regards to the robot-specialties without explicitly taking environment model in account [19, 21].

The proposed multi robot task allocation & execution technique in this thesis has showcased its resilient performance in a dynamically evolving environment. It represents example scenarios manifesting the resilience based allocation and execution of numerous assigned-tasks in environment-aware fashion keeping in view the safety constraints. Following the basic work on the multi robot task allocation and execution as explored in [6, 16, 22, 23] , in this thesis a constraint driven approach in which the accomplishment of assigned-tasks is described through an optimization based mathematical problem, is used. The scheme manifests both the utilization of control barrier functions that robots have during the conduct of assigned-tasks and pliability in situations where the surrounding circumstances of the robots may change [23–26]. The non-idealities such as dynamically evolving environment and specialties failures on the robots are effectively incorporated in our designed framework thus enabling environment-aware resilient tasks allocation and execution behaviors while keeping constrained conditions of collision and obstacle avoidance in particular focus. The schematics of a test-bed for a multi robot system is shown in the Fig. 1.2 [27].

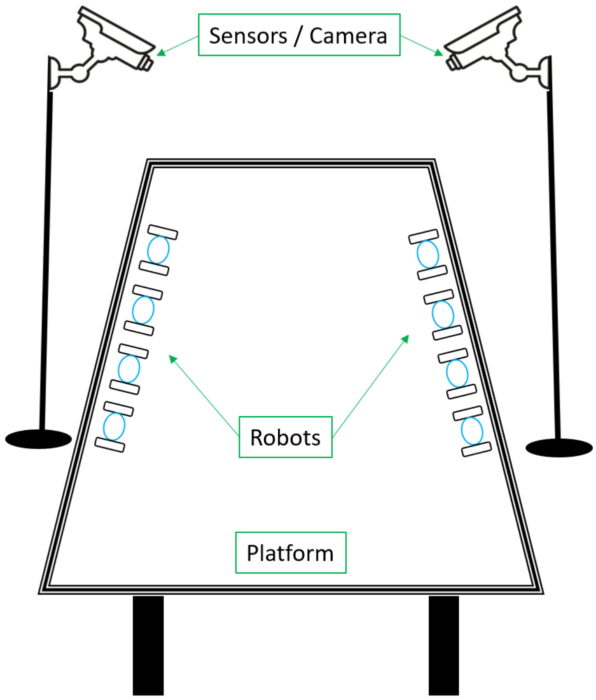


FIGURE 1.2: Schematics of a test-bed for a multi robot system consisting of eight wheeled-robots, two sensors and a platform [27].

1.2 Research Objectives

Research on an environment-aware resilient control of a multi robot system is a challenge in the broad subject of multi robot systems. The objectives of this research are as described below:

- To present a comprehensive literature review on multi robot tasks.
- To design an environment-aware multi robot tasks' resilience based control scheme capable to work in a dynamically changing environments and further verification of this designed scheme through simulations in Matlab©.

- To manifest the safety guarantees of the system, including collision prevention and obstacle avoidance, using CBF-based controllers.

1.3 Thesis Organization

In this chapter 1, a brief introduction, motivation with general problem background and the research objectives of this dissertation are discussed. A brief literature survey highlighting the importance of the multi robot system is presented. Detailed research objectives of the thesis are also presented. The subsequent chapters of the thesis are organized as follows:

- **Chapter 2**: In chapter 2, detailed literature survey highlighting importance of past efforts done by numerous researchers are discussed. Extensive literature survey paved the way towards the development of problem statement of this thesis. The contributions and methodology of research revolving around thesis problem statement are also discussed.

- **Chapter 3**: In chapter 3, the control barrier function based control design is presented for a specific scenario of adaptive cruise control. It also showcases simulation results of control barrier function based control design of an adaptive cruise control among ground vehicles.

- **Chapter 4**: In chapter 4, the optimization technique based decentralized controllers are presented through developing a multi robot task controller encompassing real-world unique scenario of multi robot formation control under confined conditions. It also presents simulation results for the multi robot task controller developed for multi robot formation control.

- **Chapter 5**: In chapter 5, the environment-aware resilient multi robot task accomplishment framework required to solve thesis problem statement is discussed with the specific scenario of dynamic task accomplishment under optimal strategy in constrained conditions. Finally, simulation results for environment-aware dynamic task accomplishment under optimal strategy in constrained conditions are presented and discussed.

- **Chapter 6**: Chapter 6, concludes the dissertation through outlining the main contributions and list of directions that can be accomplished in the future research.

Chapter 2

Literature Review, Gap Analysis and Problem Statement

2.1 Introduction

In this chapter, the relevant literature on the environment-aware resilient task allocation and execution by a multi robot system is presented. The environment-aware resilient task allocation and execution framework focuses on some special situations where tasks essentially need coordination in robots working in a team under a dynamically evolving environment. Environment-aware resilient and adaptive task allocation and execution is primarily a multi robot tasks based scheme. For an elaborated survey on general class of task allocation problems, the comprehensive literature discussed in [13, 14, 28] highlights the desired information. The work by L.E. Parker [17] has nurtured a framework for designating robots to desired team based tasks by shifting between variety of predetermined behavioral processes keeping in view the robot heterogeneity. To handle the challenges of computation based intricacy linked with such an assignment based techniques, many different methodologies are suggested in several research articles where robots can distribute tasks to other robots through bidding and auction based protocols [29–32]. In such situations where a large quantity of robots with few capabilities or abilities are there, stochastic techniques are proposed in literature in which assignment of tasks is carried out using population based distributions [33–35].

As multi robot based tasks become more intricate and diverse, robots are now considered to hold specific roles in a team of robots, thus entailing further depiction of robot's diverseness in the multi robot system [36, 37]. In the research articles of Prorok *et al.* [19], and Ravichandar *et al.* [38], the researchers have defined a broader class of robot types where each robot in team is endowed with a special capability and further developed an optimization problem oriented framework in order to assign required capabilities presented by robots for performing the tasks. In comparison to this aforementioned literature, the framework mentioned by Notomista *et al.*, has described an approach in two separate research articles [6, 16]. This framework models that how different features presented by the team of robots can equip them with specific abilities that are needed to carry-out an assigned task [6, 16]. Moreover this framework carries the manifold advantage by providing the task allocation and accomplishment framework with a required level of resilience. Infact properties of swarm-resilience or swarm-adaptivity are well researched topics in the area of assigned-task allocation and its execution or accomplishment in a multi robot system [18, 39–41].

In the aforementioned research, the adaptivity in swarms is achieved through the dynamic proclivity of robots to take part in diverse tasks built upon well established cost functions which primarily incorporate the efficacies of the robots at executing the assigned tasks [42]. Yuji Nomura [43] has proposed sensor fusion approach for environment-awareness as the advancements in perception technology for images and voice are enabling robots to acquire enhanced environmental awareness and providing opportunities to exploit their use within self-driving cars and drones. These established frameworks have not considered and further incorporated dynamically evolving environment as an integral part of the frameworks whereas in a real world scenario the changing environment can affect mission critical tasks. The peculiar changing environment conditions can influence robots in terms of unplanned deterioration in the abilities of the robots at performing assigned tasks thus affecting the operational functions of the robots. The problem raise the need of a high quality resilience in a multi robot system required under a dynamically evolving environment as until now the resilience has only been investigated in the purview of coordinated tasks along with resource-availability in such systems where there is heterogeneity [6, 20, 44]. The literature survey on multi robot system depicts that research on multi robot coordination techniques primarily aims on the design of different coordination laws with acceptable characteristics like attaining and maintaining formation-shapes, coverage of fields and tracking boundaries of areas [45, 46].

The factor of safety in multi robot system also holds due importance that what is realistically deployed on a team of robots should be safe such that robots collisions are escaped which usually require another low-level operating of collisions prevention controller that should be capable of taking-over of the robots as soon as these robots come too close to other robots in the team [47]. The outcome of this controller design is actually an amalgamation of the carefully crafted algorithms in connection with the best designed collisions prevention controller. But with the increase in quantity of robots, the robot density also grows which results into triggering of the designed collisions prevention controllers and thus they start influencing the overall behavior of the robots in team highlighting that the expected universal characteristics can not be confirmed anymore [48]. To solve such kind of problem in terms of the evading of the collisions, the solution can be provided by making the evasion of collisions as an intrinsic component of the coordinated-control strategy. Nonetheless such design approach can remarkably enhance intricacy at design stage and thus more precisely it can no longer make numerous suggested design tools as the appropriate tools for designing the required control strategy [45, 46, 49]. Plausible solution of such an issue is to adopt the control strategy by not taking into account the robots collisions and we can then further affirm that the designed safety oriented controllers are negligibly-intrusive primarily through a way that these controllers do nothing until robots collisions are about to happen [50]. Such notion is described by Tomlin *et al.* [51], for aircraft pairs and based on an optimal-control law which only comes into action when the aircrafts required to change from their current method of functions to an evading maneuver for avoiding the aircraft-collisions. Although it presents an auspicious novel design but the computing costs affiliated with on-the-fly solution of an entire Hamilton-Jacobi equations sharply became restrictive when scaling to large number of agents [51]. Furthermore such notion did not prove to be applicable for two airplanes even and modified sought solution urged towards calculating a pre-computed evading maneuver that is being stored in memory and consequently

is deployed by the aircrafts ultimately resulted as less efficient technique [51].
The aim of this research effort also accounts for the development of such controllers that should obey the coordinated-control-laws to such an extent that guarantee the collisions prevention behavior. The primary technique for developing such safety oriented controllers is to use the notion of control-barrier functions to let the robots in a team avoid from getting into unsafe sets - clearly in a minimally-intrusive way by using optimization technique oriented controllers [50]. Establishing on the idea of safety functions suggested in [52] and using the technique of control barrier function as introduced by the Ames *et al.* [53], such safety constraints can be designed which can avoid collision. Based on this notion a controller for coordinated-control can then be designed through a quadratic-program technique and resulting control law has constraints rendered by the safety functions which dictate collisions prevention behavior. The designed safety algorithm can be implemented on a team of robots.
The aforementioned use of control barrier function can be extended from prevention of collisions and further incorporated in framework presented in [6] for multi robot tasks accomplishment. This work of [6] is established on research works presented in [16, 22, 23]. In [6], an approach for the attainment of a task allocation and its accomplishment framework is researched in detail and overall the introduced approach is resilient in terms of tasks execution. Different contributions of this framework are, first the feature and capability oriented modeling for heterogeneous robots permitting pliability in assigning the tasks, second an optimization-problem oriented framework development for tasks execution permitting the robots to perform given tasks thus catering for the diverse features and capabilities showcased by robots and third an energy based tasks assignment framework, to achieve sustainability for prolonged applications. We know that in real world practical applications, the frameworks of [6, 16] are vulnerable to dynamically changing environment because environment has not been considered in these robot centric frameworks. However dynamically evolving environment such as weather/ terrain conditions changing on fly, capability specific destinations and emergence of obstacles en-route, can seriously hamper the desired resilience of the frameworks of [6, 16] to accomplish the assigned tasks. Keeping such situations in considerations, we require to carry out our research on making such frameworks

which are capable to encounter and react a changing surrounding environment in a resilient way. In this regard, a step-wise approach is required to achieve the environment-aware resilience in a multi robot system. Extensive use of control barrier function as mentioned in preceding paragraphs merits the learning of control barrier function based design, a corner-stone for the establishment of an environment-aware resilient framework.

Sequel to the details as mentioned in preceding paragraphs, first the literature survey regarding application of control barrier function based control design is showcased in this section for achieving the control design of achieving an adaptive cruise control in ground vehicles under weather based surface conditions. Next, in order to explore the development of controllers for a multi robot system to accomplish the tasks using control barrier functions, the comprehensive literature for investigating the past efforts on optimization technique based decentralized controllers is presented here as well. This extensive literature survey explores the optimization technique based decentralized controllers for designing a control strategy for the real-world scenario of constraint-oriented multi robot formation control in leaderless consensus under confined conditions. Finally, the past research efforts in terms of literature survey on the topic of multi robot task allocation and accomplishment is mentioned in detail in order to explore the adopted control strategies. This extensive literature survey helped to achieve an environment-aware resilience for a multi robot system under constrained conditions of obstacles and collisions avoidance in an optimal fashion. Thus substantial literature survey on control barrier functions and multi robot task allocation/accomplishment frameworks steered the research of this thesis towards establishing an environment-aware resilient framework by proposing a real-world scenario of dynamic tasking in multi robot system under optimal strategy in constrained conditions.

2.2 Literature Review

Sequel to the brief literature review as presented in the previous section 2.1, here the detailed literature review is presented in three fold i.e., (i) Control barrier function based control design - adaptive cruise control problem, (*ii*) Multi robot tasks accomplishment - formation control problem and (*iii*) Environment-aware resilience of multi robot tasks - dynamic tasking problem.

2.2.1 Control Barrier Function based Control Design: Adaptive Cruise Control Problem

As mentioned in previous section 2.1 of this chapter, in this section the literature survey for learning the utilization and implementation of control barrier function is presented in detail. The learning lead us to implement the control barrier function in Matlab© through its practical utilization in real-world scenario of adaptive cruise control. For this the problem of adaptive cruise control among two vehicles on a road is taken. However, it is observed that the unwanted road surface condition due to different weathers creates a hazard for the safety of vehicles and is permanently considered as major area of research in engineering all over the world [54]. It is also observed from the statistics of fatalities that adverse weather conditions cause significant effect on road surface. Primarily, weather effect on braking distances is usually neglected while modeling a system however same requires attention from the very start of research [54]. The weather effect gives challenge to adaptive cruise control in a vehicle since different weathers can cause road surface to be wet, slushy or covered with soft/compact snow. Usually, the adaptive cruise control installed on commuter automobiles assists in improving the driving ease and safety [55–58]. Thus adaptive cruise control problem can be taken as a typical control issue because it includes goals like driving at a chosen speed with safety issue of sustaining a definite distance from the leading vehicle. The different physical features of the vehicle and surface conditions also impact the vehicle greatly. Such kind of control problem becomes more interesting when different interacting constraints start violating each other that is a situation can occur when the velocity of ego vehicle becomes faster than the velocity of the front car. In such scenarios, safety constraint takes superiority and for this various solutions are also presented in literature [59]. The model predictive control based technique is used by the Musa *et al.* [60], for adaptive cruise control and weather effect is also discussed.

The adaptive cruise control problem discussed in this thesis extends the research of [53] by taking into account the weather effect as described by the Kordani *et al.* [54], and thus surface conditions due to weather effect are made part of constraint known as control barrier function. The control barrier function is discussed in detail which is associated with a set. The barrier function goes to zero as any state inside barrier function moves towards the boundary of that set, thus forward set invariance property can be utilized under certain conditions [50]. The unification of control barrier function with control Lyapunov function helps to achieve different control objectives simultaneously [53]. The control Lyapunov function based constraint helps to achieve a control objective by constructing a controller that stabilizes the given system and in this aspect control Lyapunov function is described in-detail in numerous research articles [61, 62].

This thesis research aims to propose the novel notion for designing a control technique for an ego vehicle to achieve adaptive cruise control on different surface conditions due to weather effect. The technique combines the control barrier function and control Lyapunov function through a Quadratic Program. The safety constraint is a control barrier function with surface conditions taken into account. In the next chapter 3, the control design for the above mentioned notion of adaptive cruise control under weather based surface conditions is presented with mathematical formulations. In the following section, the literature survey regarding multi robot task accomplishment using control barrier functions in a decentralized control manner is discussed in detail. This helped to achieve formation control in a multi robot system using control barrier functions.

2.2.2 Multi Robot Tasks Accomplishment: Formation Control Problem

In the previous section 2.2.1 of this chapter, the literature survey for establishing the notion of CBF based control design for adaptive cruise control among ground vehicles is presented. In this section literature survey for the development of optimization based decentralized controllers using control barrier functions is described in detail. The comprehensive literature survey aims to investigate the real-world scenario of multi robot task allocation which helped to construct the basis of practical scenario of multi robot formation control. The extensive literature review helped to identify the research gap and paved the way to design the controller for the novel notion of constraint-oriented formation control of multi robot system in leaderless consensus under confined conditions. The formation control of robots in a team is successfully achieved under different constraints. In the first case the constraints are applied on the rectangular components of robots velocities. In the second case the constraint of inter-robots collisions avoidance is incorporated in devised control strategy of first case.

The coordination in a multi robot system is essential to ensure that robots work together to perform numerous tasks in a seamless manner and the allocation of these tasks in the multi robot system can be centralized or decentralized [63]. The centralized controller assigns different tasks to each robot in a multi robot system whereas decentralized controllers let the robots to use local information [64]. The formation control is a specific problem of coordination control and during the formation control of multiple robots, enforcing constraints on the velocity of robots in a multi robot system becomes a crucial aspect that influences the overall performance, safety and efficiency of the system. Striking the right balance between safety, efficiency and adaptability is essential for the successful deployment of such a system across various applications [65–67]. One primary reason for setting the constraints on the velocity is safety. Robots moving at excessively high speeds can pose a danger to themselves, other robots, humans in the environment or even the equipment they interact with. By imposing velocity constraints, the risks of collisions and accidents are reduced [66]. Imposing velocity constraints is a strategy for promoting energy efficiency as well that is, high-speed movements typically require more energy and in scenarios where robots are powered by batteries, limiting the velocity can extend the operational time [68].

For the collision-avoidance scenarios, developing such controllers with secure demonstrable safety assurances for autonomous systems is also crucial and for this reason, the selection of the term *barrier* is influenced by the optimization literature

[50, 69]. The literature integrates barrier functions into cost functions to circumvent unfavorable areas including robots in the way, using a repelling term in the cost function. This behavior, recognized as collision avoidance or maintenance of a collision-free state, is referred to as system safety [69]. Overall enforcing constraints on velocity components contributes to the optimization of the overall system performance that is, by controlling the speed at which a system can move in each direction, it becomes possible to achieve desired outcomes efficiently. Thus constraints on the rectangular components of velocity are applied when tasks require precise control along specific axes as highlighted in [70] and [71].

In the work by Lin *et al.* [72], constraints on the position and input velocity of participating agents are considered where the constraints on the resultant velocities are from a defined set of quantitative values. The technique in [73, 74] discussed constraints on input including constrained resultant velocities where the velocities are carefully selected absolute values, while Fu *et al.* [74], discussed collision avoidance in a leader-follower topology. The Wang *et al.* [75], demonstrated constraints on the input acceleration along-with the utilization of control barrier functions as part of a collision avoidance strategy. The work by L.N. Tan [76], considers $H - \infty$ based control scheme and higher order dynamics for each robot in a multi robot system. It takes into account robustness of a distributed control framework in large-scale systems under disturbances and the robot actuators with saturation limits, meaning that control inputs like torques or forces are constrained. These limits indirectly lead to constraints on the velocities of the robots and overall obtained research results are promising. While [76] focuses more on actuators' saturation limit and disturbances, the simulation results implicitly manifest that the system's state variables such as position and velocity remain within bounded values, especially because actuator saturation directly impacts velocity. The research by Notomista *et al.* [22], resulted into formulation of a mathematical optimization problem and the desired formation control is achieved using concepts of graph theory and control barrier functions in a constraint-driven approach.

This thesis research on constraint-oriented formation control, discusses the formation control of a multi robot system by introducing constraints on the rectangular components of velocity of each participating robot and further applies the CBF-based inter-robots collision avoidance technique as in [75], thus augmenting the control technique of [22] while achieving the desired leaderless consensus in terms of formulation of formation in a two-dimensional plane. The aforementioned research contributions primarily elaborated on the constraining of the resultant velocity vector; here, the proposed approach in this thesis results in attaining the desired leaderless formation control for a multi robot system under constrained rectangular components of each participating robot's velocity through decentralized-controllers. In these constrained decentralized-controllers the inter-robots collision avoidance constraints are also incorporated. Moreover graph theory concepts are applied at the controller design stage. Thus overall the proposed approach overcomes the shortcomings of previous approaches.

In the chapter 4, the proposed concept is proved by first presenting the control theory and design related with aforementioned novel notion of constraint-oriented formation control of multi robot system in leaderless consensus under confined conditions. In the next section, the literature survey regarding multi robot task allocation and execution framework is discussed in detail. The framework leverages the concepts of optimization based decentralized controllers, control barrier functions and optimal task allocation. Together these concepts can help us to achieve an environment-aware resilient framework. The real-world scenario of proposed notion of environment-aware dynamic tasking in a multi robot system under optimal strategy resolved the problem statement of this dissertation.

2.2.3 Environment-Aware Resilience of Multi Robot Tasks: Dynamic Tasking Problem

In the previous section 2.2.2 the literature survey for the research on constraint-oriented formation control in a multi robot system under constrained conditions - a notion established on control barrier functions based decentralized controllers is presented. In this section the literature survey for establishing the notion of an environment-aware resilience of multi robot tasks using optimization technique oriented decentralized controllers based on control barrier functions is mentioned in detail. The extensive literature survey helps to carryout the research on dynamic tasking for a multi robot system under optimal strategy in order to showcase the property of resilience as briefly discussed in section 2.1 of this chapter. To make the matter more realistic for practical situations being faced by robots, the research regarding incorporation of constrained conditions in terms of obstacles avoidance, inter-robots collisions avoidance and constrained input velocities is also made part of this thesis.

In the realm of multi robot system's research, the task allocation stands as an extensively explored subject [77]. The task allocation strategies are categorized based on their practical applications [78]. Among these strategies, optimization techniques for task allocation are highlighted as particularly significant [79]. The primary objective of the task allocation is to efficiently match suitable robots with specific tasks required by the system [80]. Thus one of the most compelling problems within multi robot system is Multi Robot Task Allocation which primarily involves assigning a set of tasks to a team of mobile robots with the goal of optimizing an objective function, such as minimizing the overall mission time [9]. Another study conducted by Saravanan *et al.*, aims to provide a succinct overview of contemporary dynamic task allocation strategies [81]. It extensively discusses current approaches in dynamic task allocation, focusing particularly on their application scenarios, constraints, objective functions and methods for handling uncertainty. However considering the scenario presented, the multi robot task allocation methods are generally classified into market-based, behavior-based and optimization-based approaches.

It is observed that for the sake of simplicity, many existing studies do not consider important issues such as robots collision avoidance [9]. Various mechanisms can be employed to prevent collisions between robots and obstacles as they navigate their environment. For instance, employing barrier certificates, such as control barrier functions (CBFs), ensure safety during movement [5]. Ensuring collision-free operations is crucial in autonomous systems, requiring controllers with the robust safety assurances. This behavior, known as collision avoidance or maintaining a collision-free state, is essential for system safety [69].

In relation to the allocation and further execution of tasks by a multi robot system, the Wang *et al.*, present a time-varying constraint-driven optimization framework designed for collision-free execution of tasks by autonomous underwater vehicles, leveraging higher-order dynamics [82]. The framework formulates task costs as constraints based on control barrier functions, which are applied to minimize control efforts. However, the concepts of global task specification and current task specification are not addressed. Furthermore, obstacle avoidance is also not covered in the discussion. The Notomista *et al.*, have formulated a mathematical optimization problem and desired task execution has been achieved through a constraint driven approach [22]. Concurrently executing multiple tasks are crucial in numerous robotic applications and sequel to this prioritizing tasks becomes essential in scenarios where some tasks such as safety-critical tasks must take precedence over application-related objectives, ensuring mutual protection between the robot and its surroundings. Moreover, the ability to dynamically adjust task priorities during execution provides the multi robot system with the flexibility to adapt its objectives over time [83]. In certain scenarios in addition to prioritization of tasks, the tasks execution order is also important and the Warnakulasooriya *et al.*, have highlighted that in drone swarm applications, drones are anticipated to fulfill tasks while adhering to the specified task execution order and thus proposes a method for allocating a set of prioritized tasks among drone robots within a swarm using an auction-based algorithm causing swarm drones to follow the given priority [84]. The Lexing *et al.*, introduce a distributed hedonic coalition formation game methods for task allocation involving heterogeneous agents [85]. The method allows self-interested agents to autonomously decide whether to join or leave a coalition based on their preferences. To determine the optimal compositions of these heterogeneous coalitions, the discussed approach enables each agent to prioritize assigned tasks while explicitly considering their utility in performing them. However, in both aforementioned approaches task re-issuance in connection with speciality reassignment as well as task allocation in the presence of obstacles followed by inter-robots collision avoidance remains unexplored.

The Notomista *et al.*, have devised an innovative algorithm to optimally allocate tasks to robots while taking into consideration of their different capabilities and presents a dynamic task allocation algorithm, formulated as an optimization problem which is solved by the robots at each point in time [16]. However, the global task specification remains fixed i.e., tasks are issued prior to the start of the experiment. Furthermore, the author does not address optimal task allocation in scenarios involving multiple obstacles and inter-robot collisions. In a multi robot system, robots may possess varying capabilities or specialized skills, which enhance resilience against individual robot failures. This allows the system to adapt and continue functioning even if some robots encounter issues [6].

The Mayya *et al.*, have discussed an optimization-based formulation to allocate a team of robots to different tasks in a deterministic method and further stated that an assigned task can be completed via multiple likely amalgamations of robot capabilities whereas the proposed framework caters to the capabilities degradation effect by environmental disturbances for achieving the resilience feature [86]. However, the scenario of global re-issuance of the task to a failed/immobilized robot, if its capability improves with time, has not been discussed. Additionally, it does not address multiple obstacle avoidance and inter-robot collision avoidance scenarios under optimal task allocation situation. In another research, Mayya *et al.*, explore the computational paradigm of an optimal task allocation framework and demonstrates the effectiveness of a mixed centralized/decentralized approach [86]. The Emam *et al.*, have developed an adaptive task allocation methodology that dynamically updates the specialization of the participating robots based on their present efficiency at completing the tasks [23]. In further research by the Emam *et al.*, the authors have introduced a task allocation and execution framework for multi robot teams that explicitly accounts for disturbances by augmenting the predefined dynamic model [87]. In both approaches, dynamic global task re-issuance as well as task completion under obstacles and collisions avoidance with optimal allocation technique have not been discussed.

In this thesis research, we have also set constraints on robots velocities as it is observed that most of the literature have not incorporated the notion of velocity constraint whereas imposing constraints on the velocity of robots in a multi robot system is critical as it directly impacts system performance, safety and efficiency. Balancing these factors is key to effectively deploying such systems across diverse applications [65–67]. For tasks requiring precise control along specific axes, constraints on rectangular velocity components are implemented, as mentioned in [70] and [88]. Motivated by the above observations and based on identified research gaps as highlighted in preceding paragraphs, the proposed primary research of this thesis results into attaining the environment-aware tasks resilience through dynamic tasks issuance/withdrawal/re-issuance for the once-issued-tasks to a multi robot system with simultaneous specialty reassignment through task-specialty relationship matrix (here we termed as environment module) in an optimal way. Constrained surrounding conditions in terms of robot-to-obstacles avoidance and robot-to-robot collision avoidance are employed in proposed idea to present a real-world scenario. The proposed idea of environment-aware tasks resilience under constrained conditions is achieved through an optimization-technique based optimal task allocation framework. The framework generates velocity control inputs for each team robot under constrained velocity components, thus the proposed research overcomes numerous shortcomings of previous approaches as discussed in the extensive literature survey. The proposed idea of environment-aware resilience of multi-robotic task is based on the idea of dynamic task re-issuance and reassignment through the environment module. The environment module has environment coefficients directly related with the task issuance strategy and specialties of the robots. Overall the proposed technique imparts such flexibility in a multi robot system that multi robot tasks can be issued/withdrawn online and robots specialties can also be activated/deactivated/re-activated simultaneously under a simulated environment effect. Through environment module, the task issuance/withdrawal/re-issuance and specialty activation/deactivation/re-activation can be made tightly coupled or decoupled depending upon the scenario presented to a multi robot system. In tightly coupled case, the task re-issuance and specialty reassignment has to be a seamless simultaneous operation.

In this thesis, the resilience is manifested in two ways; first, if the specialty of a robot is deactivated through environment coefficient during mission-course, the task is re-issued simultaneously to appropriate robot in team using environment module with activation of corresponding coefficient to perform that task. In the second case, the specialty reassignment is also carried out while keeping task issuance in active mode; thus both approaches provide the resilience of tasks execution/accomplishment. The proposed notion is extended and made more realistic by incorporating the inter-robot collision prevention technique, obstacles avoidance strategy and velocity constraints in the designed environment-module-oriented optimal task-allocation technique. The unification of environment module alongwith task related cost function, multiple obstacles avoidance, inter-robot collision avoidance and velocity constraints is carried out by integrating all constraints through an optimization program. The number of robots and obstacles are also increased during the mission course to check the working of developed notion in case of scalability in terms of increase in tasks with corresponding increase in number of robots. The verification of designed optimization controller is carried out by showcasing numerous results. The proposed work also denotes the functioning of optimal constraint-driven multiple-task issuance under obstacle avoidance, collision avoidance and velocity constraints. To demonstrate the effectiveness of the adopted techniques, which include dynamic task re-issuance, specialty reassignment, obstacles/collisions avoidance and velocity constraints, detailed and conclusive comparisons of results are made in this thesis. These comparisons provide a comprehensive understanding of the proposed decentralized controllers under set of constraints.

The theoretical foundations of the proposed frameworks as developed in chapters 3, 4 and 5 are validated through simulations as well, thus directly address the research gaps identified in this literature review. This systematic progression from problem identification to solution development ensures a coherent research narrative. The control barrier function based design approach is particularly well-suited for bridging the identified gap, as evidenced by its successful application in contemporary research discussed in this section. The framework's effectiveness in addressing the problem statement is further demonstrated through its grounding in state-of-the-art CBF theory and its ability to maintain control guarantees while accommodating dynamic environmental and optimization constraints.

2.3 Gap Analysis

The existing frameworks designed for the tasks completion by the team of robots lack an environment awareness and are primarily robot centric. These frameworks only consider the team of robots and their specialties required for the execution of the assigned multi robot tasks. In real world, practical applications these robot centric frameworks are vulnerable to the dynamically changing environment. The active or inactive status of robots due to on-the-fly changes in an environment can seriously hamper the desired resilience of a multi robot framework to execute the assigned tasks. Under these circumstances, the current frameworks need be enabled to respond to such changing environment through the capability of the environment-aware resilience. The capability of the environment-aware resilience helps to achieve the switching of the assigned tasks among the robots for their continued execution.

2.4 Problem Statement

Based on the gap analysis, the challenge of the research in this thesis lies in the achievement of the multi robot tasks' resilience in the dynamically evolving environment under the conditions of constrained velocities of participating robots while avoiding obstacles and inter-robots collisions.

2.5 Research Methodology

The research methodology for the development of an environment-aware resilient multi robot system is mentioned as follows:

1. Development of an application of control barrier function - An adaptive cruise control problem for ground vehicles under weather effect on road surface.

2. Development of multi robot tasks accomplishment framework - constraint-oriented formation control of multi robot system in leaderless consensus under confined conditions.

3. Development of environment-aware resilient framework for a multi robot tasks allocation and accomplishment - dynamic tasking for a multi robot system under optimal strategy in constrained conditions.

4. Matlab© simulation of developed framework.

2.6 Research Contributions

The work in this thesis shows the conduct of desired research in a sequential manner. The problem of environment-aware resilience in multi robot system under constrained conditions is a special case of multi robot task allocation and execution framework. This intrigues to develop control design for an environment-aware resilient multi robot tasks execution by leveraging on the concept of control barrier function. For this, first a control barrier function based control design for achieving an adaptive cruise control in ground vehicles under different weather effects is presented. Then the optimization technique based decentralized controllers using control barrier functions for multi robot tasks execution is explored. Using this optimization technique, a control for multi robot formation control is developed under confined conditions leveraging control barrier functions encoded as constraints of the designed optimization program. Finally this research work has successfully developed a control design for environment-aware resilient multi robot task allocation and execution framework. The environment-aware resilience is achieved through dynamic tasking for a multi robot system under optimal strategy while avoiding multiple obstacles and inter-robot collisions.
Overall the thesis work demonstrates the effectiveness of the adopted control technique which include control barrier function based adaptive cruise control, decentralized controllers based multi robot tasks execution in formation control problem and finally environment-aware resilience through dynamic task re-issuance, specialty reassignment. Additional constraints of obstacles avoidance, collisions avoidance and velocity bounds are added in the designed optimization techniques for formation control and environment-aware resilient frameworks. Detailed comparisons of results are presented in this thesis to showcase the effectiveness of designed control algorithms. These comparisons provide a comprehensive understanding of the proposed decentralized controllers under set of constraints. Thus the research work in this dissertation has showcased major contributions in the following research paradigms:

1. Control barrier function based control - adaptive cruise control problem

2. Multi robot tasks accomplishment - formation control problem

3. Environment-aware resilience of multi robot tasks - dynamic tasking problem

The details of research contributions under aforementioned research paradigms are described in the following subsections.

2.6.1 Control Barrier Function based Control Design: Adaptive Cruise Control Problem

To find the potential of control barrier functions, here we have considered a real world problem of an adaptive cruise control among ground vehicles using control barrier function. Desired control technique incorporates weather based road surface conditions. Following are the major contributions from the conducted research:

1. Design and development of an adaptive cruise control for ground vehicles by leveraging on the idea of control barrier function.
2. Formulation of an optimization based control technique which unifies the control Lyapunov function and control barrier function through a quadratic program.
3. Proposed technique incorporates the stability and safety considerations in the designed adaptive cruise control algorithm.
4. Incorporation of the effect of weather based road surface conditions (dry, wet, slushy and snowy) in designed control technique.

2.6.2 Multi Robot Tasks Accomplishment: Formation Control Problem

To explore optimization technique based decentralized controllers for multiple tasks execution by the team of robots leveraging control barrier functions, a constraint-oriented formation control for the multi robot system is proposed and simulated in this thesis. The idea primarily addresses a real-world scenario being encountered by a multi robot system while achieving the desired objective where each participating robot can move independently in any direction and their velocities can be finite due to certain limitations. Following are the major contributions from the research on formation control problem:

1. Design and development of an optimization technique based controller for each participating robot in a multi robot formation control setting and further incorporation of constraints on the rectangular components of the robots velocities. The technique achieves a leaderless consensus leading to predefined formation shaping using control barrier functions thus leading to multi robot tasks accomplishment.

2. Proposed model of contribution 1 is extended and made more realistic by considering situations in which inter-robots collisions may occur while performing the desired tasks. This scenario is addressed by unifying the constrained-velocity-components based controller of contribution 1 and the CBF based inter-robots collision avoidance strategy. The suggested control technique is based on the approach of unifying the cost function of the desired objective and collision avoidance, as separate CBF based constraints of a single quadratic program.

3. To prove the efficacy of the devised control technique pivoting across the proposed ideas of velocity components and inter-robots collision avoidance constraints, detailed and conclusive comparisons of the obtained results are drawn in this dissertation.

2.6.3 Environment-Aware Resilience of Multi Robot Tasks: Dynamic Tasking Problem

As already mentioned that environment-aware resilience of multi robot tasks is a special case of the multi robot task allocation and accomplishment framework. It results into development of a control technique capable for dynamic tasks reissuance and reassignment optimally even under constrained conditions. The proposed research on this special topic has resulted into several outcomes which are made part of this thesis. Following are the major contributions from the research conducted:

1. To achieve environment-aware resilience in a multi robot system while carrying out the assigned tasks optimally, novel idea of dynamic task re-issuance and reassignment through environment module is devised and implemented in this thesis. The proposed environment module has environment coefficients directly affecting the tasks issuance strategy and specialties of robots. This imparts such flexibility in a multi robot system that we can issue/withdraw the assigned tasks online while robots specialties can also be activated/deactivated simultaneously, overall providing a resilience in tasks execution by a multi robot system.

2. Through environment module tasks resilience is achieved in two ways, first through task issuance/re-issuance strategy and second through specialty activation/deactivation mechanism. Using environment module, the task re-issuance/withdrawal and specialty activation/deactivation is made tightly coupled or decoupled depending upon the scenario presented to the multi robot system. In tightly coupled case, the task re-issuance and specialty re-assignment has to be a seamless simultaneous operation. If a specialty of the robot is deactivated through environment coefficients during mission-course, the task is re-issued simultaneously to appropriate robot in team using environment module with activation of corresponding coefficient to perform that task. Similarly a specialty re-assignment is also carried out while keeping task issuance in active mode.

3. Proposed ideas at contributions 1 and 2 are extended, made more realistic by incorporating the inter-robot collision prevention technique, obstacles avoidance strategy and velocity constraints in the designed environment-module-oriented optimal task-allocation technique. Unification of environment module, cost function, multiple obstacles avoidance, inter-robot collision avoidance and velocity constraints is carried out by integrating all in an optimization program. The cost function, inter-robots collision avoidance and obstacles avoidance are encoded through control barrier functions.

4. No. of robots and obstacles are increased during the mission course to check the working of developed optimization technique. Verification of designed optimization controller is carried out by showcasing numerous results.

5. Proposed technique of contribution 3 ensures functioning of optimal constraint-driven multiple-task issuance/assignment while avoiding obstacles and inter-robot collisions in the presence of velocity constraints.

2.7 Chapter Summary

After the discussion on the research motivation, background and objectives in chapter 1, in this chapter a thorough review of the past efforts for multi robot control are discussed. The comprehensive literature review is presented sequentially in sections 2.2.1, 2.2.2 and 2.2.3 of this chapter. The section 2.2.1 presents literature review regarding control design using control barrier function through a specific scenario of the attainment of an adaptive cruise control among ground vehicles using control barrier function under weather based surface conditions. The section 2.2.2 presents literature review regarding multi robot tasks accomplishment related techniques. This literature survey helped to explore the research area regarding development of the optimization technique based decentralized controllers using control barrier function. These efforts helped to identify the research gap in the specific area of multi robot task accomplishment and further guided to propose a novel idea of constraint-oriented formation control of a multi robot system in leaderless consensus under confined conditions. The section 2.2.3 showcases detailed literature survey regarding development of multi robot task allocation and accomplishment in an optimal way through optimization technique oriented decentralized controllers. The comprehensive and detailed literature survey helped to create and develop the core problem statement of this dissertation through identification of research gap from this extensive literature review. The problem statement presented in section 2.4 assisted in identifying the areas of concern in the previous studies. Finally, the chapter is concluded by highlighting the objectives of the proposed research. In the chapter 5, control design for solving the problem statement is discussed in detail.

Chapter 3

CBF based Adaptive Cruise Control

3.1 Introduction

In this chapter, the basis of environment-aware resilient multi robot system is presented by first designing a simple control strategy using control barrier function. For designing a control strategy using control barrier function, a special case of adaptive cruise control under weather based surface conditions is considered here. The successful development of desired control algorithm for an adaptive cruise control in ground vehicles using control barrier function under weather based surface conditions paved the way for its implementation to get the research results as intended. As mentioned previously, the notion of control barrier functions is extensively used in multi robot systems. After exploiting the notion of control barrier function through the example of an adaptive cruise control, a control strategy for the development of optimization based decentralized controllers using control barrier function is designed as well. This design helped to achieve the multi robot tasks execution in a constraint-oriented approach. Using this constraint-oriented approach, the control design for the leaderless formation control of a multi robot system with constrained rectangular components of velocities under confined conditions is developed and implemented to prove the efficacy of designed algorithm (chapter 4). After developing the concepts of decentralized optimization controllers using control barrier functions, the control design for an environment-aware resilience of multi robot tasks through the scenario of dynamic tasking for a multi robot system under optimal strategy in constrained conditions is presented in detail in chapter 5.

3.2 Proposed Control Barrier Function based Control Design: Adaptive Cruise Control Problem

In this section, an application of the control barrier function based control design for a practical scenario of adaptive cruise control under weather effect based surface conditions is discussed by presenting a solid research. The objective of this research scenario is to take into account the effect of surface conditions changing due to evolving weather. A design technique is then formulated which combines the concepts of control Lyapunov function and control barrier function through an optimization technique. The technique ensures the stability and safety considerations in an adaptive cruise control setting. For the aforementioned research the surface condition formulae by Kordani *et al.*, are used for different road conditions [54]. Forgoing the research on the adaptive cruise control using control barrier function by Ames *et al.* [53], is studied and further extended by proposing the novel control technique for ground vehicles to achieve adaptive cruise control on different surface conditions occurring due to weather effect. The applied technique utilizes the combination of control barrier function and control Lyapunov function with surface condition taken into account. The presence of a control barrier function ensured avoidance of unsafe set of values (unacceptable states) whereas existence of control Lyapunov function helped to reach a target state. The objectives of stability (attainment of desired state) as well as safety (collision avoidance) through control Lyapunov function and control barrier function respectively act as constraints of a quadratic program. It is observed that the aforementioned control technique is helpful under different surface conditions happening due to effect of the changing weather [89]. Moreover it is proved that simultaneous achievement of control objective (desired target state by control Lyapunov function) and safety objective (allowable states represented by control barrier function) can be met. In next sections, these design constructions are demonstrated through Matlab© based simulations in the context of weather based different surface conditions to achieve adaptive cruise control for wheeled vehicles where the attainment of target velocity is taken as soft constraint and hard constraint is imposed on the attainment of desired safety objective by ego vehicle.

3.2.1 Problem Formulation

The goal here is to take into account the effect of weather based surface conditions by formulating a control technique which unifies the control Lyapunov function and control barrier function. The unification is done through quadratic program. The technique ensures the stability and safety considerations in an adaptive cruise control problem. The weather based surface conditions are considered in the design of desired control barrier function based control technique.

3.2.2 System Model

In this section, the system model, control Lyapunov function, control barrier function and weather based surface conditions are explained in detail. To stimulate these deliberations, discussion on control Lyapunov function is initiated here and further a discourse is adopted that how it can be utilized to produce controller that can impose stability. A detailed analysis on the articulation of control barrier function based controller constructed with surface conditions is presented here. These controllers are integrated collectively in a quadratic program. For the proposed scenario consider a nonlinear system as mentioned in equation (3.1). The system is affine in control input:

$$\dot{x} = f_v(x) + g_v(x)\,u \tag{3.1}$$

Here x is the state of the system, f_v represents drift term and g_v represents the actuator effect term. Both f_v and g_v are Lipschitz continuous in x. Moreover $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$.

3.2.3 Stability Objective: Control Lyapunov Function

The nonlinear control affine system given in equation (3.1) is required to be stabilized exponentially to a point x_e. Consider a function $V(x)$ which is a positive definite and a continuously differentiable function such that $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$. A control law u subject to inequality of expression (3.2) is required to be sought that can drive $V(x)$ to zero:

$$\inf_{u \in U} \dot{V}(x,\ u) \leq -\lambda V(x) \tag{3.2}$$

Here $\dot{V}(x,\ u) = L_{f_v}V(x) + L_{g_v}V(x)u$ where $L_{f_v}V(x),\ L_{g_v}V(x)$ are Lie derivatives. The term λ is a locally Lipschitz extended class K function and is strictly increasing with $\lambda(0) = 0$. The system can be made stable by finding a control input $u = k(x)_{clf}$ that helps to define a stable system as depicted by the Lyapunov function in expression (3.2) which is an exponentially stabilizing control Lyapunov function as given by:

$$\inf_{u \in U} [L_{f_v}V(x) + L_{g_v}V(x)u] \leq -\lambda V(x) \tag{3.3}$$

$$k(x)_{clf} := \{u \in U : L_{f_v}V(x) + L_{g_v}V(x)u \leq -\lambda V(x)\} \tag{3.4}$$

The expression (3.5) is a quadratic program (QP) formulation depicting a CLF constraint linear in u.

$$\begin{aligned} &\operatorname*{argmin}_{u \in U} \frac{1}{2}u^T u \\ &\text{subject to :} \\ &L_{f_v}V(x) + L_{g_v}V(x)u + \lambda V(x) \leq 0 \end{aligned} \tag{3.5}$$

3.2.4 Safety Objective: Control Barrier Function

The stability requires making a system to reach a target state (point or set) whereas safety phenomenon can be understood as avoiding an unsafe set that is, ensuring the forward invariance of a set. Consider a set S_1 (equation (3.6)) which is a zero superlevel set of a function $f_1(x)$. The function $f_1(x)$ is a continuously differentiable function such that $f_1(x) : D \rightarrow \mathbb{R}^n$. Here $D \subset \mathbb{R}^n$ such that

$S_1 \subset D$. The term ∂S depicts boundary of a set S_1.

$$S_1 = \{x \in D \subset \mathbb{R}^n : \; f_1(x) \geq 0\} \tag{3.6}$$

$$\partial S_1 = \{x \in D \subset \mathbb{R}^n : \; f_1(x) = 0\} \tag{3.7}$$

$$\text{interior}(S_1) = \{x \in D \subset \mathbb{R}^n : \; f_1(x) > 0\} \tag{3.8}$$

The set S_1 as defined in equation (3.6) is a safe set. Consider the term u where $u = k(x)$ is a feedback controller and defines a Lipschitz continuous control law. Consider α as an extended class K_∞ function. This function α is strictly increasing with $\alpha(0) = 0$.

$$\sup_{u \in U} \left[L_{f_v} f_1(x) + L_{g_v} f_1(x)u\right] + \alpha(f_1(x)) \geq 0 \tag{3.9}$$

The expression (3.9) is for all $x \in D$. This implies that $f_1(x)$ is a control barrier function and any Lipschitz control law $k(x)_{CBF}$ that justifies the constraint of the expression (3.9) renders the set S_1 safe [50]. The term $k(x)_{CBF}$ is given by as follows:

$$k(x)_{CBF} := \{u \in U : L_{f_v} f_1(x) + L_{g_v} f_1(x)u + \alpha(f_1(x)) \geq 0\} \tag{3.10}$$

The expression (3.11) is a quadratic program (QP) formulation and depicts a CBF constraint linear in u:

$$\begin{aligned} &\underset{u \in U}{\text{argmin}} \; \frac{1}{2} u^T u \\ &\text{subject to :} \\ &L_{f_v} f_1(x) + L_{g_v} f_1(x)u + \alpha(f_1(x)) \geq 0 \end{aligned} \tag{3.11}$$

3.2.5 System Objectives Unification through Quadratic Program

To achieve the stability and safety objectives, Quadratic Program based coupling of the CLF and CBF constructions of expressions (3.5) and (3.11) respectively can be done as shown in expression (3.12) [53]. Such amalgamation of stability objective characterized by CLF is structured to render trajectory inside the safe

TABLE 3.1: Weather based surface conditions.

Surface conditions and d_{eff}				
Dry	Wet	Slush	Soft snow	Compact snow
d_{eff}=d_{br}	d_{eff}=1.7d_{br}	d_{eff}=2d_{br}	d_{eff}=3d_{br}	d_{eff}=4d_{br}

set characterized by control barrier function. The control barrier function renders a system, a guaranteed safety. The control barrier function here ensures collision avoidance between vehicles. The control Lyapunov function ensures stability that is, reaching a target state.

$$
\begin{aligned}
& u^{'} = \underset{u \in U}{\operatorname{argmin}} \ \frac{1}{2} u^T H u + F^T u \qquad (3.12) \\
& \text{subject to :} \\
& L_{f_v} V(x) + L_{g_v} V(x) u + \lambda V(x) \leq \delta \\
& L_{f_v} f_1(x) + L_{g_v} f_1(x) u + \alpha(f_1(x)) \geq 0
\end{aligned}
$$

where H and F are weight matrices and can be selected based upon control input weights. Here δ is slack variable required to make CLF constraint flexible while the set S_1 manifests the characteristics of forward invariance. Term $\dot{V}(x,\ u) = L_{f_v} V(x) + L_{g_v} V(x) u$ where $L_{f_v} V(x),\ L_{g_v} V(x)$ are Lie derivatives. The term λ is an extended class K function and is strictly increasing with $\lambda(0) = 0$.

3.2.6 Weather based Surface Conditions

The unification of CBF and CLF for ground vehicles can be employed in order to achieve the adaptive cruise control on different surface conditions. The weather based surface conditions tabulated in Table 3.1 are taken into account [54]. The Table 3.1 depicts the different equations for calculating distances under braking effect based on surface conditions. The condition is incorporated as hard constraint as implemented through CBF. Here d_{eff} in Table 3.1 denotes the effective distance as function of distance (d_{br}) under braking effect.

3.2.7 Adaptive Cruise Control under Weather based Surface Conditions

This section derives the expression for the adaptive cruise control among two vehicles in a setting depicted in Fig. 3.1 with the purview of control barrier function. The weather based surface conditions of Table 3.1 are used for simulation purpose. Keeping in view the stability and safety requirement of the given problem of adaptive cruise control, the dynamics of the vehicle is defined as given by:

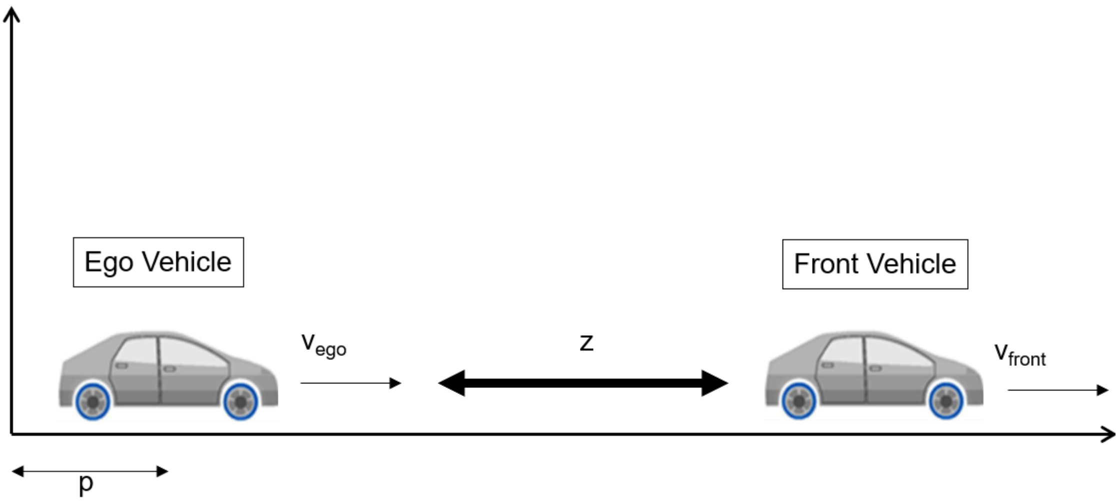


FIGURE 3.1: Front (lead) and ego vehicles in cruise control setting. v_{ego} and v_{front} are velocities of ego and front vehicles. z is distance between vehicles. p is position of ego vehicle.

$$m\frac{dv_{ego}}{dt} = F_w - F_r \tag{3.13}$$

where F_w is the wheel force, F_r is the rolling resistance, m denotes the mass of given vehicle denoted as ego vehicle and v_{ego} is the vehicle's velocity. Here the coefficients F_{r_0}, F_{r_1} and F_{r_2} are determined empirically.

$$F_r(v_{ego}) = F_{r_0} + F_{r_1}v_{ego} + F_{r_2}v_{ego}^2 \tag{3.14}$$

Another vehicle denoted as front (lead) vehicle moving at constant velocity of v_{front} is considered here. Distance between the front (lead) vehicle and ego vehicle is z:

$$\dot{z} = v_{front} - v_{ego} \tag{3.15}$$

Consider a state vector denoted as x and $x = [p \; v_{ego} \; z]^T$ with p as position of vehicle. Dynamics of the given system then becomes as of equation (3.16):

$$\dot{x} = \underbrace{\begin{bmatrix} v \\ -\frac{1}{m}F_r(v) \\ v_{front} - v_{ego} \end{bmatrix}}_{\Longleftrightarrow \atop f_v(x)} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \\ 0 \end{bmatrix}}_{\Longleftrightarrow \atop g_v(x)} u \tag{3.16}$$

where $u = F_w$ is the control input. Weather based surface conditions cause an effect on the distance under braking effect (d_{br}) as mentioned in Table 3.1 that is, $d_{eff} = 1d_{br}$, $1.7d_{br}$, $2d_{br}$, $3d_{br}$ and $4d_{br}$. Values for d_{eff} are made part of hard constraint defined by the standard expression of the control barrier function. Here c_d is deceleration constant and $g = 9.8 \; m/s^2$. The braking distance d_{br} is given by as follows [53]:

$$d_{br} = \frac{1}{2}\left(\frac{(v_{ego} - v_{front})^2}{c_d g}\right) \tag{3.17}$$

3.2.8 Stability Objective as Soft Constraint

In adaptive cruise control when sufficient headway is reached, the goal becomes that ego vehicle with velocity v_{ego} attains the desired velocity $v_{desired}$. Since safety in terms of collision avoidance with front vehicle is more important, the objective to reach desired velocity becomes a soft constraint.

$$v_{ego} - v_{desired} \longrightarrow 0 \tag{3.18}$$

3.2.9 Safety Objective as Hard Constraint

Keeping a safe distance from the front car should be a hard constraint since safety is an important design factor. The following general rule for headway distance to ensure safety can be used [53]. Here T_h is look-ahead time.

$$z \geq T_h v_{ego} \tag{3.19}$$

To avoid acceleration or deceleration more than a certain factor, constraints are required to be imposed on control input u as mentioned below:

$$-u_{min} \leq u \leq u_{max} \tag{3.20}$$

where $u_{max} = mc_a g$ and $u_{min} = -mc_d g$. Here c_a and c_d are acceleration and deceleration factors [53].

3.2.10 Formulation of Adaptive Cruise Control Constraints with Weather based Surface Conditions

In this section, the hard and soft constraints as defined in previous section are enclosed in an optimization based quadratic program. The soft constraint is converted into a control Lyapunov function and hard constraint is converted into a control barrier function. Weather based surface conditions of Table 3.1 are incorporated in the control barrier function designed for the hard constraint. The defined soft and hard constraints are unified for the formulation of control barrier function and control Lyapunov function based quadratic program to achieve desired objective of adaptive cruise control under weather based surface conditions.

a. **Stability Constraint.** The synthesizing of a soft constraint for driving the speed of ego vehicle (v_{ego}) to desired speed that is, the achieving of speed regulation using the control Lyapunov function is required to be carried out here. The stability objective here is to drive the $v_{ego} \longrightarrow v_{desired}$. The soft constraint is rewritten here as given by:

$$\text{Drive} \quad y: \quad v_{ego} - v_{desired} \longrightarrow 0 \tag{3.21}$$

A Lyapunov function should be zero at the equilibrium point x_e and $x_e = [0 \ v_{desired} \ 0]^T$ which is positive everywhere in the region [53]. The candidate Lyapunov function $V(x)_{clf}$ can be used here:

$$V(x)_{clf} = (v_{ego} - v_{desired})^2 \tag{3.22}$$

The general expressions for the control Lyapunov function can be written as given by:

$$\dot{V}(x,\ u)_{clf} + \lambda V(x)_{clf} \leq 0 \tag{3.23}$$

$$L_{f_v}V(x)_{clf} + L_{g_v}V(x)_{clf} + \lambda V(x)_{clf} \leq 0 \tag{3.24}$$

Here $L_{f_v}V(x)_{clf} = \nabla V(x)_{clf}.f_v(x)$ ad $L_{g_v}V(x)_{clf} = \nabla V(x)_{clf}.g_v(x)$. After detailed derivations, the control Lyapunov function expression becomes as given by:

$$(v_{ego} - v_{desired})\ [\frac{2}{m}(u - F_r(v)) + \lambda(v_{ego} - v_{desired})] \leq \delta \tag{3.25}$$

The variable δ is a relaxation factor for keeping the constraint as soft. By setting $\delta = 0$ makes the above mentioned soft constraint as a hard constraint which is not required here. The validity of the control Lyapunov function can be checked by analyzing the two cases such as when $v_{ego} < v_{desired}$ and $v_{ego} > v_{desired}$. In both cases the control Lyapunov function constraint of expression (3.25) remains valid.

b. **Safety Constraint**. The prime objective here is to build a hard constraint for achieving the safety among two vehicles. Here a control barrier function is designed with safety objective of $z \geq T_h v_{ego}$ as already mentioned (expression (3.19)). The intuitive choice for the control barrier function denoted by $f_1(x)_{cbf}$ is given by:

$$f_1(x)_{cbf} = z - T_h v_{ego} \tag{3.26}$$

To verify that expression (3.26) is a valid control barrier function then following expression must be verified:

$$\underset{u \in U}{inf}[L_{f_v}f_1(x)_{cbf} + L_{g_v}f_1(x)_{cbf}u] + \alpha(f_1(x)_{cbf}) \geq 0 \tag{3.27}$$

here $L_{f_v}f_1(x)_{cbf} = \nabla f_1(x)_{cbf}.f_v(x)$ and $L_{g_v}f_1(x)_{cbf} = \nabla f_1(x)_{cbf}.g_v(x)$. Expression (3.27) under maximum deceleration $u = -mc_d g$ is given by:

$$T_h c_d g + v_{front} + \alpha z - (1 + \alpha T_h)v_{ego} \geq 0 \tag{3.28}$$

where the velocity v_{front} is velocity of the front (lead) vehicle. If velocity v_{ego} is bigger in value than the velocity v_{front}, the constraint of expression (3.28) gets invalid making the chosen control barrier function of expression (3.19) as an invalid control barrier function.

The braking distance d_{br} (equation (3.17)) required to decelerate from v_{ego} to v_{front}, is needed to be included in expression (3.19) to achieve a valid control barrier function. The new control barrier function then becomes as described in equation (3.29). Here effective distance (d_{eff}) for dry surface in accordance with Table 3.1 is as $d_{eff}= d_{br}$:

$$f_1(x)_{cbf} = z - T_h v_{ego} - d_{eff} \tag{3.29}$$

As mentioned in Table 3.1, different control barrier functions for surfaces such as dry, wet, slush, soft snow and compact snow can be defined by taking d_{eff} values as $1d_{br}$, $1.7d_{br}$, $2d_{br}$, $3d_{br}$ and $4d_{br}$ in equation (3.29), respectively. To verify that such control barrier function expressions are valid control barrier functions or not, first the control barrier function for dry surface can be taken here:

$$L_{f_v} f_1(x)_{cbf} + L_{g_v} f_1(x)_{cbf} u + \alpha(f_1(x)_{cbf}) \geq 0 \tag{3.30}$$

After necessary derivations the expression (3.30) becomes:

$$\begin{aligned}\left(\frac{T_h}{m} + \frac{v_{ego} - v_{front}}{m c_d g}\right)(F_r - u) + v_{front} - v_{ego} + \alpha z - \alpha T_h v_{ego} - \\ -\frac{1}{2}\left(\frac{(v_{ego} - v_{front})^2}{c_d g}\right) \quad \geq \quad 0\end{aligned} \tag{3.31}$$

Expression (3.31) under maximum deceleration $u = -m c_d g$ is given by:

$$\begin{aligned}\frac{T_h F_r}{m} - \frac{v_{front} F_r}{m c_d g} + T_h c_d g + \alpha z + \left(\frac{F_r}{m c_d g} - \alpha T_h\right) v_{ego} - \\ -\frac{1}{2}\left(\frac{(v_{ego} - v_{front})^2}{c_d g}\right) \quad \geq \quad 0\end{aligned} \tag{3.32}$$

If velocity v_{ego} is bigger in value, the expression (3.32) remains valid. After solving the control barrier functions for wet, slush, soft snow and compact snow, it is found that the control barrier functions for these surface conditions also remain valid numerically as depicted in simulation results in next section.

3.2.11 Formulation of Quadratic Program

The stability and safety constraints as formulated in previous sections can be combined through a quadratic program as mentioned below [53]:

$$u^{'}(x) = \underset{u^* \in U}{\operatorname{argmin}} \quad \frac{1}{2} u^{*T} H_1 u^* + F_1^T u^* \tag{3.33}$$

subject to:

$$A_1 u^* \leq b_1$$

Here

$$A_1 = \begin{bmatrix} L_{g_v} V & -1 \\ -L_{g_v} f_1 & 0 \\ 1 & 0 \\ -1 & 0 \end{bmatrix}, \quad b_1 = \begin{bmatrix} -L_{f_v} V - \lambda V \\ -L_{f_v} f_1 + \alpha(f_1) \\ u_{max} \\ -u_{min} \end{bmatrix} \tag{3.34}$$

and

$$u^* = \begin{bmatrix} u \\ \delta \end{bmatrix} \tag{3.35}$$

- **Cost Function.** The control objective of reaching the desired velocity as mentioned in expression (3.21) can be taken as a cost function $J_1(u)$. Following is the expression from the system in equation (3.13):

$$m\dot{y} = u - F_r \implies \dot{y} = \mu = \frac{1}{m}(u - F_r) \tag{3.36}$$

Cost related with stability objective can be calculated as $\mu^T \mu$. Formulated quadratic program in expression (3.33) gives such value of u which minimizes the cost function $J_1(u)$ as given by:

$$J_1(u) = \mu^T \mu = \frac{1}{m^2}(u^T u - 2u^T F_r + F_r^2) + \delta^2 \tag{3.37}$$

where

$$H_1 = 2 \begin{bmatrix} 1/m^2 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } F_1 = -2 \begin{bmatrix} F_r/m^2 \\ 0 \end{bmatrix} \tag{3.38}$$

The quadratic program mentioned in expression (3.33) can be simulated through a coding scheme and the results can be obtained. After formulation of control barrier function based control design (expression (3.33)) for an adaptive cruise control, in the section 3.3 the detailed discussion on simulation results is presented and elaborated. In the chapter 4, the notion of control barrier function is extended by presenting control design for the decentralized optimization based controllers using control barrier functions. The work lead to construct a multi robot task execution scheme to accomplish the assigned tasks.

3.3 Simulation Results and Discussions

In this section, the simulation results for the control barrier function (CBF) based control design are presented. In the section 3.2, the optimization based control design for achieving an adaptive cruise control is developed. Thus the quadratic program (QP) of equation (3.33) mentioned in section 3.2 can be simulated in Matlab© through a coding scheme. The obtained simulation results are shown in Fig. 3.2. For these simulation results, the parameters mentioned in Table 3.2 are utilized [53]. System defined in equation (3.16) is triggered from initial conditions described as state vector x where $x = (0\ 26\ 120)$. The control barrier function based constraint is incorporated with weather based surface conditions for implying the hard constraint in this adaptive cruise control problem. The existence of control barrier function ensures the forward invariance of the defined set. The control objective in terms of achieving the desired velocity is manifested as soft constraint as constructed using control Lyapunov function (CLF). The soft constraint does not oppose the attainment of control barrier function based hard constraint. Such behavior is proven by depicting the fact in Fig. 3.2 that is when v_{ego} rises exponentially from 26 m/s and it tries to reach the desired velocity $v_{desired} = 30\ m/s$ in 6.7 secs. Meanwhile when the distance z to the front vehicle gets too small and at this point of time the control input u' lessens as control barrier function based

TABLE 3.2: System parameters for ACC setup.

Design parameters	Parameters	
	Value	Units
F_{r0}	0.1	N
F_{r1}	5	Ns/m
F_{r2}	0.25	Ns^2/m
$v_{desired}$	30	m/s
v_{front}	20	m/s
α	5	-
λ	5	-
c_a	0.3	-
c_d	0.3	-
m	1650	kg
g	9.8	m/s^2
T_h	1.8	Sec
p	0	m
v_{ego}	26	m/s
z	120	m

hard constraint comes into action. The hard constraint reduces the speed of the ego vehicle and ultimately it approaches to the speed of front vehicle of 20 m/s ensuring a safe distance of 36 m. The control Lyapunov function constraint is also shown which gets sufficiently relaxed as soon as CBF constraint comes into action and meanwhile relaxing factor that is, the slack variable starts increasing by taking positive values. Finally the designed QP controller (equation (3.33)) justifies all the defined control objectives and constructed constraints. In Fig. 3.3 the results of multiple control barrier functions constructed for weather based surface conditions (Table 3.1) are simultaneously demonstrated through simulations. These simulations depict that for different surface conditions, results of system parameters are satisfactory. The Fig. 3.3 shows that as value of d_{eff} increases, the v_{ego} tries to reach $v_{desired}$ early (3.2 secs for $d_{eff} = 4d_{br}$) due to lessening of frictions as surface condition changes. The control input starts lessening at earlier stage since the CBF and CLF constraints for wet, slush, soft snow and compact snow also start activating at earlier stages. Finally the distance to the front (lead) vehicle also reaches a safe value. It can be seen that as the condition of surface changes, all the system parameters exhibit satisfactory behavior. This implies that the proposed control barrier functions for the different surfaces remain valid and the QP controller achieved through these equations satisfy the desired control objectives including stability and safety objectives. It is pertinent to mention that the selection of the class K function is guided by three key considerations: (1) the chosen class K function must be strictly increasing to ensure a well-defined relationship essential for stability analysis, (2) it must be Lipschitz continuous to guarantee confined sensitivity to input variations, which is crucial for avoiding abrupt changes that could lead to numerical instabilities, (3) empirical validations through multiple simulation trials are conducted to rigorously assess performance under varying conditions, ensuring dependability across different operational scenarios, and (4) the function must maintain quadratic program (QP) feasibility, meaning it should not introduce surplus constraints that could render the optimization problem unsolvable. Our fixed choice of the class K function was carefully selected to satisfy these criteria, and it sufficed for proof-of-concept validation. Beyond theoretical compliance, this choice demonstrated consistent feasibility across all test scenarios, successfully meeting all imposed constraints without exceptions. This robustness is particularly important in real-world control applications, where infeasible solutions or constraint violations could lead to system failures or degraded performance. Furthermore, the Lipschitz continuity condition ensures that the function does not exhibit excessively steep gradients, which could otherwise increase noise or disturbances in practical implementations. The strictly increasing property, meanwhile, guarantees a useful feature for recovery-based control strategies. Empirical validation through extensive simulations further reinforced confidence in the function's suitability, as it exhibited predictable and repeatable behavior under stochastic uncertainties and model perturbations. By adhering to these well-justified selection criteria, the class K function not only fulfills theoretical requirements but also remains practically feasible, contributing to the overall reliability and effectiveness of the control framework.

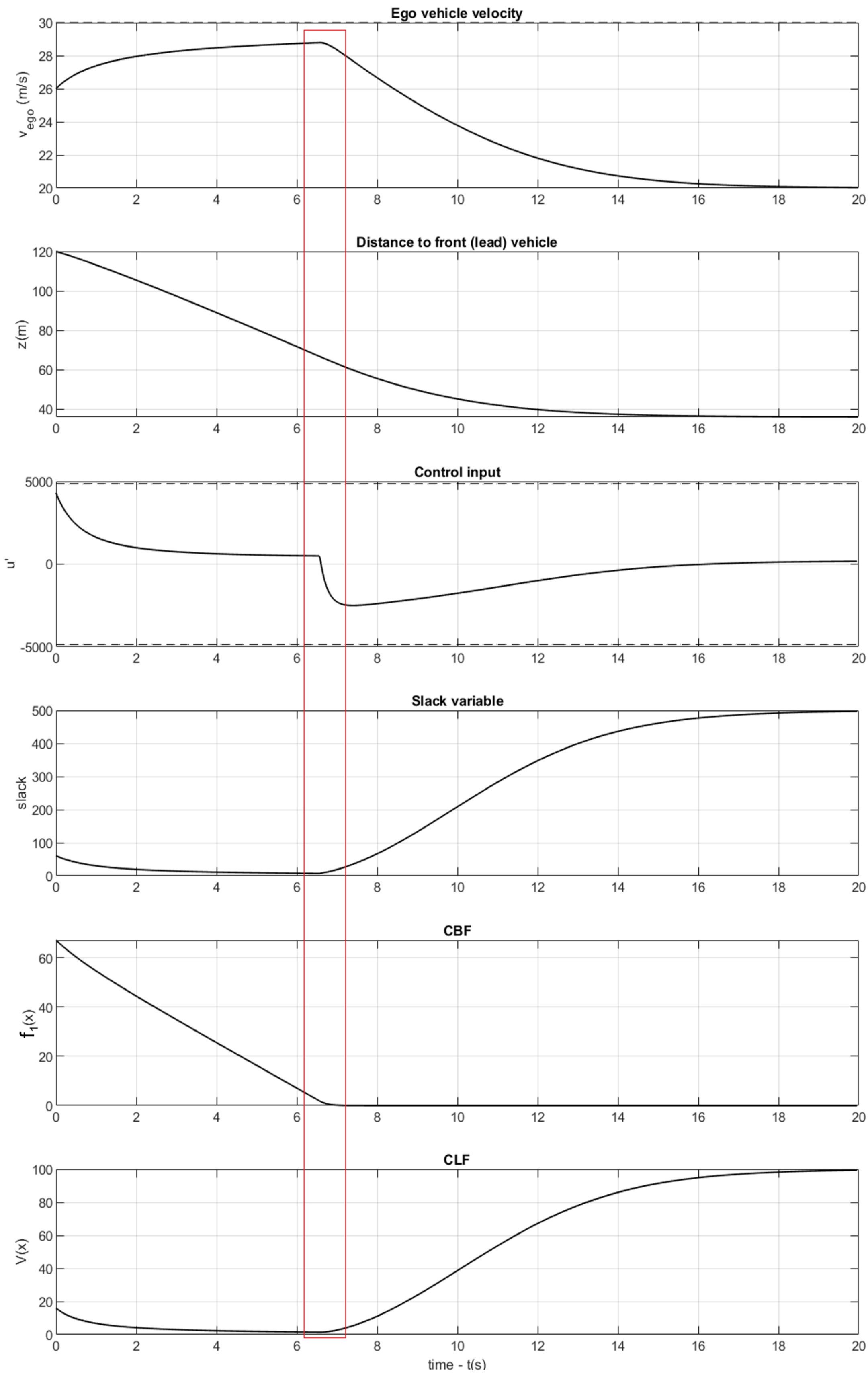


FIGURE 3.2: ACC simulation results for dry surface ($d_{eff} = d_{br}$). Red outlined area shows the sudden drop in value of u' to avoid collision as v_{ego} approaches the desired velocity of 30 m/s at 6.7 $secs$ with corresponding invocation of hard and soft constraints (CBF and CLF), finally maintaining constant values of $v_{ego} = 20$ m/s and $z = 36$ m.

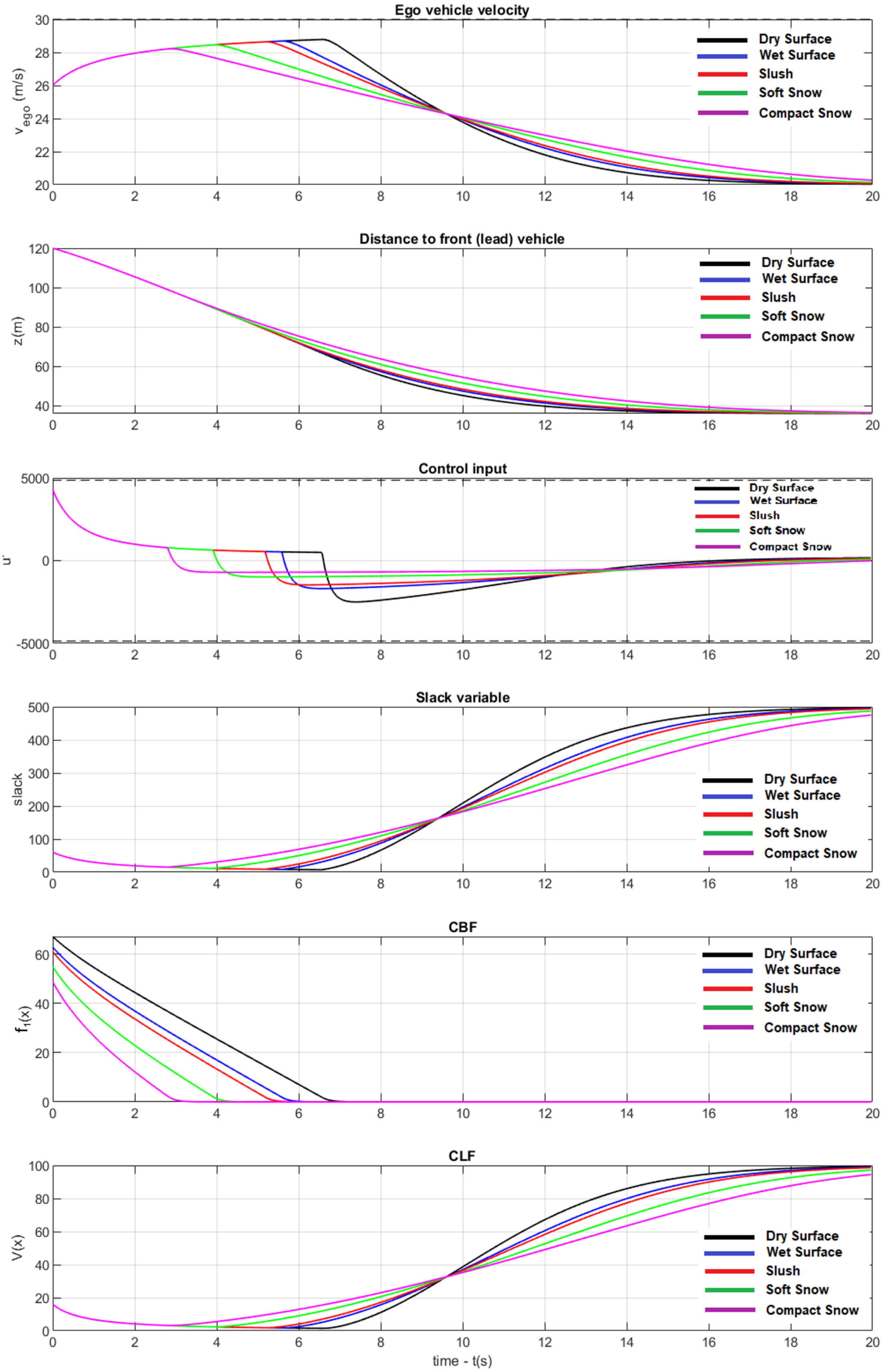


FIGURE 3.3: Simulation results for dry, wet, slush, soft and compact snow. The v_{ego} tries to reach 30 m/s at 3.2 *secs* for compact snow ($d_{eff} = 4d_{br}$), 4 *secs* for soft snow, 5.3 *secs* for slush, 5.8 *secs* for wet and 6.7 *secs* for dry surface with earlier decrease in control inputs, increase in slack variables and invoking of CBF, CLF until the distance z reaches a safe value of 36 m.

3.4 Chapter Summary

In this chapter the designing of a control strategy using control barrier function is considered. A special case of an adaptive cruise control under weather based surface conditions is presented. A control strategy for an adaptive cruise control in ground vehicles using control barrier function under weather based surface conditions is developed and simulated in Matlab©. The presented adaptive cruise control technique utilized control barrier function as part of a quadratic program. The quadratic program unified control barrier function and control Lyapunov function with surface condition taken into hard constraint defined through control barrier function. The simulated results found through this technique exhibited the intended safety and stability objectives on different weather based surface conditions as discussed in section 3.3.

Chapter 4

MRTA based Formation Control

4.1 Introduction

In this chapter, the task execution framework using control barrier function through constraint-oriented approach is developed. As mentioned earlier, the notion of control barrier functions is extensively used in multi robot systems. After exploiting the notion of control barrier function through the example of an adaptive cruise control in chapter 3, here a control strategy for the development of optimization based decentralized controllers using control barrier function is designed as well. This design helped to achieve the multi robot tasks execution in a constraint-oriented approach. Using this constraint-oriented approach, the control design for the leaderless formation control of a multi robot system with constrained rectangular components of velocities under confined conditions is developed and simulated to prove the efficacy of designed algorithm. The idea primarily addresses a real-world scenario being encountered by a multi robot system while achieving the desired objective where each participating robot can move independently in any direction and their velocities can be finite due to certain limitations. After developing the concepts of decentralized optimization controllers using control barrier functions, the control design for an environment-aware resilience of multi robot tasks through the scenario of dynamic tasking for a multi robot system under optimal strategy in constrained conditions is presented in detail in chapter 5.

4.2 Proposed Multi Robot Tasks Accomplishment Framework: Formation Control

In this section, a multi robot control scheme for multi robot task accomplishment is established by constructing decentralized controllers using control barrier functions to accomplish the assigned tasks. Consider the accomplishment of n_t different tasks by a robot as described by the under mentioned optimization problem:

$$\begin{aligned} &\min_u \|u\|^2 \\ &\text{subject to:} \\ &\text{constraints}_j(q,u) \geq 0,\ \forall \in \{1, ..., n_t\} \end{aligned} \tag{4.1}$$

here q represents the state of a robot in a team, u defines the control input to the robot and $\text{constraints}_j(q,u)$ manifest as constraint based function enforcing the performance and accomplishment of task named as j. If n_r represents number of robots in a team, then each robot $i \in \{1, ..., n_r\}$ is modeled as control affine dynamical system as shown below:

$$\dot{q}_i = f_v(q_i) + g_v(q_i)u_i \tag{4.2}$$

where q_i is the state of the robot i, u_i is the control input to the robot i, f_v and g_v are locally Lipschitz continuous vector fields. The control barrier functions can be used to define the assigned tasks as set-oriented tasks for their accomplishment. Safety phenomenon can be understood as avoiding an unsafe set that is ensuring the forward invariance of a set and any control law u that justifies the constraint render a certain set as safe set [50]. Consider a set S_2, a zero superlevel set of a function $f_2(q)$. The function $f_2(q)$ is a continuously differentiable function such that $f_2(q) : D \subset \mathbb{R}^n$ as defined below. Here ∂S_2 depicts the boundary of set S_2:

$$S_2 = \{\ q \in D \subset \mathbb{R}^n : f_2(q) \geq 0\} \tag{4.3}$$

$$\partial S_2 = \{\ q \in D \subset \mathbb{R}^n : f_2(q) = 0\} \tag{4.4}$$

$$\text{interior}(S_2) = \{\ q \in D \subset \mathbb{R}^n : f_2(q) > 0\} \tag{4.5}$$

The set S_2 as defined above is a safe set. Consider the term u which defines a Lipschitz continuous control law and γ defined as an extended class K_∞ function. This function γ is strictly increasing with $\gamma(0) = 0$. If $D \subset \mathbb{R}^n$ such that $S_2 \subset D$, then:

$$\sup_{u \in U} [L_{f_v} f_2(q) + L_{g_v} f_2(q) u] + \gamma(f_2(q)) \geq 0 \tag{4.6}$$

The function $f_2(q)$ is a control barrier function and any Lipschitz control law u that justifies the constraint (expression (4.6)) renders the above defined set S_2 as safe set [50]. The terms L_{f_v} and L_{g_v} are Lie derivatives in the direction of f_v and g_v. The technique designed on this theorem can be used to fabricate and develop a control scheme which permits a heterogeneous or diversified multi robot system to carry-out the set of assigned tasks required to be accomplished. In the following section, the constraint-oriented approach for solving a task accomplishment problem of multi robot formation control using control barrier functions is explored in detail.

4.2.1 Constraint-Oriented Multi Robot Formation Control in Leaderless Consensus under Confined Conditions

In this section, the control design strategy for decentralized optimization-technique based controllers constructed using control barrier functions is presented. It is proved that using such control design strategy, the multiple tasks can be executed by multiple robots in a team. For this, a constraint-oriented approach for the formation control of a first-order multi robot system is carried out in a leaderless consensus where the rectangular velocity components of participating robots are subject to constraints. These robots also avoid inter-robots collisions through the control barrier functions based safety certificates. The desired formation control of different robots and their inter-robots collision avoidance in a team of robots are achieved by solving an optimization problem based on control barrier functions whereas graph theory concepts are used to represent the interaction relations among robots. The devised quadratic program subject to the velocity and safety conditions, minimizes the desired cost function encoded in the control barrier function and generates the separate control inputs for each robot. The rectangular components of the velocity against each robot are kept constrained during the entire operation of the formation control. After designing the control strategy in upcoming sections of this chapter, several simulation examples with different values of velocity constraints are presented in section 4.3, to illustrate the operation of the designed constrained controllers. These constrained controllers ensure that the desired leaderless-consensus based formation remains attainable in a safe manner. The simulated results prove the efficacy of devised optimization-technique based decentralized controllers. It is seen that the robots in the team successfully attained the desired formation in a leaderless consensus, deploying themselves in a plane under constrained rectangular velocities without colliding with each other.

4.2.2 Problem Formulation and Graph Theory

As mentioned briefly in the previous section (4.2) a single integrator dynamical system is considered i.e., $\dot{q} = u$, where q is the state of the system, u is the control input to the system, $q \in \mathbb{R}^n$, $u \in \mathbb{R}^n$, $\mathbb{R}$ denotes the set of real numbers, n is the dimension ($n = 2$) [90]. The objective is to minimize the cost function $J_2(q)$ where $J_2 : \mathbb{R}^n \rightarrow \mathbb{R}_+$, $\mathbb{R}_+$ denotes set of positive real numbers.

$$\text{Objective} := \min_{u} \; J_2(q) \tag{4.7}$$

We define $f_2(q) = -J_2(q)$ as a barrier function and $J_2(q) \geq 0$. The zero-super level set S_2 is a safe set, given by:

$$S_2 = \{q \mid f_2(q) \geq 0\} \tag{4.8}$$

Where $f_2(q) : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable function as defined in previous section. If the following condition is satisfied then f_2 is called the control barrier function.

$$\dot{f}_2(q) = \frac{\partial f_2(q)}{\partial q} u \geq -\gamma(f_2(q)) \tag{4.9}$$

Here the term γ, as mentioned in previous section, where $\gamma : \mathbb{R} \rightarrow \mathbb{R}$ is an extended class k_∞ function, is defined on the entire real line: $\mathbb{R} = (-\infty, \infty)$ [50]. The solution of the optimization problem of equation (4.10) solves the problem of minimization of cost $J_2(q)$ by driving state q to a stationary point $\hat{q}$ of cost $J_2(q)$ such that $\frac{\partial J_2(\hat{q})}{\partial q} = 0 \Longleftrightarrow \hat{q} = 0$ [22]:

$$\begin{aligned} &\underset{u}{\operatorname{argmin}} \; ||u||^2 \\ &\text{subject to:} \\ &\frac{\partial f_2(q)}{\partial q} u \geq -\gamma(f_2(q)) \end{aligned} \tag{4.10}$$

The equation (4.10) can be used for the execution of the desired task that is, the objective of formation control as shown in Fig. 4.1 by producing unconstrained velocities. However to drive robots under constrained rectangular components of velocities while ensuring the accomplishment of the desired task, equation (4.10) can be augmented by imposing constraints on these components (Fig. 4.2) during the entire process of attaining the desired formation. The constraints are denoted by the minimum velocity lower bound $u_{min} \in \mathbb{R}_-$ and the maximum velocity upper bound $u_{max} \in \mathbb{R}_+$. The modified constrained optimization problem takes the following form:

$$\begin{aligned} &\underset{u}{\operatorname{argmin}} \; ||u||^2 \\ &\text{subject to:} \\ &\frac{\partial f_2(q)}{\partial q} u \geq -\gamma(f_2(q)) \\ &u_{min} \leq u \leq u_{max} \end{aligned} \tag{4.11}$$

In subsequent sections during the implementation of equation (4.11), the u_{min} and u_{max} can be written in terms of the rectangular components u_x and u_y of the resultant velocity vector u for each robot as shown in Fig. 4.2 denoted as u_{xmax}, u_{xmin}, u_{ymax} and u_{ymin} respectively. Here u_{xmax}, u_{ymax}, u_{xmin} , u_{ymin} are maximum and minimum values of rectangular components of robot velocities in a two-dimensional plane (x, y) and u_{xmax}, $u_{ymax} \in \mathbb{R}_+$, u_{xmin}, $u_{ymin} \in \mathbb{R}_-$. These values are taken in such a manner that the quadratic program of equation (4.11) remains feasible. As described in the above expression, for a team of n_r robots,

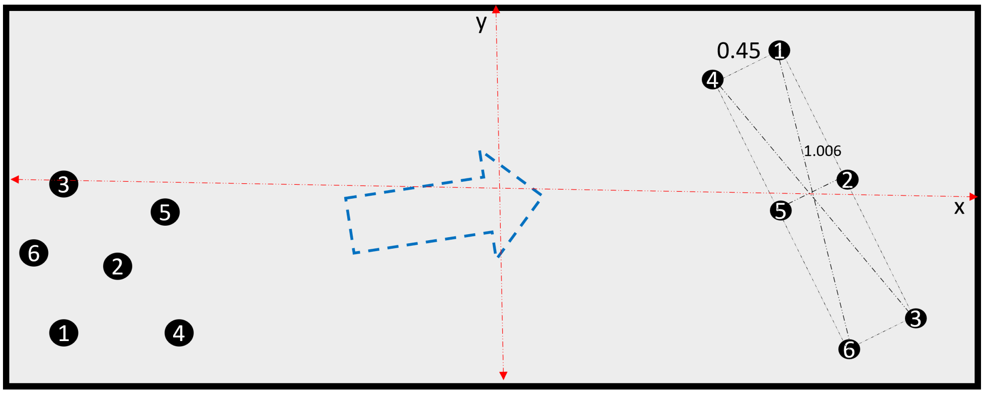


FIGURE 4.1: Six robots forming a rectangular shape for inter-robots linear distance $d_{ij} = 0.45\ m$ and inter-robots diagonal distance $d_g = 1.006\ m$.

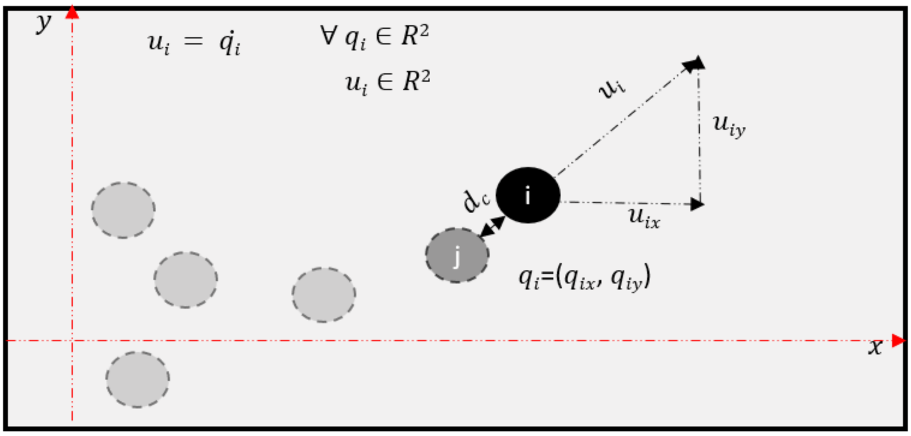


FIGURE 4.2: Robot i in (x, y) plane, q_i and u_i are position and velocity of robot i, u_{ix} and u_{iy} manifest the rectangular components of velocity u_i. $d_c = 0.1\ m$ denotes minimum threshold distance between robot i and j.

the rectangular components based velocity vectors u_{min} and u_{max} are given by:

$$u_{min} = [u_{1xmin},\ u_{1ymin},\ u_{2xmin}, ..., u_{nxmin},\ u_{nymin}]^T \tag{4.12}$$

$$u_{max} = [u_{1xmax},\ u_{1ymax},\ u_{2xmax}, ..., u_{nxmax},\ u_{nymax}]^T \tag{4.13}$$

4.2.3 Deriving a General Expression for Inter-Robots Collisions Avoidance

The goal of the collision avoidance strategy is to allow the robots to execute a task while ensuring that they maintain a minimum threshold distance with other robots while coming in the vicinity of each other, allowing the task to be executed without colliding with each other [24, 27]. The desired behavior is enforced through a constraint formulated by introducing another control barrier function denoted as $c_1(q)$ and $c_1(q) \geq 0$. The zero-super level set K_1 is another safe set in addition to S_2 where $K_1 = \{x \mid c_1(q) \geq 0\}$ and function $c_1(q)$ is a continuously differentiable function $c_1 : \mathbb{R}^n \rightarrow \mathbb{R}$. If the following condition is satisfied then c_1 is called the control barrier function.

$$\dot{c_1}(q) = \frac{\partial c_1(q)}{\partial q} u \geq -\beta(c_1(q)) \tag{4.14}$$

The designed CBF describes a set in the robot's state space in which the robots are able to avoid collisions when a need arises. Here $\beta : \mathbb{R} \rightarrow \mathbb{R}$ is an extended class k_∞ function [50]. To ensure collision avoidance, this set K_1 must be rendered forward invariant. The forward invariance is a property of a dynamical system where a subset of the state space remains invariant under the system's evolution over time. In other words, if the system starts within this subset, it will stay within it for all future times. [50].

4.2.4 Constrained Formation Control with Inter-Robots Collision Avoidance

Multiple control barrier function based constraints can be combined into an optimization problem as formulated in well known literature [22, 91]. Thus the constraints given in equations (4.11) and (4.14) can be combined to obtain the mathematical optimization problem of expression (4.15) for constrained formation control with collision avoidance. The optimization based formulation of expression (4.15) enables the robots to avoid collisions while attaining the desired objective under constrained velocities. The collision avoidance operation starts as soon as the robots reach within a predefined minimum threshold distance from the robots in their vicinity where they stop executing the task and performs the collision avoidance operation. Once the robots avoid collisions, they return to accomplish assigned tasks to complete the desired objective while being constrained by the defined control barrier functions. This is achieved by executing the controller using the following quadratic program [5]:

$$\begin{aligned}
&\underset{u}{\operatorname{argmin}}\ ||u||^2 \\
&\text{subject to:} \\
&\frac{\partial f_2(q)}{\partial q}u \geq -\gamma(f_2(q)) \\
&\frac{\partial c_1(q)}{\partial q}u \geq -\beta(c_1(q)) \\
&u_{min} \leq u \leq u_{max}
\end{aligned} \tag{4.15}$$

In the above expression, the collision constraint specifies a set of admissible inputs that allows robot i to satisfy the desired constraint. In the next section, the KKT conditions are discussed. They are the necessary conditions for the optimality in constrained optimization problems.

4.2.5 KKT Conditions

The generalized form for KKT based optimality conditions against our problem is as follows [22, 90, 92]:

$$\frac{\partial f_2(q)}{\partial q}\hat{u} + \gamma(f_2(q)) \geq 0 \tag{4.16}$$

$$\hat{u} - \hat{u}_{min} \geq 0 \tag{4.17}$$

$$-\hat{u} + \hat{u}_{max} \geq 0 \tag{4.18}$$

$$L_p \geq 0 \tag{4.19}$$

Here $\hat{u}$, $\hat{u}_{min}$ and $\hat{u}_{max}$ are optimal points for u, u_{min} and u_{max}. Term L_p is the Lagrange multiplier and $p \in \{1, 2, 3\}$.

$$\begin{aligned}
&L_1\left(-\frac{\partial f_2(q)}{\partial q}\hat{u} - \gamma(f_2(q))\right) = 0 \\
&L_2,\ L_3 = 0 \\
&2\hat{u} + L_1\left(-\frac{\partial f_2(q)^T}{\partial q}\right) + L_2(-1) + L_3(1) = 0
\end{aligned}$$

From the above expressions, if $L_1 > 0$, then the value of L_1 becomes:

$$L_1 = \frac{-2\gamma f_2(q)}{\left\| \frac{\partial f_2(q)}{\partial q} \right\|^2} \tag{4.20}$$

Since $f_2(q) = -J_2(q)$, therefore the expression for $\hat{u}$ is given by:

$$\hat{u} = -\frac{\gamma f_2(q) \left(\frac{\partial f_2(q)}{\partial q} \right)^T}{\left\| \frac{\partial f_2(q)}{\partial q} \right\|^2} = \frac{\gamma(-J_2(q)) \left(\frac{\partial J_2(q)}{\partial q} \right)^T}{\left\| \frac{\partial J_2(q)}{\partial q} \right\|^2} \tag{4.21}$$

With the above mentioned $\hat{u}$, we have the following equation [22]:

$$\dot{J}_2 = \left(\frac{dJ_2}{dt} \right) = \frac{\partial J_2}{\partial q} \dot{q} = \frac{\partial J_2}{\partial q} \hat{u} = \frac{\gamma(-J_2(q)) \left\| \frac{\partial J_2(q)}{\partial q} \right\|^2}{\left\| \frac{\partial J_2(q)}{\partial q} \right\|^2} = \gamma(-J_2(q)) \tag{4.22}$$

The solution of expression (4.11) gives u of equation (4.21), as time $t \rightarrow \infty$, $q(t) \rightarrow \hat{q}$ such that $\frac{\partial J_2(\hat{q})}{\partial q} = 0$ and $J_2(\hat{q}) = 0$, meaning that state converges to the stationary point of cost $J_2(q)$ that is, $J_2 \rightarrow 0$ as $t \rightarrow \infty$. Thus $\hat{u}$ defined in equation (4.21) solves the minimization objective $(\min_u f_2(q))$ leading to the accomplishment of task encoded in cost $J_2(q)$ which means that stationary point of the cost has reached [22].

4.2.6 Leaderless Consensus in a Multi Robot System

Consensus is a general term which means having robots come to a global agreement on a state value whereas formation means making the robots move to a desired geometric shape [46]. Formation control approaches are of two types. One is the leader-follower approach in which any robot can be chosen as leader while remaining robots follow the leader whose movement is constrained by a predefined trajectory. Other approach is leaderless approach for formation maintenance [46]. Thus each approach depends on how the robots are linked in the communication network [46, 93]. We can say that in a leaderless consensus, all robots in a multi robot system are equal, which means that no single robot is privileged [49]. The objective for all robots to reach a mutual contract (consensus) by cooperating with each other, typically through local communication using graph theory [49]. Here, to attain the desired formation, graph theory can be used to model the relationships between robots in a multi robot system consisting of n_r robots. A graph $g = (v, e)$ is composed of n vertices or nodes $(v_1, ..., v_n)$ and edges $(e_1, ..., e_n)$ connecting the vertices. The position say q of the robot can be taken as vertex v of a graph network. Robots in a graph network are associated with each other based on the assumption mentioned in [22] that is, robots are equipped with sensors and are able to measure the distances between them. Moreover, it is assumed that the communication between robots in the team of robots is ideal that is, no delays or information loss is there. Thus edges show a communication path between the graph vertices that is, the information flows bidirectionally between the robots. An *ith* robot in a multi robot system located at the graph vertex with position denoted by q_i can calculate the distance d_{ij} between *jth* robot placed at the graph vertex denoting robot position q_j and vice versa, where $d_{ij} = q_i - q_j$ and $i,\ j \in \{1, 2, 3, ..., n_r\}$. The neighborhood Φ of robot i in the graph is defined as follows:

$$\Phi_i = \{v_j \in\ v \mid (v_i,\ v_j) \in\ e\} \tag{4.23}$$

These graphs can be represented in terms of matrices. For an undirected graph, the degree of vertex, $\deg(v_i)$, is equal to the number of vertices that are adjacent to vertex v_i in g. The degree matrix of graph g with n number of vertices is a diagonal matrix containing the vertex-degrees of g on the diagonal and is given by $\deg(g)$ [46]:

$$\mathrm{Deg}(g) = \begin{pmatrix} \deg(v_1) & 0 & \cdots & 0 \\ 0 & \deg(v_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \deg(v_n) \end{pmatrix} \tag{4.24}$$

The adjacency matrix $A(g)$ is a symmetric $n \times n$ matrix describing the adjacency relationships in a graph $g(v, e)$ as given by:

$$A(g) = \begin{cases} 1 & \text{if } v_i,\ v_j \in\ e \\ 0 & \text{otherwise.} \end{cases}$$

Graph Laplacian Lap(g) is the matrix representation of a graph $g(v, e)$. The graph Laplacian of an undirected graph $g(v, e)$ is given by:

$$\mathrm{Lap}(g) = \mathrm{Deg}(g) - A(g) \tag{4.25}$$

The objective of consensus is to agree on a common value for all participating robots in a multi robot system [94]. If g is the communication graph among n_r robots in a multi robot system and $Q = (q_1, q_2, q_3, ..., q_{n_r})^T$ represents their ensemble state then the consensus protocol takes the following form [94]:

$$\dot{Q} = -\mathrm{Lap}(g)\ Q \tag{4.26}$$

If the graph is connected and undirected, then Lap is symmetric and positive semi-definite with the following properties of its eigen values (eig):

$$\mathrm{eig}_1 = 0 \tag{4.27}$$

$$\mathrm{eig}_1 \le \mathrm{eig}_2 \le \cdots \le \mathrm{eig}_{n_r} \text{ and } \mathrm{eig}_2 > 0 \tag{4.28}$$

To achieve our desired objectives for a leaderless consensus scenario, it is assumed that the graph is undirected and connected as depicted in Fig. 4.3. In the case of the formation control problem of a multi robot system with n_r robots, the robot network encoded through the graph $g(v, e)$ can achieve formation control by driving the robots in such a way that $\| q_i - q_j \|$ converges asymptotically to the desired inter-robot distances for all i, j such that q_i, $q_j \in e$. Moreover, in a multi robot system following a leaderless consensus, as soon as the desired consensus is reached, the velocity of each participating robot finally converges to zero [73]. Considering the consensus problem for the robots i and j in Fig. 4.1 to reach a

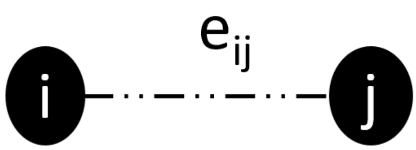


FIGURE 4.3: An undirected edge e_{ij} between robots i and j - connected graph.

mutually agreed common location in space, the following consensus equation can be written [46, 93]:

$$\dot{q}_i = -\sum_{j \in \Phi_i} (q_i - q_j) \tag{4.29}$$

Where Φ_i denotes the near-by robots of ith robot in a multi robot system. Using the above-mentioned basic form of the consensus equation, one can attempt to solve the formation control problem of a multi robot system consisting of n_r robots by simply running the consensus equation over the desired inter-robots distances [93]. In leaderless consensus, robots decide mutually to place themselves at common location in space [46, 93] which is opposite the case of leader-follower approach. In leader-follower consensus, the multi robot system's goal is not to settle on a common state value, but followers have to track the leader's state [49, 95]. The leader's state commands the behavior of the followers, making them to track to the leader's state or trajectory [49, 95].

4.2.7 Constraint-Oriented Formation Control under Constrained Velocities in Leaderless Consensus

As single integrator dynamics is considered here and also that each robot can be controlled through the velocity [6]. If q is the position of the robot then following is the first-order dynamics of a robot i in a multi robot system:

$$\dot{q}_i = u_i \tag{4.30}$$

Here i, $q_i \in \mathbb{R}^2$ and $u_i \in \mathbb{R}^2$ denote the robot i, the position of robot i and the velocity of robot i respectively, in a multi robot system consisting of n_r robots and $i \in \{1, 2, 3, ..., n_r\}$. If $f_i(q)$ is a one-to-one function between robot i and near-by robot j then each robot receives a control input u_i as given by:

$$\begin{aligned} &\min_{u_i} \; ||u_i||^2 \\ &\text{subject to:} \\ &-\frac{\partial J_{2i}(q)}{\partial q_i} u_i \geq -\gamma(-J_{2i}(q)) \\ &u_i \leq u_{max} \\ &u_i \geq u_{min} \end{aligned} \tag{4.31}$$

If $d_{ij} \in \mathbb{R}_+$ is the inter-robot distance in a multi robot system, the total cost $J_2(q)$ of a multi robot system consisting of n_r robots is given by [22]:

$$J_2(q) \quad = \quad \sum_{i=1}^{n_r} \sum_{j \in \Phi_i} \frac{1}{2}((\|q_i - q_j\|)^2 - d_{ij}^2)^2 \tag{4.32}$$

The formation control problem (cost $J_2(q)$) of equation (4.32) is a consensus problem [95] since as mentioned earlier that there is no robot with privileged role as leader [46, 49], the consensus is a leaderless consensus [46, 73].
In upcoming section the equation (4.32) gives us leaderless consensus based formation control for our multi robot system consisting of n_r robots by placing these robots in a predefined shape anywhere in space while maintaining their relative positions as per predefined inter-robots distances [46, 73, 95]. Here Φ_i denotes the near-by robots of the ith robot in a multi robot system. The pairwise cost between robot i and the near-by jth robot is given by $J_{2i}(q)$ as follows:

$$J_{2i}(q) = \sum_{j \in \Phi_i} \frac{1}{2}((\|q_i - q_j\|)^2 - d_{ij}^2)^2 \tag{4.33}$$

The derivative of $J_{2i}(q)$ is given by:

$$\frac{\partial J_{2i}(q)}{\partial q_i} = \sum_{j \in \Phi_i} 2((\|q_i - q_j\|)^2 - d_{ij}^2)(q_i - q_j) \tag{4.34}$$

The control input for the ith robot can be written while summing the constraints for each robot as follows:

$$\min_{u_i} \; ||u_i||^2 \tag{4.35}$$

subject to:

$$-(\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} 2(((\|q_i - q_j\|)^2 - d_{ij}^2)(q_i - q_j)))u_i$$

$$\geq -\gamma(\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} (-\frac{1}{2}((\|q_i - q_j\|)^2 - d_{ij}^2)^2))$$

$$u_{xmin} \leq u_{ix} \leq u_{xmax}$$

$$u_{ymin} \leq u_{iy} \leq u_{ymax}$$

Here u_{ix} and u_{iy} represent the components of velocity u_i in a two-dimensional space along x and y axes (Fig. 4.2). The above minimization expression (4.35) becomes a mathematical optimization problem and is further categorized as a quadratic program [50, 92].

The standard expression of a quadratic program based controllers for solving optimization program of (4.35) is shown in expression (4.36). In expression (4.36), H_2 is $2n_r \times 2n_r$ symmetric matrix, F_2 is a real valued vector, A_2 is $1 \times 2n_r$ dimensional matrix and b_2 is 1×1 real-valued vector. Moreover in the constraints of equation (4.36), u_{min} and u_{max} are the lower and upper bounds for rectangular velocity components (u_x,u_y) against each robot's velocity vector u in a two-dimensional plane (x, y). Both u_{min} and u_{max} are $1 \times 2n_r$ real valued vectors.

$$\begin{aligned} &\min_u \frac{1}{2} u^T H_2 u + F_2^T u \\ &\text{subject to:} \\ &A_2 u \leq b_2 \\ &u_{min} \leq u \leq u_{max} \end{aligned} \tag{4.36}$$

Here A_2 and b_2 are from equation (4.36) as given by:

$$A_2 = \sum_{i=1}^{n_r} \sum_{j \in \Phi_i} 2((\|q_i - q_j\|)^2 - d_{ij}^2)(q_i - q_j)), \quad \forall i \neq j \tag{4.37}$$

$$b_2 = -\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} \frac{1}{2}((\|q_i - q_j\|)^2 - d_{ij}^2)^2, \quad \forall i \neq j \tag{4.38}$$

The above-mentioned optimization problem (equation (4.36)) renders $2n_r$ number of control inputs (u) for a multi robot system consisting of n_r robots by constraining the $2n_r$ rectangular velocity components. In the section 4.3, simulation results as an outcome of expression (4.36) show that by constraining the rectangular components of the velocities, the desired robots formation is achievable. The simulation results without the constraints on the robots velocities for comparison purposes are also presented in the same section. In the following section another scenario of constraint-oriented formation control under constrained rectangular velocity components is presented with inter-robots collisions avoidance.

4.2.8 Formation Control under Constrained Velocity Components with Inter-Robots Collisions Avoidance

To implement collision-free formation control for the ith robot in n_r robots, the general expression of (4.15) is rewritten here as follows [96]:

$$
\begin{aligned}
&\min_{u_i} ||u_i||^2 \\
&\text{subject to:} \\
&-\frac{\partial J_{2i}(q)}{\partial q_i} u_i \geq -\gamma(-J_{2i}(q) \\
&\frac{\partial c_{1i}(q)}{\partial q_i} u_i \geq -\beta(c_{1i}(q)) \\
&u_{min} \leq u_i \leq u_{max}
\end{aligned}
\tag{4.39}
$$

To avoid inter-robots collisions, a robot i must maintain a minimum safe distance from its neighborhood robots such that $d_c \in \mathbb{R}_+$ [5].

The pairwise collision avoidance cost is given by $c_{1i}(q)$. The pairwise safe set K_{1ij} which is a super-level set of function $c_{1i}(q)$ is given by:

$$
K_{1ij} \quad = \{c_{1i}(q) = \quad (\|q_i - q_j\|)^2 - d_c^2 \geq 0\} \tag{4.40}
$$

where $i,\ j \in \{1...n_r\}$ and $i \neq j$. The control input for ith robot and the corresponding constraints for each robot can be written as follows:

$$
\min_{u_i} ||u_i||^2 \tag{4.41}
$$

subject to:

$$
(-\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} 2(((\|q_i - q_j\|)^2 - d_{ij}^2)(q_i - q_j)))u_i \tag{4.41a}
$$

$$
\geq -\gamma(-\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} \frac{1}{2}(((\|q_i - q_j\|)^2 - d_{ij}^2)^2)) \tag{4.41b}
$$

$$
2(q_i - q_j)u_i \geq -\beta((\|q_i - q_j\|)^2 - d_c^2) \tag{4.41c}
$$

$$
u_{xmin} \leq u_{ix} \leq u_{xmax} \tag{4.41d}
$$

$$
u_{ymin} \leq u_{iy} \leq u_{ymax} \tag{4.41e}
$$

The controller for formation control with collision avoidance under constrained velocity components can be expressed by the following standard quadratic program:

$$\min_{u} \frac{1}{2}u^T H_3 u + F_3^T u \tag{4.42}$$

subject to:

$$A_3 u \leq b_3$$

$$A_c u \leq b_c$$

$$u_{min} \leq u \leq u_{max}$$

In expression (4.42), H_3 is $2n_r \times 2n_r$ symmetric matrix, F_3 is a real valued vector, A_3 is $1 \times 2n_r$ dimensional matrix and b_3 is 1×1 real-valued vector. Here A_3, A_c, b_3 and b_c are given as:

$$A_3 = \sum_{i=1}^{n_r} \sum_{j \in \Phi_i} 2((\|q_i - q_j\|)^2 - d_{ij}^2)(q_i - q_j), \quad \forall i \neq j \tag{4.43}$$

$$A_c = -2(q_i - q_j), \quad \forall i \neq j \tag{4.44}$$

$$b_3 = -\sum_{i=1}^{n_r} \sum_{j \in \Phi_i} \frac{1}{2}((\|q_i - q_j\|)^2 - d_{ij}^2)^2, \quad \forall i \neq j \tag{4.45}$$

$$b_c = (\|q_i - q_j\|)^2 - d_c^2, \quad \forall i \neq j \tag{4.46}$$

As an extension of the optimization problem of expression (4.36), the optimization problem of expression (4.42) renders the capability to generate $2n_r$ control inputs for n_r robots by constraining $2n_r$ rectangular velocity components as well as each robot to avoid $n_r - 1$ probable collisions.

In section 4.3, the simulation results from integrating the velocity and collision avoidance constraints (expression (4.42)) demonstrate the successful attainment of the desired formation control, validating the effectiveness of the proposed approach. These simulation results align closely with those obtained by simulating equation (4.36), confirming theoretical expectations with empirical evidence. Furthermore, the strength of the control strategy is highlighted by varying velocity constraints. In the next chapter 5, the detailed design of the control framework for the multi-robot task allocation and accomplishment scheme is presented, laying the foundation for scalable and adaptive coordination in complex environments.

4.3 Simulation Results and Discussions

In this section, the simulation results for the multi robot task accomplishment using the decentralized optimization controllers based on the control barrier functions are presented. Here, the assumptions mentioned in [22] are taken that is, the robots are equipped with sensors and are able to measure the distances between them. Moreover, it is assumed that the communication between robots in the team is ideal that is, no delays or information loss is there. Based on these assumptions, an algorithm is developed for the simulation purpose using a quadratic program to achieve the simulation results for the constraint-oriented formation control of a multi robot system in leaderless consensus under confined conditions; a control scheme already developed in section 4.2.1. Using the developed algorithm a Matlab© routine for the optimization programs of (4.36) and (4.42) can then be programmed in which n_r robots can be defined with their initial positions taken within an area of 3×2 square meter. This area extends from x_{min} to x_{max} along x-axis and y_{min} to y_{max} along y-axis, where values for x_{min}, y_{min}, x_{max} and y_{max} are -1.5, -1, 1.5 and 1 respectively. The positions of the robots are initialized in $\mathbb{R}^2$ using the Matlab© command *rand*. The neighborhood Φ of each robot in the team is identified. The corresponding degree matrix (Deg), adjacency matrix (A) and graph Laplacian (Lap) are calculated to achieve the desired formation (section 4.2.1). By selecting the *ith* robot out of n_r robots and calculating its positions q_{ix}, q_{iy} alongside the positions of neighborhood robots q_{jx}, q_{jy}, the A_{2ij}, b_{2ij}, A_{3ij} and b_{3ij} of each robot against its neighborhood can be calculated for the mathematical expressions of (4.36) and (4.42). These quadratic problems can be solved at each time step to obtain u_{ix}, u_{iy}, u_{jx} and u_{jy} for each robot in order to move the robots in a plane while calculating the updated positions q_{ix}, q_{iy}, q_{jx} and q_{jy} using equations (4.50) and (4.51) until the desired formation is achieved as of Fig. 4.1. The following parameters are used here to perform different experiments.

- No. of robots: $n_r = 6$
- Minimum threshold distance for collision avoidance: 0.1 m
- Resultant velocities vector:

$$u_{2n_r \times 1} = [u_{1x},\ u_{1y},\ u_{2x},\ u_{2y}, ...,\ u_{6y}]^T \tag{4.47}$$

- Lower bound for $2n_r$ rectangular velocity components:

$$u_{min} = -[u_{1xmin},\ u_{1ymin},\ u_{2xmin},\ u_{2ymin}, ..., u_{6ymin}]^T \tag{4.48}$$

- Upper bound for $2n_r$ rectangular velocity components:

$$u_{max} = [u_{1xmax},\ u_{1ymax},\ u_{2xmax},\ u_{2ymax}, ..., u_{6ymax}]^T \tag{4.49}$$

- The formulae for calculating the updated positions q_{ix}, q_{iy} of the robot i in x and y directions where t and dt denote the time and time-step respectively, are:

$$q_{ix}^{t+dt} = q_{ix}^{t} + u_{ix}^{t} dt,\ \ i \in \{1, ..., n_r\} \tag{4.50}$$

$$q_{iy}^{t+dt} = q_{iy}^{t} + u_{iy}^{t} dt,\ \ i \in \{1, ..., n_r\} \tag{4.51}$$

- Graph Laplacian (Lap) is given by:

$$\text{Lap} = \begin{bmatrix} 3 & -1 & 0 & -1 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 3 & -1 & 0 & 1 \\ -1 & 0 & -1 & 3 & -1 & 0 \\ 0 & -1 & 0 & -1 & 3 & -1 \\ -1 & 0 & -1 & 0 & -1 & 3 \end{bmatrix} \tag{4.52}$$

To make a predefined formation shape of the Fig. 4.1 against the values for the inter-robots linear distance d_{ij}= 0.45 m and diagonal distance d_g = 1.006 m, the weight matrix W (equation (4.53)) showing the relationships between robots and their neighbourhood (Table 4.1) becomes as follows:

$$W = \begin{bmatrix} 0 & 0.45 & 0 & 0.45 & 0 & 1.006 \\ 0.45 & 0 & 0.45 & 0 & 0.45 & 0 \\ 0 & 0.45 & 0 & 1.006 & 0 & 0.45 \\ 0.45 & 0 & 1.006 & 0 & 0.45 & 0 \\ 0 & 0.45 & 0 & 0.45 & 0 & 0.45 \\ 1.006 & 0 & 0.45 & 0 & 0.45 & 0 \end{bmatrix} \tag{4.53}$$

Table 4.1: Robots and their neighborhood.

Robot r_i	Neighborhood (w_{ij}/r_i)
r_1	$0.45/r_2$, $0.45/r_4$, $1.006/r_6$
r_2	$0.45/r_1$, $0.45/r_3$, $0.45/r_5$
r_3	$0.45/r_2$, $1.006/r_4$, $0.45/r_6$
r_4	$0.45/r_1$, $1.006/r_3$, $0.45/r_5$
r_5	$0.45/r_2$, $0.45/r_3$, $0.45/r_6$
r_6	$1.006/r_1$, $0.45/r_3$, $0.45/r_5$

For the simulation purpose, following values of the velocity constraints are used.

$$\text{Velocity constraint 1} = \begin{cases} u_{min} & = -[1, 1, 1, ..., 1]^T \\ u_{max} & = [1, 1, 1, ..., 1]^T \end{cases} \tag{4.54}$$

$$\text{Velocity constraint 2} = \begin{cases} u_{min} & = -[1.5, 1.5, 1.5, ..., 1.5]^T \\ u_{max} & = [1.5, 1.5, 1.5, ..., 1.5]^T \end{cases} \tag{4.55}$$

$$\text{Velocity constraint 3} = \begin{cases} u_{min} & = -[2, 2, 2, ..., 2]^T \\ u_{max} & = [2, 2, 2, ..., 2]^T \end{cases} \tag{4.56}$$

In the succeeding sections while showing numerous simulation results, the resultant velocity u for each robot is plotted where $u_i = \sqrt{u_{ix}^2 + u_{iy}^2}$ and $i \in \{1, 2, 3, 4, 5, 6\}$. Moreover for the scenarios considered here, owing to the random initializations of q_{ix} and q_{iy} in any quadrant of the Cartesian quadrant based experimental area, the corresponding effect on the magnitude of u_i against each robot varies even when the values for bounds on the velocity components are increased in magnitude that is, from equations (4.54) to (4.56). It is pertinent to mention that the formation control is also achieved in [22] but without constraints on the velocities; however here the desired formation is achieved through the constrained velocity components coupled with the collision avoidance strategy in a leaderless consensus as observed from the simulations shown in the next sections, thus successfully validating the established notions. It is seen in the following subsections that case I (Figures

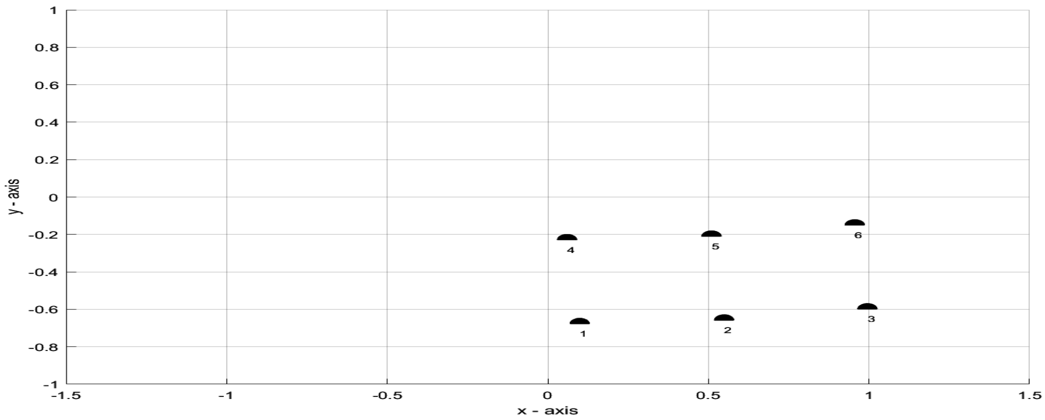
Fig. 4.4(a), Fig. 4.5(a), Fig. 4.6(a)) and case II (Figures Fig. 4.10(a), Fig. 4.11(a), Fig. 4.12(a)) manifest a leaderless consensus based formation control as robots finally formed the desired shape anywhere in space, while these robots ensure to sustain their relative positions to each other [46, 49, 97].

4.3.1 Case I: Simulation Results for Constraint-Oriented Formation Control under Velocity Constraints in Leaderless Consensus

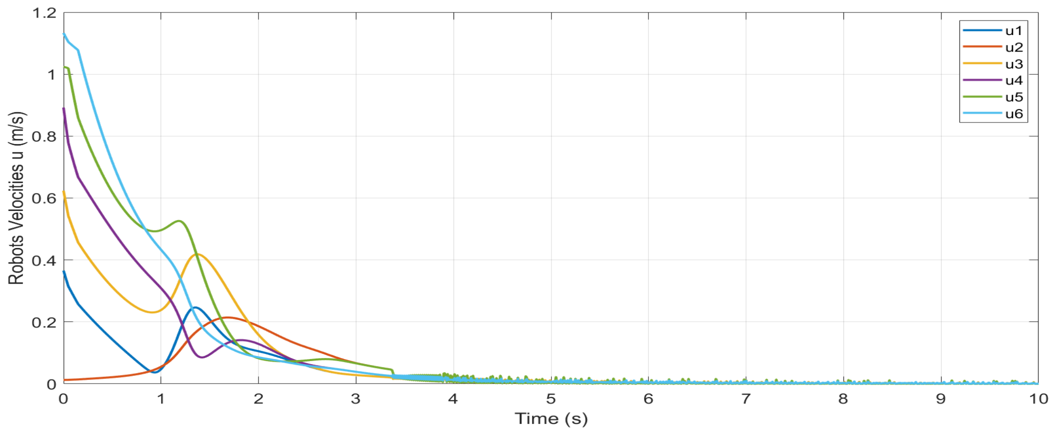
In this section, the simulation results of equation (4.36) against $H_1 = I_{2n_r \times 2n_r}$ and $F_1 = Zeroes_{1\times 2n_r}$ are presented. It can be observed from the Fig. 4.4(a) that the rectangular formation is achieved under the velocity constraints with upper and lower bounds as of the equation (4.54). The resultant velocities of the robots start with different initial values as shown in Fig. 4.4(b) however these velocities after decreasing over a period of time finally become zero as soon as the desired formation is achieved and mathematically the cost J_2 corresponding to that particular robot becomes zero that is, the distance between robots becomes equal to the weights defined in the matrix W (equation (4.53)) and $\frac{\partial J_2(\hat{q})}{\partial q} = 0$. For elaboration and validation of the proposed notion, different simulations are performed with the different values for the upper and lower constraints of the velocity components as mentioned in the equations (4.55) and (4.56) respectively. The corresponding simulation results are shown in the Figs. 4.5 and 4.6. The simulation results show that a rectangular formation shape is achieved under the constrained rectangular velocity components in a leaderless consensus.

4.3.2 Case II: Simulation Results for Constraint-Oriented Formation Control under Velocity Constraints with Collisions Avoidance in Leaderless Consensus

The equation (4.42) having $H_3 = I_{2n_r \times 2n_r}$ and $F_3 = Zeroes_{1\times 2n_r}$ with control barrier function based collision avoidance constraint, can be solved and the obtained



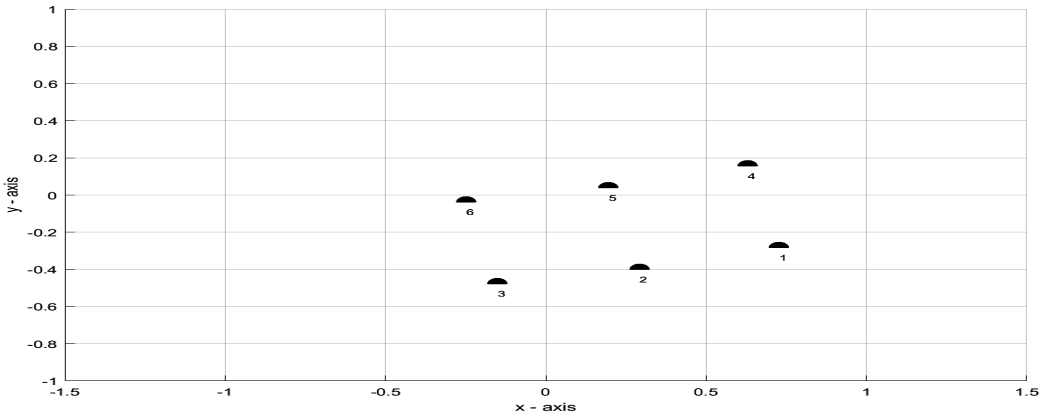
(a)



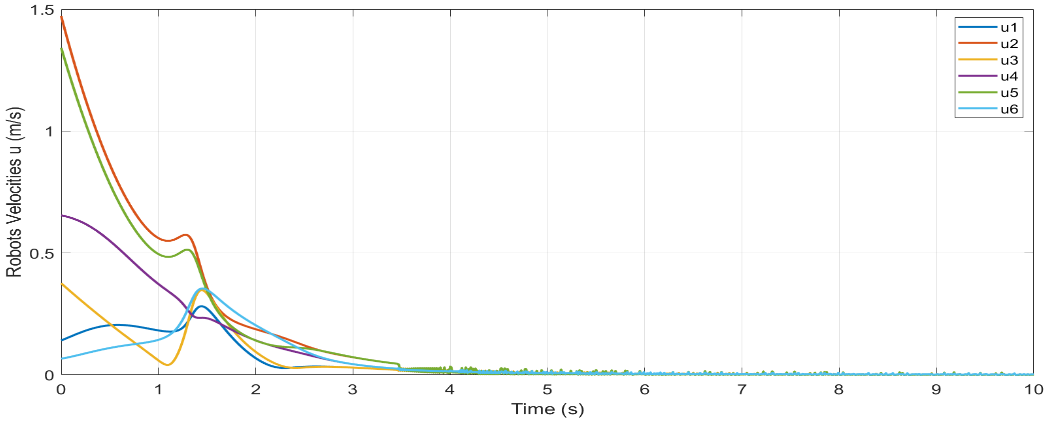
(b)

FIGURE 4.4: Velocity constraint 1: Fig. 4.4(a) Formation control with values for rectangular velocities $u_{min} = -[1, 1, 1, ..., 1, 1, 1]$ and $u_{max} = [1, 1, 1, ..., 1, 1, 1]$ (equation (4.36)). Fig. 4.4(b) Resultant velocities ($u_1 - u_6$) of six robots, finally become zero as soon as formation is achieved.

results can further be compared with the results of case I achieved by solving the equation (4.36). With an appropriate value of $D \in \mathbb{R}_+$ in the equation (4.40), the simulation results of the equation (4.42) are shown in the Figs. 4.8 and 4.9. The simulation of the equation (4.36) gives us the results of the Fig. 4.7 that is, the robots 2 and 3 collide with each other because no collision avoidance strategy is incorporated in the equation (4.36). The Figs. 4.8 and 4.9 show the results with control barrier functions based collision avoidance as constraints. The collision between the robots 2 and 3 is imminent as shown in Fig. 4.8(a). However the robot 3 evaded the collision by changing its trajectory course under the influence of its CBF as soon as it approached the robot 2 as shown in Fig. 4.8(b). In Fig. 4.9(a), the robot 3 places itself away after changing its trajectory until complete rectangular formation is achieved as shown in Fig. 4.9(b). The Figs. 4.10, 4.11 and



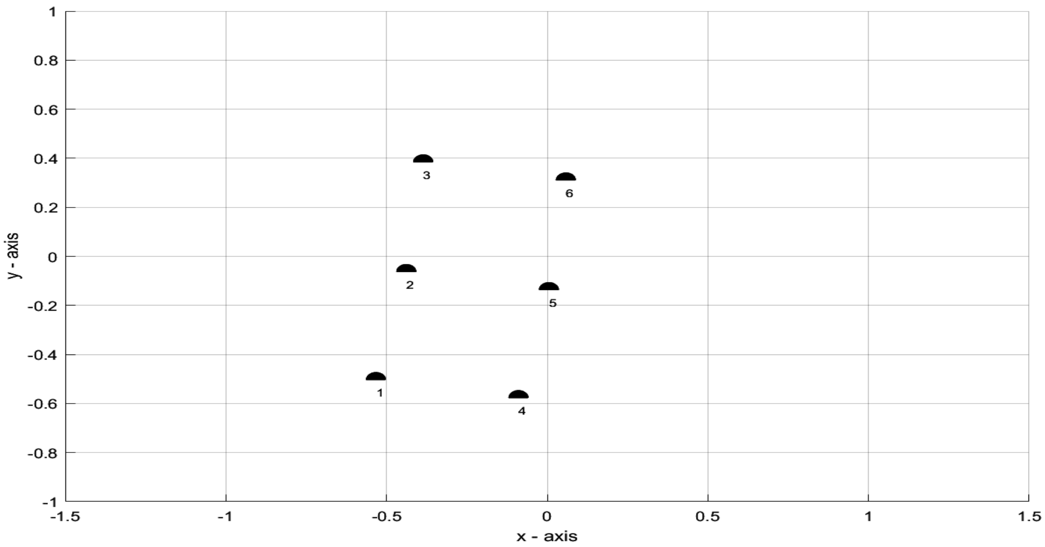
(a)



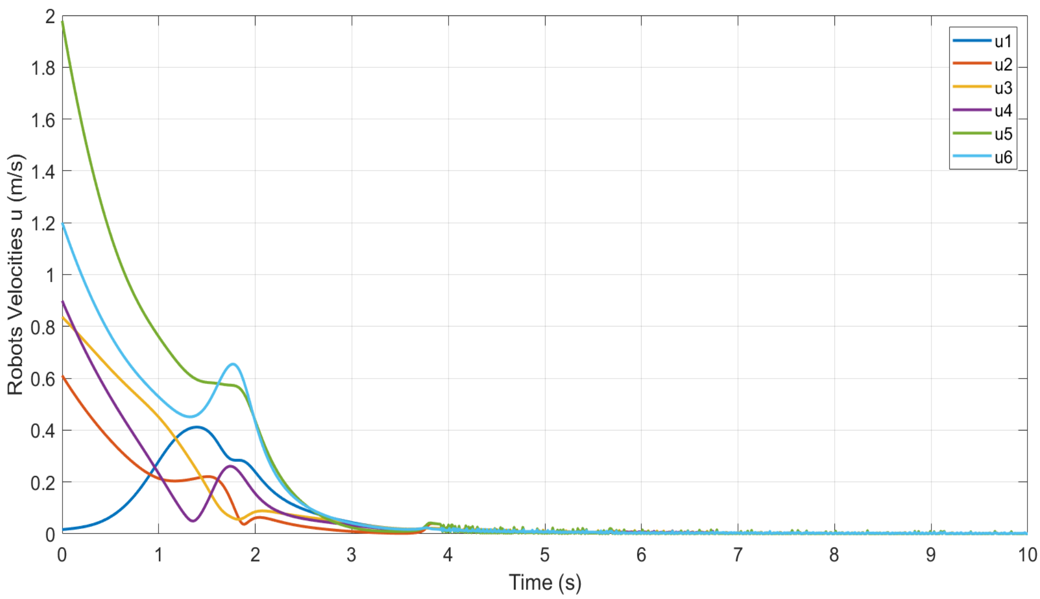
(b)

FIGURE 4.5: Velocity constraint 2: Fig. 4.5(a) Formation control with values for rectangular velocities $u_{min} = -[1.5, 1.5, 1.5, ..., 1.5, 1.5, 1.5]$ and $u_{max} = [1.5, 1.5, 1.5, ..., 1.5, 1.5, 1.5]$ (equation (4.36)). Fig. 4.5(b) Robots resultant velocities ($u_1 - u_6$).

4.12 present the simulation results of the optimization program of (4.42) indicating that the desired formation remained achievable under all the constraints that is, the control barrier functions based collision avoidance in addition to the constraints on the rectangular components of the robots velocities. These simulation results can be compared with the results achieved by simulating the optimization program of (4.36) where a smooth change in the velocities is observed as shown in the Figs. 4.4, 4.5 and 4.6. However in the Figs. 4.10, 4.11 and 4.12, abrupt change in the velocities of the robots during the simulation is observed which occurred owing to the evasion of inter-robots collisions. This abrupt change occurred due to the enforcement of the safety certificate of the equation (4.40) further encoded in the optimization program of (4.42) through the collision avoidance based control barrier function.



(a)

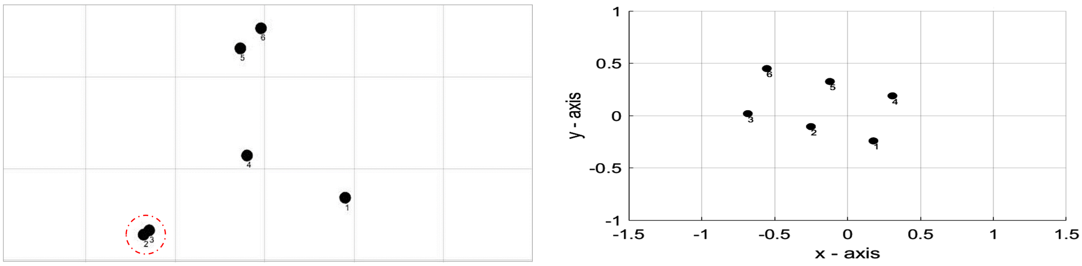


(b)

FIGURE 4.6: Velocity constraint 3: Fig. 4.6(a) Formation control with values for rectangular velocities $u_{min} = -[2, 2, 2, ..., 2, 2, 2]$ and $u_{max} = [2, 2, 2, ..., 2, 2, 2]$ (equation (4.36)). Fig. 4.6(b) Resultant velocities $(u_1 - u_6)$.

4.3.3 Case III: Simulation Results for Formation Control without Velocity Constraints in Leaderless Consensus

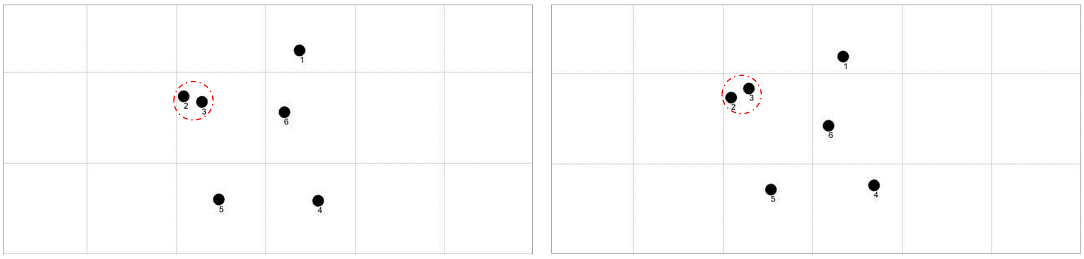
The simulation results of the optimization program of (4.36) without constraints on velocities are shown in Fig. 4.13(a). It is observed that rectangular formation is achieved without velocity constraints and inter-robots collision avoidance as also



(a) Snapshot taken during simulation, robots 2 and 3 collide with each other because no collision avoidance strategy is incorporated.

(b) Rectangular shape formation.

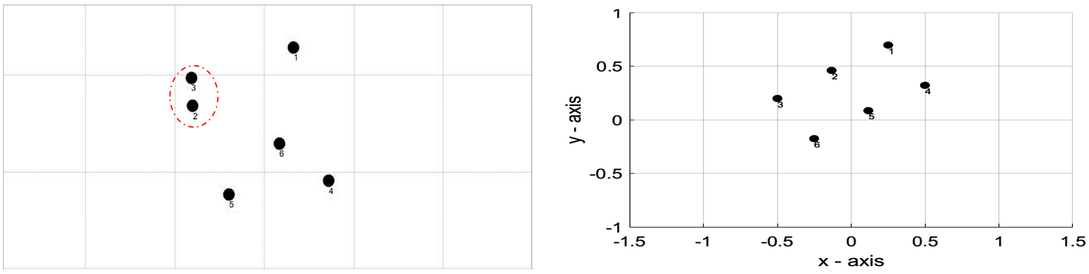
FIGURE 4.7: Formation forming under constrained rectangular velocities with no collision avoidance strategy (equation (4.36)).



(a) Snapshot during simulation; collision between robots 2 and 3 is imminent (equation (4.42)).

(b) Simulation snapshot shows that robot 3 is evading an imminent collision from robot 2 due to imposed collision avoidance constraint.

FIGURE 4.8: Collision avoidance operation while formation forming under constrained rectangular velocities.

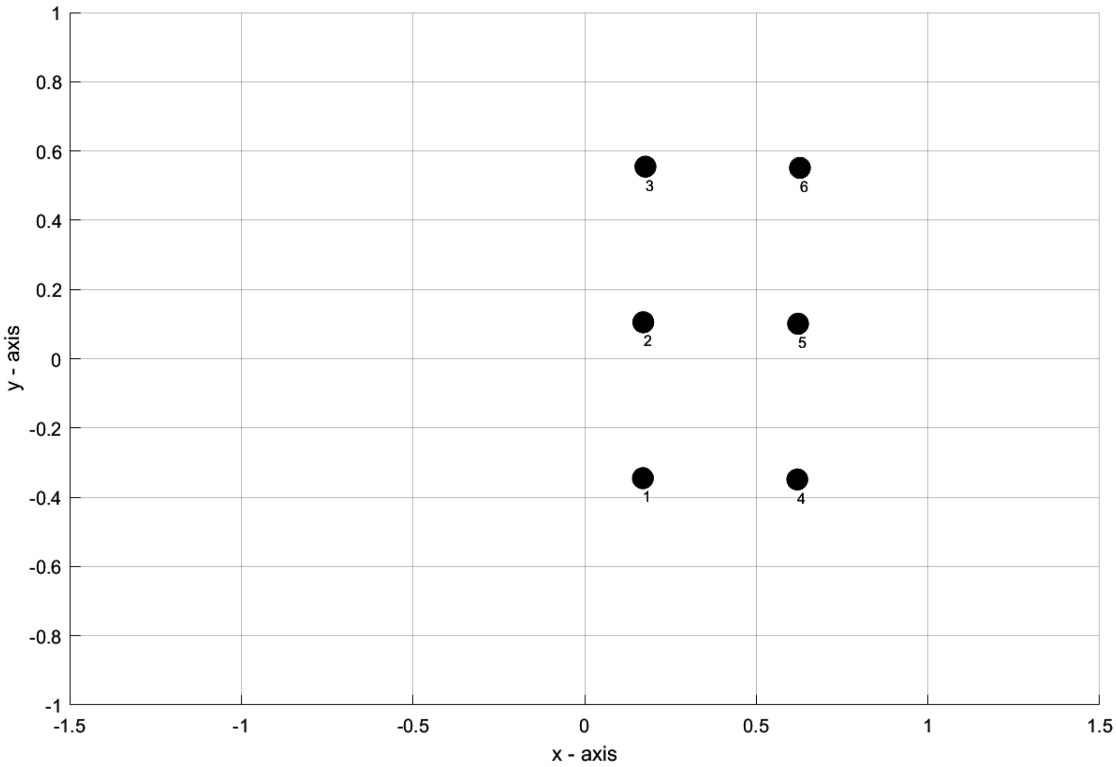


(a) Simulation snapshot during formation control, robot 3 finally evaded the collision with robot 2 by moving safely around robot 3.

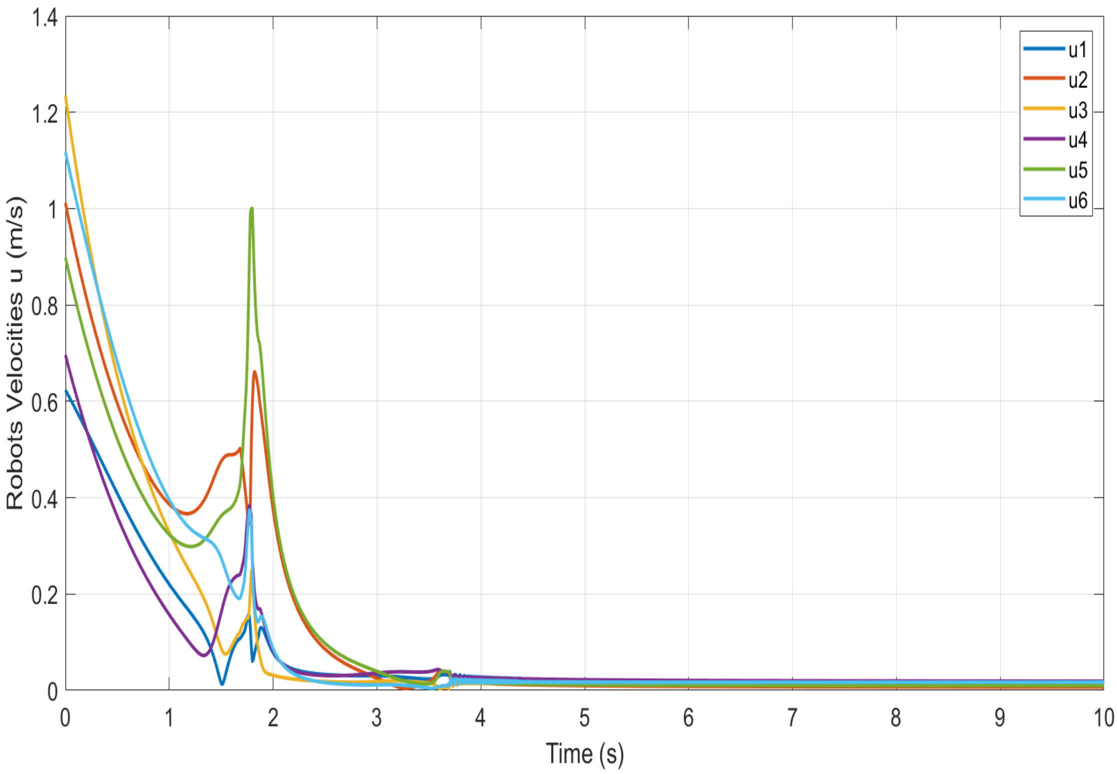
(b) Desired shape formation upon completion of simulation.

FIGURE 4.9: Formation forming under collision avoidance constraint (equation (4.42)).

manifested in [22]. After establishing the aforementioned notion of decentralized controllers, in the upcoming chapter 5, the theory and simulation results on the environment aware resilience of a multi robot system are presented. The chapter provides a detailed discussion on the proposed scenario. The attainment of desired resilience of tasks is presented through the scenario of dynamic tasking for a multi robot system under optimal strategy in constrained conditions, control scheme for which developed in the section 5.3.2.

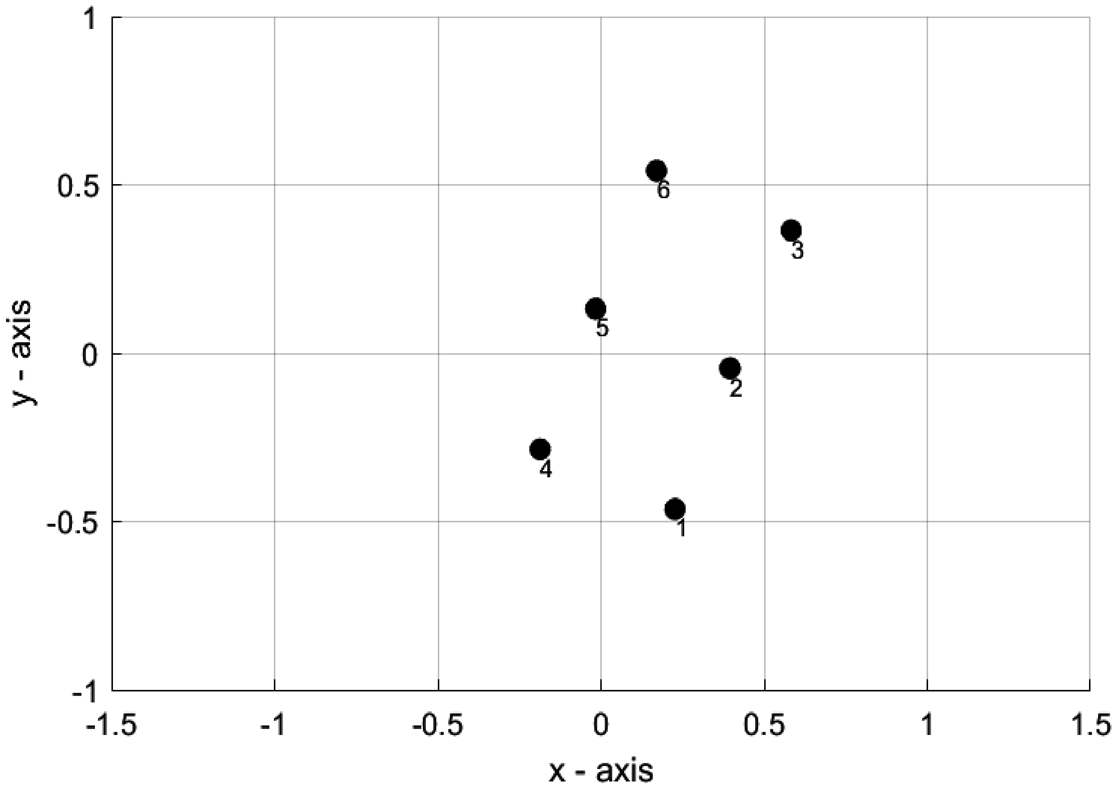


(a)

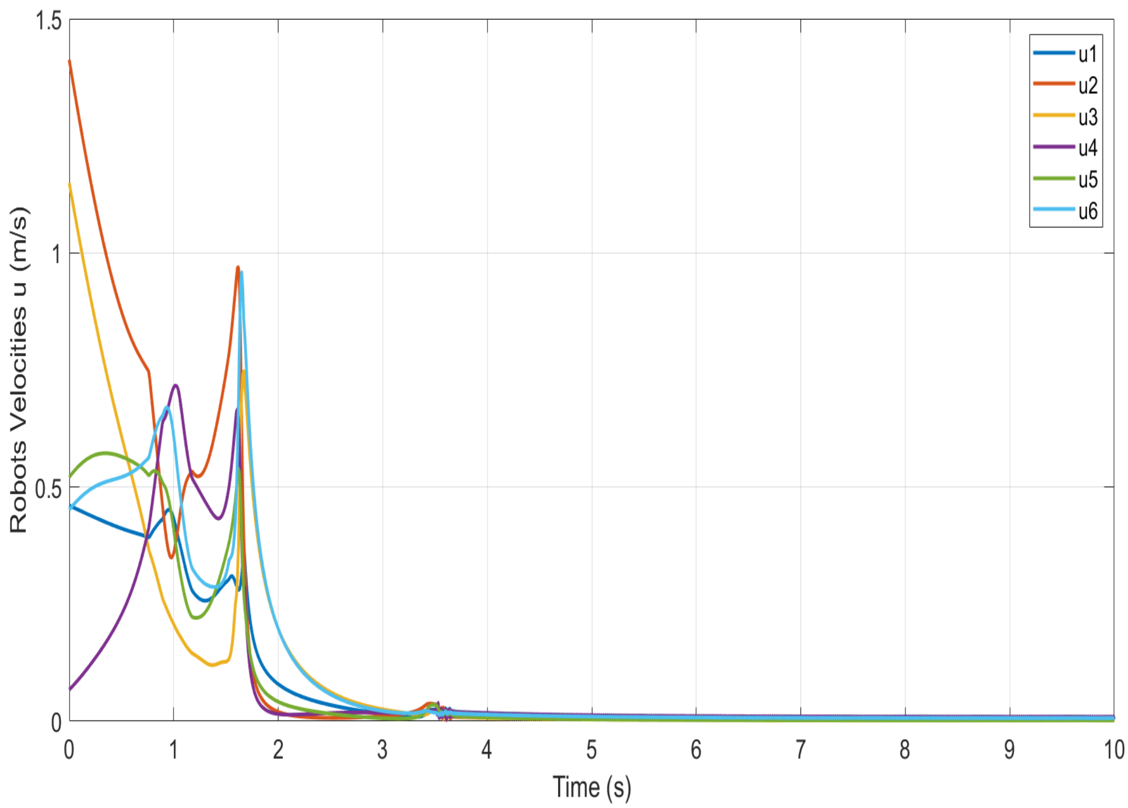


(b)

FIGURE 4.10: Velocity constraint 1: Fig. 4.10(a) Formation control of six robots under collision avoidance constraint with $u_{min} = -[1, 1, 1, ..., 1]$ and $u_{max} = [1, 1, 1, ..., 1]$ (equation (4.42)) . Fig. 4.10(b) Sharp changes in resultant velocities $(u_1 - u_6)$ indicate collision avoidance operation.

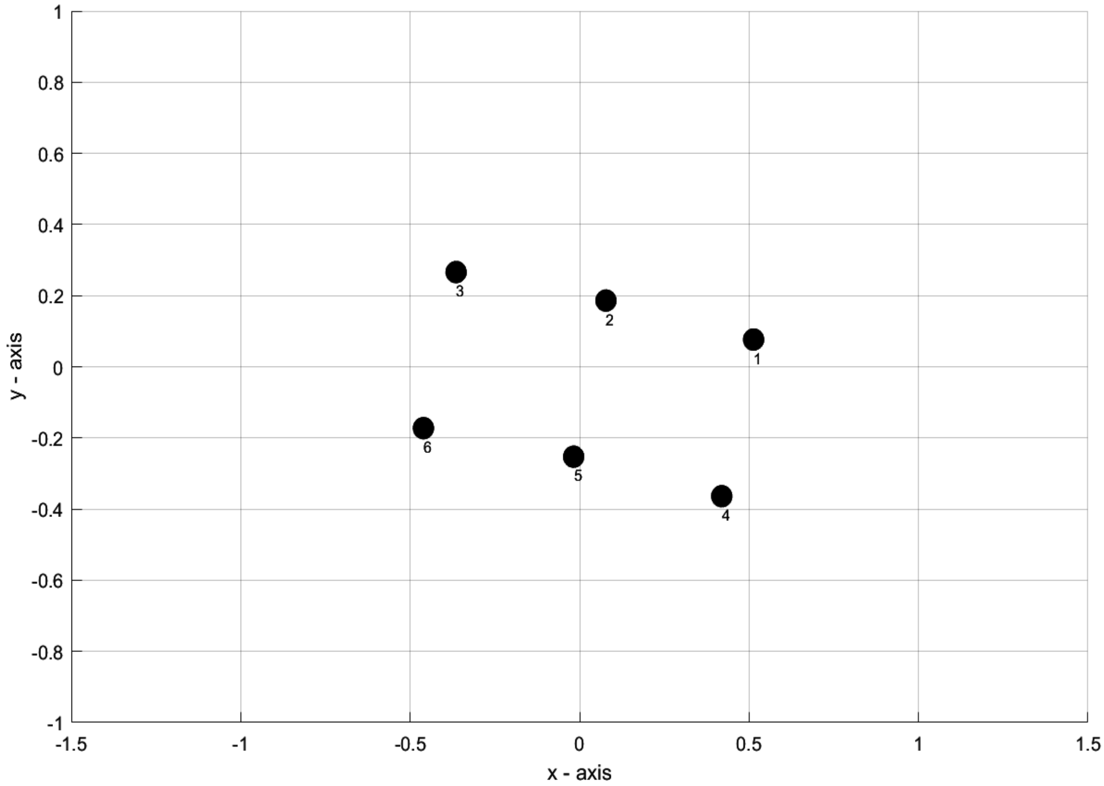


(a)

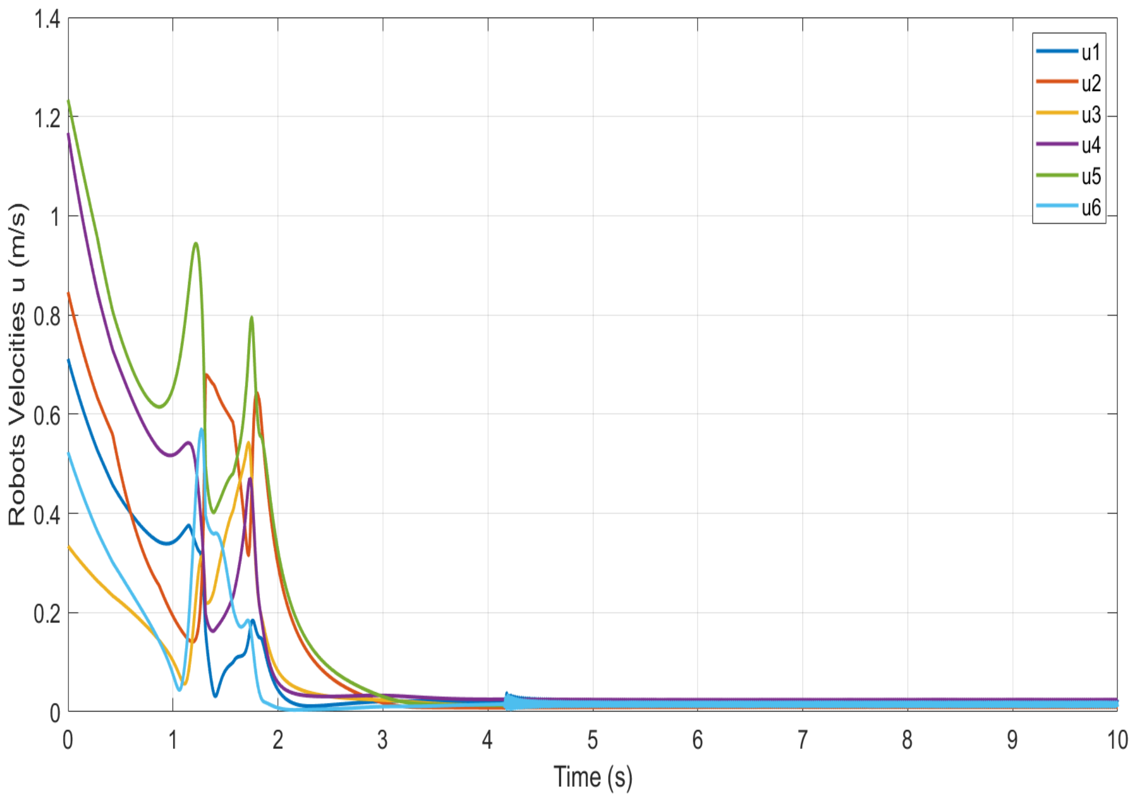


(b)

FIGURE 4.11: Velocity constraint 2: Fig. 4.11(a) Formation control of six robots under collision avoidance constraint with $u_{min} = -[1.5, 1.5, ..., 1.5]$ and $u_{max} = [1.5, 1.5, ..., 1.5]$) (equation (4.42)). Fig. 4.11(b) Resultant velocities ($u_1 - u_6$).

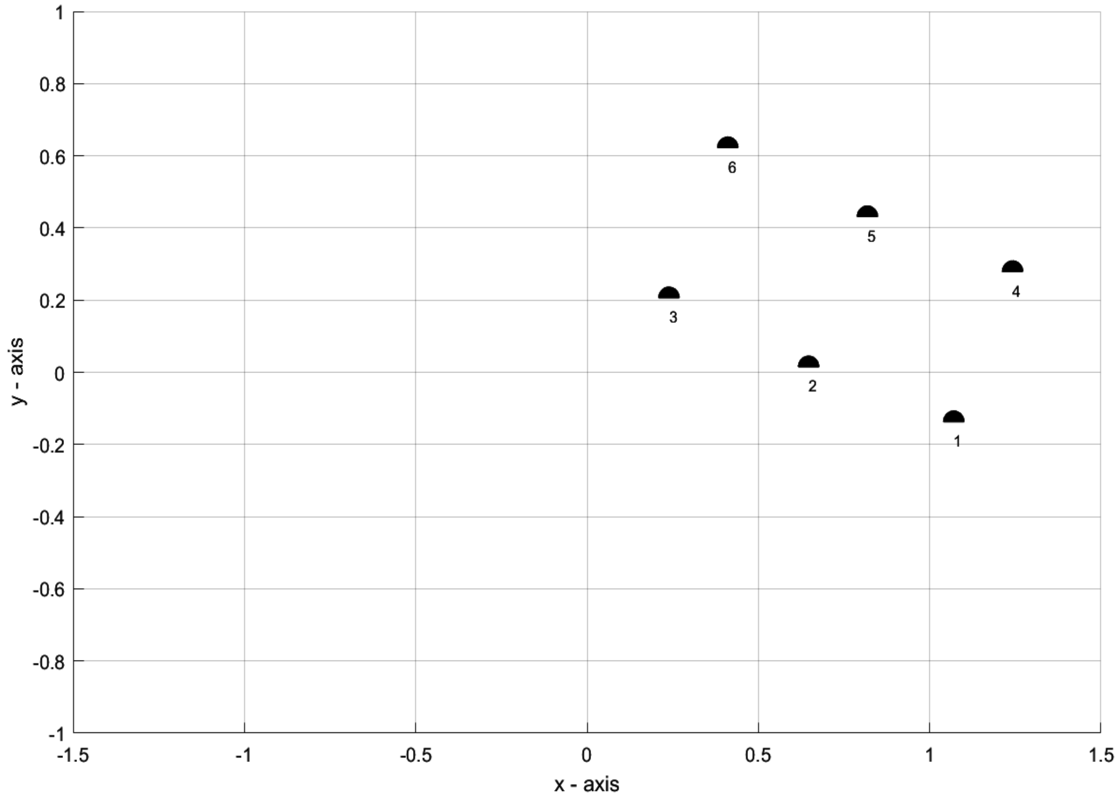


(a)

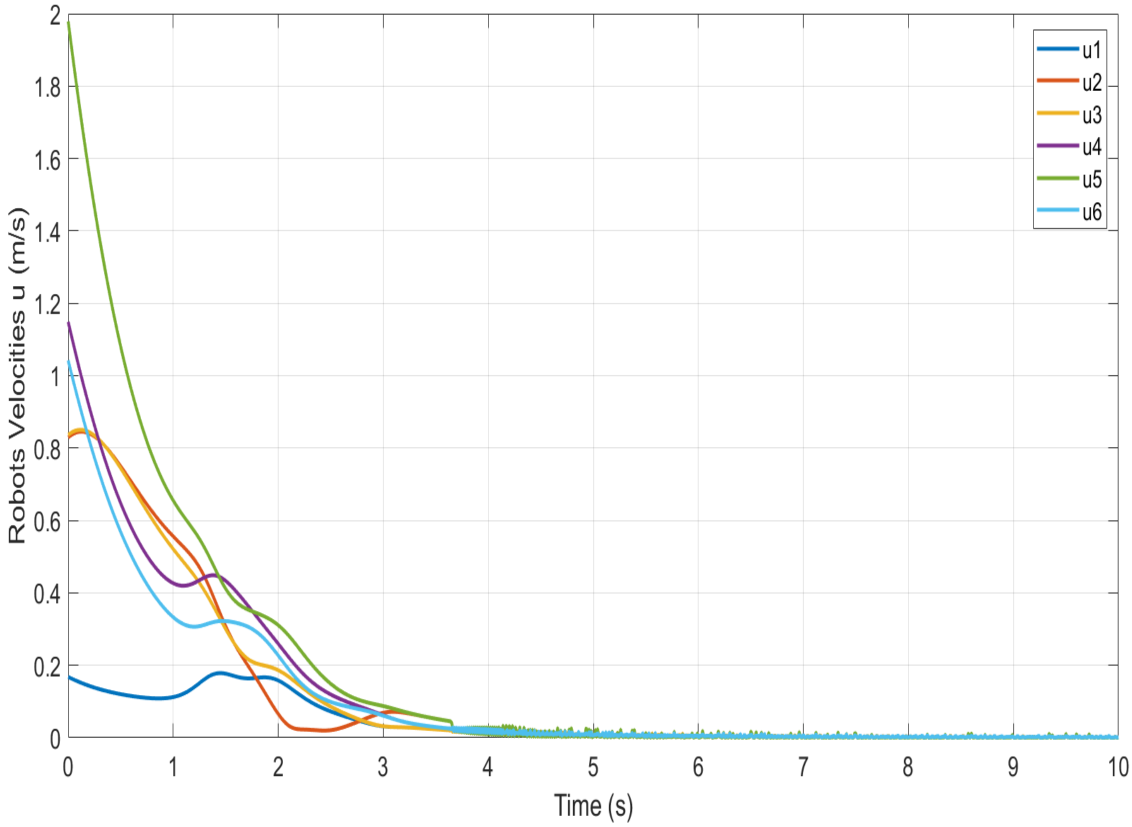


(b)

FIGURE 4.12: Velocity constraint 3: Fig. 4.12(b) Formation control of six robots under collision avoidance constraint with $u_{min} = -[2, 2, 2, ..., 2]$ and $u_{max} = [2, 2, 2, ..., , 2]$ (equation (4.42)). Fig. 4.12(b) Resultant velocities ($u_1 - u_6$).



(a)



(b)

FIGURE 4.13: Case III: Simulation results without constraints on velocities and collision avoidance.

4.4 Chapter Summary

In this chapter the task execution framework using the control barrier functions through constraint-oriented approach for a multi robot formation control alongwith simulation results is presented. A control strategy is designed for the development of an optimization based decentralized controllers using control barrier functions in order to achieve the tasks execution through a constraint-oriented approach. This constraint-oriented control design for the formation control of a multi robot system with constrained rectangular components of velocities under confined conditions is successfully simulated in Matlab©. During the attainment of desired formation, the constraints are enforced on the rectangular components of velocities and the desired formation shape remained attainable. The cost function for achieving the desired formation is embedded as the control barrier function based constraint of an optimization problem. The inter-robots collision avoidance is added through a separately designed control barrier function based collision avoidance as constraint of the same optimization problem. The results proved that robots achieved the desired formation in both cases as presented in section 4.3. Thus through the formation control simulation results, it is evident that the concrete base of decentralized optimization based controllers using control barrier functions is developed as part of constraints of an optimization problem. This effort provides a firm base for the development of the environment aware resilience of multi robot tasks.

Chapter 5

Environment-Aware Resilient Multi Robot Framework

5.1 Introduction

In this chapter, the system modeling and control design technique for the conduct of intended research are presented in detail alongwith simulation results. A scenario for the development of an environment-aware multi robot task allocation and accomplishment framework is established for an application purpose. Consider a situation in which a team of robots in a multi robot system is required to be deployed in a dynamically evolving environment to execute different tasks. Every robot in team has set of specialties such as wheels, camera, ultrasound sensors, carrying-capacity and environment sensing technologies like uphill/off-road/marshy area due to rain etc. Such specialties permit the robots to manifest different capabilities such as moving in a field, surveillance of surroundings, obstacles detection, carrying-goods and traction control on uphill/off-road/marshy/slippery surfaces. The assignment of tasks to these robots for their accomplishment need a collection of particular capabilities. These capabilities can be acquired through different combinations of the presented specialties by all the robots in a team. The successful accomplishment of every assigned task is subject to minimum quantity of robots having special capabilities to perform those tasks. In such kind of scenario, control design concerns are, first the allocation of tasks to the robots keeping in view the evolving environment and the required specialties to perform allocated tasks, second the accomplishment of the allocated tasks by yielding a suitable control input to each robot in the team based on the information induced by environment. The solution of these design concerns can yield an environment-aware resilient task execution framework as given in upcoming sections. In the following section the system modeling is discussed in detail. This system modeling helps to establish the solution for the problem statement of this research thesis.

5.1.1 System Modeling

The prime focus here is to design and develop a framework that produces a workable relationship between robots in team and tasks required to be carried-out by these robots while taking into account the heterogeneity of robots, the capability demands of assigned tasks and dynamically changing environment. A practicable relationship is required to be built among the specialties essentially needed by tasks as presented by robots in order to carry-out those assigned tasks. Such mappings or relationships are helpful in identifying unworkable situations as tasking a wheeled robot to carry-out a flying task can be thought of as an unworkable job. Robots have to perform tasks (Fig. 5.1(a)) and to perform those tasks robots need to have specialties (Fig. 5.1(b)). The Fig. 5.2 manifests tasks and associated spe-

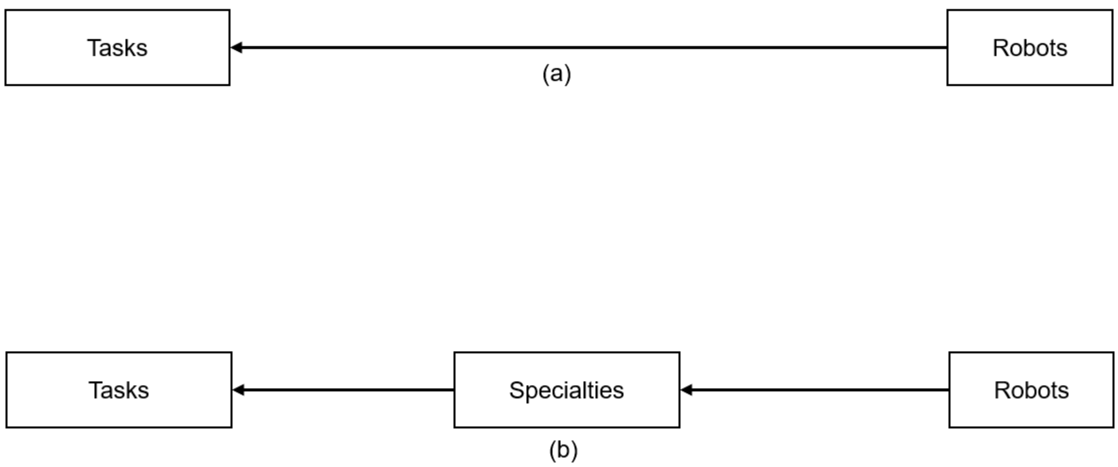


FIGURE 5.1: System modeling (a). robot-tasks relationship. (b). robot-specialties-tasks relationships.

cialties needed to carry-out the assigned tasks. The robot-specialty relationship builds a direct relationship between each robot and the specialties presented by the robot [16]. As illustrated in the Fig. 5.2 there are 4 robots labeled as r_1, r_2, r_3 and r_4. The robot r_1 has specialty C_1 with specialty components c_{11}, c_{12}, c_{13} and c_{14}. The specialty components c_{11}, c_{12}, c_{13} and c_{14} denote r_1's capabilities to perform tasks T_1, T_2, T_3 and T_4 respectively. Similarly, robots r_2, r_3 and r_4 have respective specialty components to perform T_2, T_3 and T_4 respectively. The specialty components may denote robot wheels, speed, camera, goods-carrying capacity, obstacles detection and traction-ability like all-wheel-drive or front-wheel-drive.

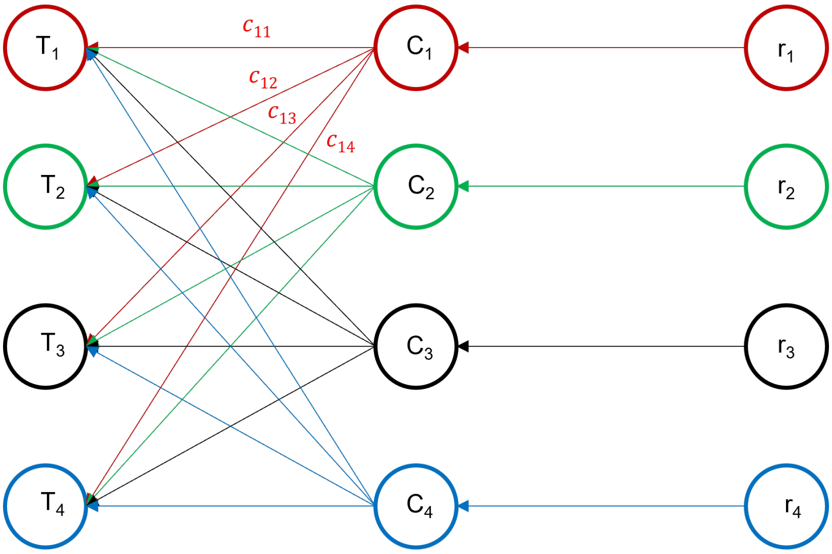


FIGURE 5.2: Robots - specialties - task relationships.

5.1.2 Specialty - Task Relationship

In a typical application, a team of robots is tasked simultaneously to complete assigned tasks whereas each task requires diverse types of specialties. Consider an example situation in which there is a task of transportation of certain products from one location to another location. To perform this task, we may require robot's specialties like goods carrying-capacity, traction or non-traction attributes of robots and also tracks or wheels for transportation. Here it is required to establish the relationship between specialties imparted by robots and tasks required to be performed by these robots. On left hand side of Fig. 5.2, tasks and corresponding specialties can be mapped. By observing the edges falling on T_1 in Fig. 5.2, one can find that specialties C_1, C_2, C_3 and C_4 can perform task T_1.

5.1.3 Robot - Specialty Relationship

As described earlier every robot in a team suitable for carrying-out tasks holds diverse specialties such as wheels, traction-ability or camera [98]. There is a need to define a relationship between robots and their presented specialties for carrying out tasks in a harmonized manner depending on these specialties. The essential prerequisites for a robot to select it as a potential candidate for a task depends upon capabilities imparted by their specialties and for the same purpose the specialization matrix C_i of robot i is required to be defined here. The specialty component c_{ij} of robot i for task j is mentioned below:

$$c_{ij} = \begin{cases} 1 & T_j = 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

If $c_{ij} = 1$ then robot i can execute task T_j, where $T_j = 1$ means that task j is executed. The specialization component c_{ij} can be changed on-the-fly.

5.1.4 Environment - Robots Relationship

The real world applications such as transporting of medicines by team of robots to specific affected destinations involve performing of different tasks in dynamically evolving environment. These destinations may get susceptible to changing/unforeseen environment like sudden rain, marshy soil, snow covered area, off-road and uphill drives. Robots in team have different specialties in terms of overcoming or responding such a changing environment. Switching of task to suitable robot requires incorporation of environment in framework because evolving, unknown and unaccounted for environment conditions can hamper the mission critical tasks. Such scenarios define that how certain tasks can be performed with resilience under evolving environment conditions. To account for environment in a robot centric framework, new mapping between robots specialty and environment as shown in Fig. 5.3 is required to be defined. However introduction of environment as an integral part of existing framework, requires new relationship to be inculcated that is environment-specialty relationship as shown in Fig. 5.3.

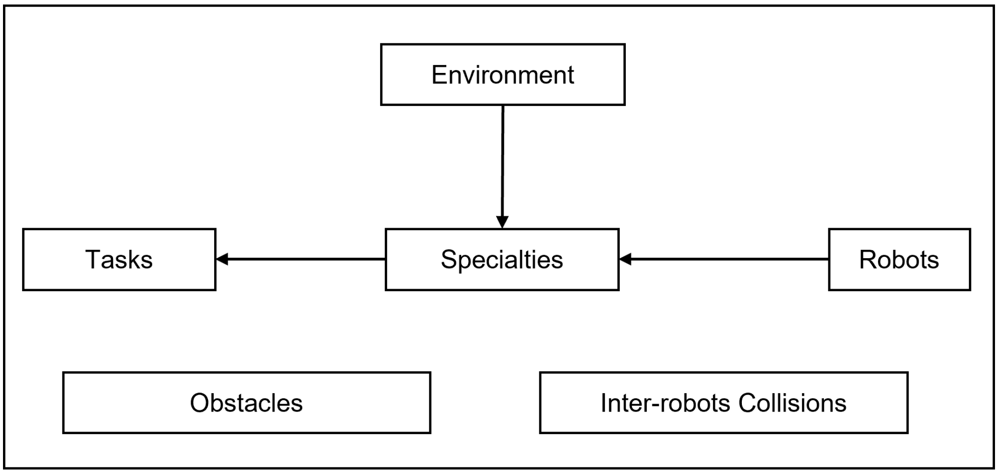


FIGURE 5.3: Proposed framework with environment-specialty relationship. Presence of obstacles and inter-robots collision avoidance manifest real-world scenario.

5.2 Control Design Strategy

The basis of environment-aware resilient multi robot system requires the development of a task execution framework using control barrier function through constraint-oriented approach in an optimal fashion with task prioritization. In forthcoming sections the control design development for an environment-aware resilient framework using the concepts as already mentioned in section 5.1.1 is elaborated in detail. The environment-aware resilience of multi robot tasks is proved through solving and then simulating the scenario of dynamic tasking for a multi robot system under optimal strategy in constrained conditions as will be presented in sections 5.3 and 5.4.

5.3 Proposed Environment-Aware Resilient Multi Robot Framework: Dynamic Tasking Problem

Multi robot task allocation and accomplishment are discussed in this section through a specific formulation. The scheme to be used for multi task accomplishment is presented here in terms of the extended set oriented tasks [83]. The extended set oriented task based scheme permits to define miscellaneous tasks designated as sets as briefly introduced in the previous section. Indeed, control barrier functions are successfully implemented to define numerous tasks for different robots specifically in robots team extending from the simultaneous control of multi robot upto accomplishment of the multi tasks in the robot based manipulators [16, 83]. Among these extended set oriented tasks, there is a specific type of multi robot tasks under the ambit of coordinated control that are performed through the realization of a cost function which is made asymptotically stable [7, 16]. Moreover in [16], the accomplishment of these tasks is carried out through a constraint oriented optimization problem - a unique scheme used for having prolonged operation of a robot [15]. To explain more on subject matter, suppose there is a function f which is continuously differentiable and function of the state q_i of a robot i. Moreover it is supposed that the dynamics of the robot are affine in control (equation (4.2)).

The accomplishment of the task specified by the minimization of function f can be resolved through a constraint-oriented optimization problem. The slack variable in an optimization problem is used for manifesting the extent upto which control barrier function based constraint can be relaxed meanwhile ensuring the feasibility of the defined optimization program as manifested below. In multi task and multi robot based situations, this optimization framework inherently permits robots to combine different constraints, whereas each constraint represent an assigned task into framework.

$$\begin{aligned} &\min_{u_i,\sigma_i} \|u_i\|^2 + \sigma_i^2 \\ &\text{subject to:} \\ &L_{f_v} f(q) + L_{g_v} f(q) u_i \geq -\rho(f(q)) - \sigma_i \end{aligned} \tag{5.2}$$

Where the assigned task is encoded by the constraint in which $f(q)$ is a control barrier function that renders the safe set S asymptotically stable. Here ρ is an extended class K_∞ function and σ_i is a slack variable which quantifies the extent to which the constraint can be relaxed ensuring that the optimization program remains feasible. In a multi task, multi robot based settings, this framework allows robots to combine multiple constraints, each representing a task, into a single framework. For a task n encoded in f_n, where $n \in \{1, ..., n_t\}$, the constraint-based optimization problem for the robot i can be written as:

$$\min_{u_i,\sigma_i} \|u_i\|^2 + \sigma_i^2 \tag{5.3}$$
$$\text{subject to:}$$
$$L_{f_v} f_n(q) + L_{g_v} f_n(q) u_i \geq -\rho(f_n(q)) - \sigma_{i,n}$$

Where $\sigma_i = [\sigma_{i,1}....\sigma_{i,n}]^T$ represents the slack variables corresponding to each task being executed by the robot i. The tasks encoded by the control barrier functions are not restricted to be dependent only on the state of the robot i, but rather on the ensemble state of the robots $q = [q_1^T, ..., q_{n_r}^T]^T$ thus allowing the framework to encompass coordinated multi robot tasks. With this framework in place, the slack variables σ_i present a natural way of encoding task priorities for the individual robots.

5.3.1 Dynamic Tasking for a Multi Robot System under Optimal Strategy in Constrained Environment

The resilience is the capability of a multi robot control algorithm based task execution framework to respond to specialty failures on the robots during the mission course in an evolving environment. There is a need to design a framework to render the heterogeneity of robots manifested in terms of their suitabilities at performing miscellaneous tasks under an environment-aware scenario. The idea of suitability originates out of the well defined notions of specialties and corresponding capabilities showcased by a robot. The benefit from the task accomplishment frameworks as mentioned in [6, 16] can be deliberated here and thus an attempt can be made in order to make our proposed framework resilient and adaptive through an optimization based technique under changing environment conditions. Consider an example scenario for the team of robots consisting of differently specialized robots which are tasked to reach a specific destination in the field. Unforeseen scenario such as sudden intense rain can cause soil to turn into loose marshy area and such situation may only be made negotiable through a different robot with the specific specialty such as traction-ability i.e., a four wheel drive robot. Thus an unforeseen failure of robot specialty appearing on the way can thwart the execution of the assigned task such as transportation of certain goods, henceforth making the robot invalid to handle this specific environment based situation. In such scenario the solution is to develop an environment-aware resilience that is the task reassignment to appropriate robot i.e., the task of transportation be switched to different robot having the specialty of traction-ability during the aforementioned evolving environment. In such cases, realtime updating of the robots specialization for performing a task to account for on-the-fly changes at each point in time is essentially needed.

The current framework of [16] is robot centric and does not include environment in framework whereas as illustrated in above example it is quite obvious that in actual scenarios like transportation of specific equipment by swarm to certain places in field, unexpected and harsh changes in environment may thwart the mission progress as certain robot specialties may become invalid. The defining of new mappings such as environment to specialty mapping as shown in Fig. 5.3 caters for change in specializations of certain robots towards performing the tasks under changing environment scenario. Such an environment-aware specializations of robots render on-the-fly switching of task to a suitable robot in the team in an evolving, unknown and unaccounted-for environment conditions thus overall achieving the desired resilience of tasks. To attain this resilience the optimization problem coupled with real-time updates in the environment-aware specializations towards performing of tasks as well as capabilities of the robots, can be resolved at each point in time. This scheme devises a feedback mechanism as shown in Fig. 5.4. An algorithm-update rule focused at altering the robot's environment-aware specialization at performing certain tasks in a team can be formulated. This law is based on calculated versus desired progress of robots at accomplishing the tasks that robots are tasked to perform. Moreover the scheme introduced in [99] is useful to learn more about uncertain conditions. However the change in environment is accounted for by changing the associated values in the environment based relationships. In the upcoming section, the environment-aware multi robot system is developed using dynamic framework in which robots can undergo several environmental effects on their specialties. Specialties are activated and deactivated under environment effect while robots continue performing the numerous tasks in an optimal fashion which is, optimizing the control inputs, tasks' prioritization and issuance related optimization variables. Different obstacles to robots are presented during the mission course. The developed environment-aware controller is designed keeping in view all the necessary constraints required for the task allocation and accomplishment. Here, under the influence of environment-

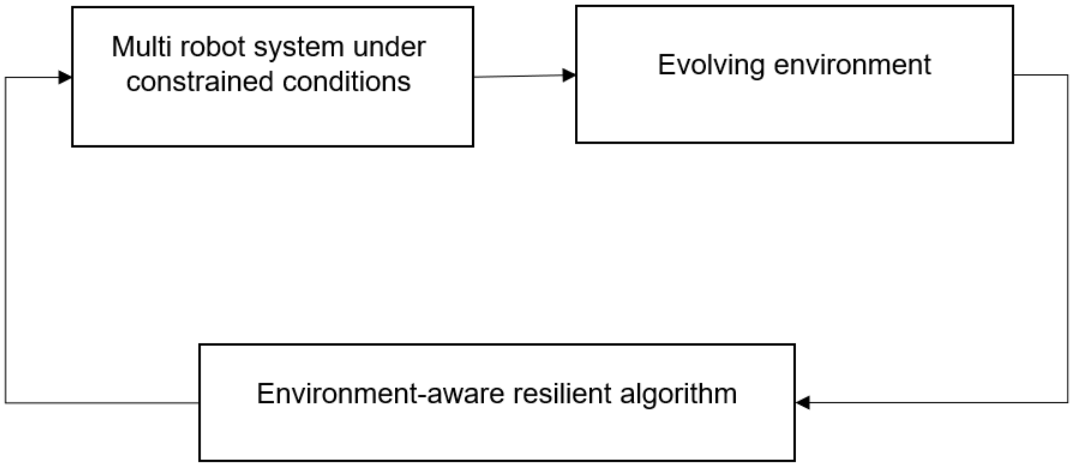


FIGURE 5.4: Proposed environment-aware multi robot system

specialty relationship defined in Fig. 5.3, it is required to redefine the framework of Fig. 5.2 by introducing robot to environment-oriented specialty relationship. For the purpose the environment oriented specialization matrix C_i^E of robot i with its components c_{ij}^E for a certain task j is introduced here. The detailed framework is depicted in Fig. 5.5. The generalized expressions for the environment module E and specialization matrix C_i^E are defined comprehensively in following subsections.

5.3.2 Conceptual Deliberation

The control design strategy for the environment-aware resilience of multi robot tasks is presented in this part of section. For this the tasks are dynamically switched under the influence of an evolving environment. As briefly mentioned in the literature review (section 2.2.3 of chapter 2), tasks are usually issued once at the initiation of a multi robot based mission to robots possessing specific specialties. During the mission course, under the environment effect a necessity can arise

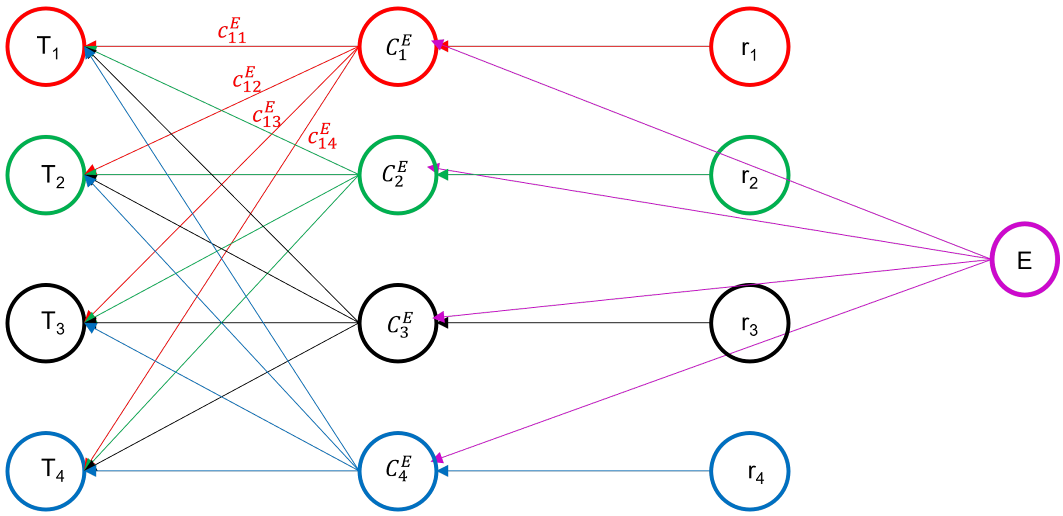


FIGURE 5.5: Robots - environment oriented specialties - task relationships. E is the environment module. C_i^E is an environment oriented specialization matrix of the robot i. c_{ij}^E is specialization component of robot i for a certain task j.

where these tasks issuance/withdrawal/re-issuance among robots may be required under the influence of active/inactive/re-active status of specialties.

A simultaneous mechanism for tasks issuance/withdrawal/re-issuance coupled with specialties should be developed while catering all such robots that can perform the tasks. The scenario merits corresponding activation/deactivation of specialties connected with the tasks issuance/withdrawal/re-issuance strategy. For this, a novel dynamic task-specialty matrix based mechanism termed as environment module is proposed and developed in this thesis and further incorporated into an optimization technique.

The environment module has environment coefficient vectors each associated with the task issuance strategy and specialties of the robots in the team. In addition to minimization between task re-issuance and current task preferences, the proposed optimization technique-based problem also minimizes robots constrained velocities and task-related slack variables, subject to the control barrier function based constraint which encodes desired objective cost function along with other constraints primarily required for an optimal task execution. The environment module in the optimization program let the robots issue/re-issue, active/inactive the specialties in an online way leading to the accomplishment of the tasks optimally. Capability of robot-to-robot collision avoidance and robot-to-obstacle avoidance is incorporated through separate control barrier functions. The environment aware resilience of multi robot tasks with three different cases such as (i) simultaneous task re-issuance and specialties activation/ deactivation under environment influence, (ii) mid-course withdrawal of tasks with active environment coefficients and (ii) mid-course degradation of environment coefficients with active task issuance, is devised in this thesis.

The above designed optimization technique-based decentralized controllers let the robots safely accomplish the respective tasks optimally under constrained velocities as same is showcased in the section 5.4 of simulation results. Thus the research work provide an evidence of working of our proposed technique developed for an environment-aware resilience of multi robot tasks.

5.3.3 General Problem Formulation

As considered in section 4.2.1 of chapter 4, here a single integrator dynamical system is deliberated as well that is, $\dot{q} = u$, where q is the state of the system, u is the control input $q \in \mathbb{R}^n$, $u \in \mathbb{R}^n$, $\mathbb{R}$ denotes the set of real numbers, n being the dimension [90]. The desired goal is to minimize a cost function, say, $J(q)$ where $J : \mathbb{R}^n \to \mathbb{R}_+$, and $\mathbb{R}_+$ denotes set of positive numbers on a real line. A function $f(q) = -J(q)$ classified as barrier function can be defined here, where $J(q) \geq 0$. The zero super level set S is called as a safe set, where S is given by:

$$S = \{q \mid f(q) \geq 0\} \tag{5.4}$$

Here $f(q)$: $\mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. If the following setting on derivative of $f(q)$ is satisfied then $f(q)$ denotes a control barrier function.

$$\dot{f}(q) = \frac{\partial f(q)}{\partial q} u \geq -\rho(f(q)) \tag{5.5}$$

In above equation, term $\rho :\ \mathbb{R} \to \mathbb{R}$ is an extended class k_∞ function, defined on entire real line: $\mathbb{R} = (-\infty, \infty)$ [50]. The solution of the optimization problem of (5.6) solves the problem of minimization of cost $J(q)$ by driving the state q to a stationary point $\hat{q}$ of cost $J(q)$ such that $\frac{\partial J(\hat{q})}{\partial q} = 0 \Longleftrightarrow \hat{q} = 0$ [22]:

$$\begin{aligned} &\min_{u} \ ||u||^2 \\ &\text{subject to:} \\ &\frac{\partial f(q)}{\partial q} u \geq -\rho(f(q)) \end{aligned} \tag{5.6}$$

The optimization program of (5.6) can be used for the execution of the desired task. It is a constraint-driven approach as already discussed in previous chapter.

In order to execute T different tasks denoted as $T_1, ..., T_n$ with varying preferences where tasks are indexed from 1 to n_t, a slack variable $\sigma \in \mathbb{R}$ in the optimization problem can be introduced in order to set the preferences of assigned tasks among n_r robots in a multi robot system that is to execute a task with more preference than others. The σ denotes the limit to which the constraint on a task can be defied.

The desired optimization program for performing n_t tasks by robot i in a multi robot system consisting of n_r robots is given by [16]:

$$\begin{aligned} &\min_{u_i} \ ||u_i||^2 + ||\sigma_i||^2 \\ &\text{subject to:} \\ &\frac{\partial f_{i,n}(q)}{\partial q_i} u_i \geq -\rho(f_{i,n}(q)) - \sigma_{i,n} \\ &||\sigma_i|| \leq \sigma_{max} \end{aligned} \tag{5.7}$$

Here, $f_{i,n}(q)$ is one-to-one function between a robot i and the destination d_n, $\sigma_i = [\sigma_{i,1}, ..., \sigma_{i,n_t}]^T$ denotes the set of limits on slack variable σ for a robot i against n_t tasks and σ_{max} is the maximum permissible limit of introduced slack variable.

The optimization problem of expression (5.7) produces control input u_i at each point in time in order to drive the robot to perform the task n depending upon the task preference, where $n \in \{1, ..., n_t\}$. The following optimization program of (5.8) subject to task and slack constraints, generates the minimized desired control input u_i in order to optimally allocate the tasks among robots in a team of robots, while assuring that robot i performs the task T_n with highest preference [16]:

$$\min_{u,\sigma,\phi} \tau\, ||\psi^d - \psi||^2 + \sum_{i=1}^{n_r} ||u_i||^2 + ||\sigma_i||^2_{C_i} \tag{5.8a}$$

subject to:

$$\frac{\partial f_{i,n}(q)}{\partial q_i} u_i \geq -\rho(f_{i,n}(q)) - \sigma_{i,n} \tag{5.8b}$$

$$\frac{\sigma_{i,m}}{\zeta} \geq \sigma_{i,n} - \sigma_{max}(1 - \phi_{i,n}) \tag{5.8c}$$

$$\mathbf{1}^T \phi_i = 1 \tag{5.8d}$$

$$||\sigma_i|| \leq \sigma_{max} \tag{5.8e}$$

The above optimization program minimizes the difference between desired task specification (ψ^d) and current task preference (ψ) where τ is scaling constant, provides trade-off between meeting the desired task specifications and current task preferences.

The desired tasks specifications ψ^d in the optimization program of (5.8a) is given by:

$$\psi^d = [\psi_1^\theta,\, \psi_2^\theta, ..., \psi_{n_t}^\theta]^T \tag{5.9}$$

The term ψ_n^θ denotes desired fraction of robots for executing task T_n with highest priority. The current tasks vector ψ in optimization program of (5.8a) is given by:

$$\psi = \frac{1}{n_r}[\psi_1,\, \psi_2, ..., \psi_{n_r}]\phi \tag{5.10}$$

The term ψ ignores the outcome against a robot to take a task T_n if it has no eligibility to perform the task T_n and this is achieved by projecting the vector of preferences ϕ_i in the column space of specialization matrix C_i of a robot i [16].

The term $\psi_i = C_i C_i^p$ where C_i^p is the Penrose inverse of robot i specialty. The specialization matrix C_i of a robot i is given by:

$$C_i = diag(c_{i,1}, ..., c_{i,n}) \tag{5.11}$$

For executing task T_n, the specialty $c_{i,n}$ of a robot i at performing the task T_n should hold the relationship of $c_{i,n} > c_{i,m}$ which means that the robot i is better at performing the task T_n with specialty $c_{i,n}$ than the task T_m with its specialty $c_{i,m}$. The term $||\sigma||^2_{C_i}$ in optimization program of (5.8a) describes assigning weights to each component of σ proportionally to the corresponding entry in the specialization matrix of the robot that is, $||\sigma||^2_{C_i} = \sigma^T C_i \sigma$ [16]. Here if the task for each robot is to reach desired destination in a plane then the cost f_n can be taken in the optimization program of (5.8b) as follows [16]:

$$f_n(q) = -||q - d_n||^2 \tag{5.12}$$

Where d_n is the desired point (destination) and q is the current state of the robot in two-dimensional plane. Moreover $m \neq n$, and m, $n \in \{1, ..., n_t\}$. The equation (5.8c) manifests task T_n can be performed with more preference than task T_m for larger value of ζ while ensuring that the following relationship holds:

$$\phi_{i,n} = 1 \Rightarrow \sigma_{i,n} \leq \sigma_{i,m} \,/\, \zeta, \forall n \neq m \tag{5.13}$$

Where ζ is a scaling value and allows to encode relative effectiveness among different tasks such that $\zeta > 1$. In equation (5.8d), the term ϕ_i is vector of task preferences for robot i and $\phi_i = [\phi_{i,1}, ..., \phi_{i,n_t}]^T$. If $\phi_{i,n} = 1$ then task T_n has the highest preference for robot i than task T_m and at any point in time only one term in ϕ_i is valued as 1, thus we can write $\mathbf{1}^T\phi_i$=1. The vector containing the task preferences for entire multi robot system is $\phi = [\phi_1^T, \phi_2^T, ..., \phi_{i,n_r}]^T$ [16].

5.3.4 Problem Formulation: Dynamic Task Re-issuance and Specialty Reassignment

Consider a multi robot system consisting of such robots which can perform all the issued tasks because these robots possess specialties to perform those tasks. Here the robot specialty can be a robot's carrying-capacity, maneuvering technology such as wheels or traction-ability etc. The robot task can be delivery of goods to certain specific destinations in the field using its specialties. The issuance of tasks to the team robots is usually carried once prior the start of multi robot based mission which is unable to be altered during the mission mid-course. A need can arise where it is mandatory to re-issue tasks to robots during the mid-course and simultaneously reassign specialties among robots, because of the typical geographical situations such as marshy, snowy, uphill or off-road as being faced by robots while performing a task such as carrying certain goods to respective destinations or performing surveillance, etc. For this it is very much required to have a robot capability-control with sensing ability, installed on each specialty of these well-equipped robots. This capability-control with sensing-ability can sense the surrounding environment/terrain, update real-time about the effectiveness of the robot's specialty and activate/deactivate/re-activate a robot specialty.

An environment module E as depicted in Fig. 5.6 is incorporated in order to manipulate such capability-controls of the robots in a multi robot system. The environment module E is able to control the issuance/withdrawal/re-issuance of tasks and can further activate/deactivate specialties of all the robots by manipulating each robot's capability-control as depicted by Fig. 5.7 through environment coefficients vector e_i of each participating robot. This capability-control corresponding to a single robot in team can be termed as the environment coefficients vector e_i of a robot i. Each element of e_i always corresponds to the specific specialty of the robot. Consider another scenario in which the degradation of a robot's sensing-ability to a certain threshold during the mission course makes a robot unaware of its surrounding environment. For the first scenario mentioned in the preceding paragraph, it is needed to incorporate an environment module E in the optimization problem of (5.8). For the second scenario, multi robot resilient algorithm at each time step evaluates the $\triangle E$ where $\triangle E = E(t+dt) - E(t)$ that is, calculation of the difference in E at each simulation timestep dt. If t is simulation time then the environment module E for a multi robot system consisting of n_r robots is given by:

$$E(t) = [e(t)_1^T,\, e(t)_2^T, ..., \, e(t)_{n_r}^T]^T \tag{5.14}$$

$$E(t+dt) = [e(t+dt)_1^T,\, e(t+dt)_2^T, ..., \, e(t+dt)_{n_r}^T]^T \tag{5.15}$$

The term e_i is environment coefficients vector for robot i where e_i is given by:

$$e_i = [e_{i,1}, e_{i,2}, ..., e_{i,n_t}]^T, \;\, e_{i,n} \in \{0, 1\} \tag{5.16}$$

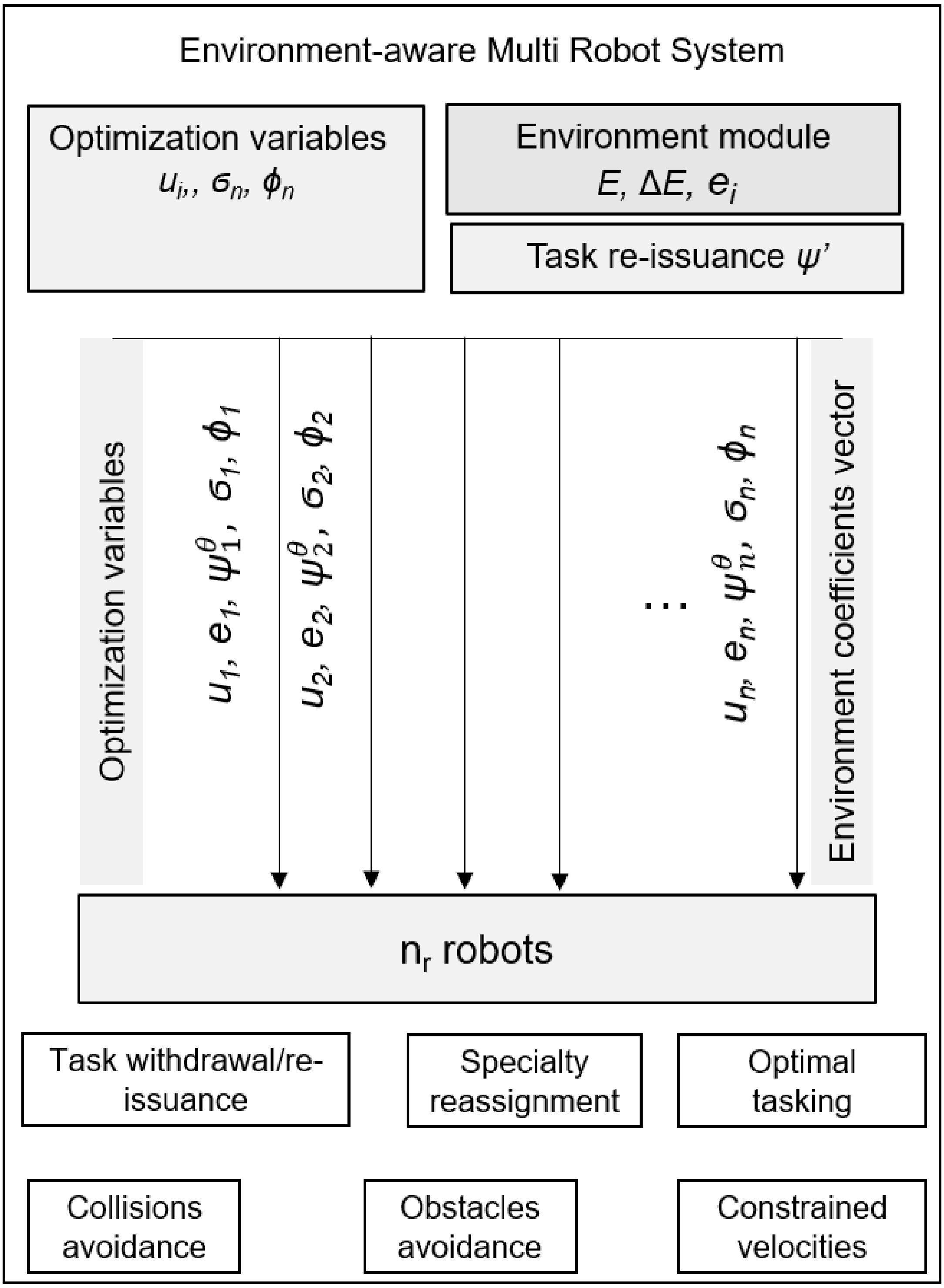


FIGURE 5.6: Top view of proposed system's model.

The term $e_{i,n}$ in above equation is an environment variable of robot i against the task n. The specialty matrix C_i against robot i as given by equation (5.11) can be modified for obtaining an environment-aware specialty matrix (C_i^E) as defined by following equation:

$$C_i^E = diag\left((e_i)^T\right) C_i \tag{5.17}$$

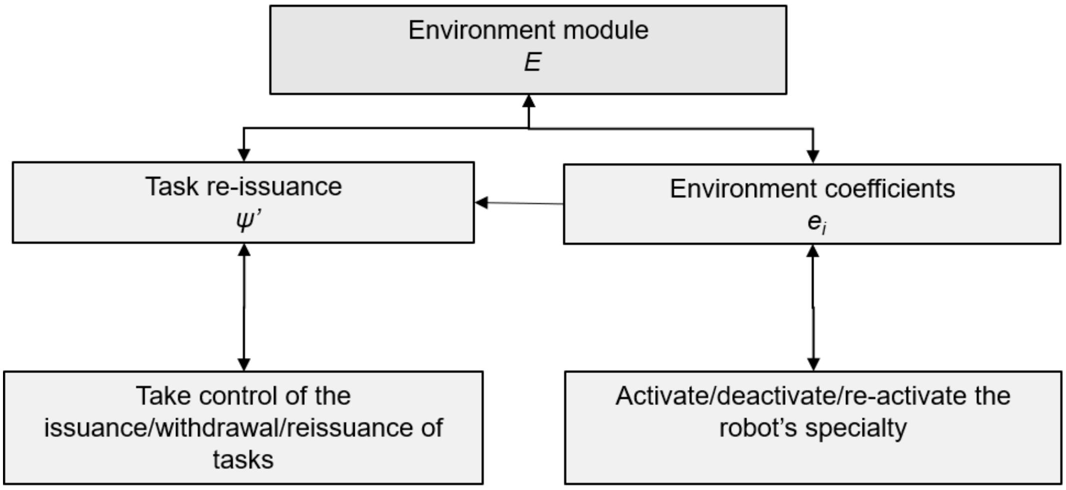


FIGURE 5.7: Schematics of environment module. Bidirectional arrows depict information flow.

The value of every element of environment coefficients vector for robot i can also activate or deactivate robot's specialties in addition to having an intelligent sensing ability.

Each element of environment vector can have one value at a given time that is, either 0 or 1 depending upon situation presented to the multi robot system during the mission course. In case of the degradation of a robot sensing-ability of the capability-control (environment coefficients vector), the corresponding environment coefficient variable manifests the same at each time step dt according to following well known exponential decay formula:

$$e_{i,n}^{t} = e_{i,n}^{t_o}(1-a)^{t} \tag{5.18}$$

Here $n \in \{1, ..., n_t\}$, term $e_{i,n}^{t}$ is value of $e_{i,n}$ at current time t and term $e_{i,n}^{t_o}$ is value of $e_{i,n}$ at initial time t_o. The constant a is the rate of depreciation or appreciation in the value of $e_{i,n}$ with time. Thus a rule can be designed for the environment module to calculate the environment variable $e_{i,n}^{t}$ at each time step and compare its value against its previous value that is, if at certain time t any element of the environment vector depreciates by a (say 30%), then the value of $e_{i,n}$ may be dictated by following rule as follows:

$$e_{i,n}^{t} = \begin{cases} 0 & \text{if} \ \ a \geq 0.3 \\ 1 & \text{otherwise} \end{cases} \tag{5.19}$$

The task re-issuance matrix can be defined by modifying equation (5.9) as given by:

$$\psi(t)^{'} = \frac{1}{\psi_1^{\theta}(t) + \psi_2^{\theta}(t) + ... + \psi_{n_t}^{\theta}(t)}(diag((\psi^{\theta}(t))^T ..., diag((\psi^{\theta}(t))^T))_{1\times 2n_t} E_{2n_t \times 1} \tag{5.20}$$

In above equation, the fraction $\frac{1}{\psi_1^{\theta}(t)+\psi_2^{\theta}(t)+...+\psi_{n_t}^{\theta}(t)}$ denotes desired fraction of robots that need to perform task T_n and term $\psi_n^{\theta} \in \{0, 1\}$ denote the issuance status of task T_n where $\psi_n^{\theta} = 1$ means that the task is issued and $\psi_n^{\theta} = 0$ means the task is either not issued or withdrawn.

The modified current task specification ψ^* as compared to equation (5.10) is given by:

$$\psi^* = \frac{1}{n_r}[\psi_1^*,\ \psi_2^*, ..., \psi_{n_r}^*]\phi \tag{5.21}$$

The term $\psi_i^* = C_i^E C_i^{E+}$ where C_i^{E+} is Penrose inverse of robot $i's$ specialty. Following optimization program contains dynamic task re-issuance and specialty reassignment capability.

$$\min_{u,\sigma,\phi} \tau\, ||\psi^{'} - \psi^*||^2 + \sum_{i=1}^{n_r} ||u_i||^2 + ||\sigma_i||^2_{C_i^E} \tag{5.22a}$$

subject to:

$$\text{(5.8b) to (5.8e)} \tag{5.22b}$$

$$u_{imin} \leq u_i \leq u_{imax} \tag{5.22c}$$

Due to mechanical limitations of robots or mission related velocity compulsions, constraining of robots velocities is required during the mission course for smooth accomplishment of tasks, the optimization program of (5.8) can be augmented by imposing constraints on each robot's rectangular components of velocities as shown in equation (5.22c).

The constraints are denoted by the minimum velocity lower bound $u_{min} \in \mathbb{R}_{-}$ and maximum velocity upper bound $u_{max} \in \mathbb{R}_{+}$. In subsequent sections during the implementation, the u_{min} and u_{max} can be written in terms of rectangular components u_x and u_y of resultant velocity vector u for each robot denoted as u_{xmax}, u_{xmin}, u_{ymax} and u_{ymin} respectively. Here u_{xmax}, u_{ymax}, u_{xmin}, u_{ymin} are maximum and minimum values of rectangular components of robot velocities in a two dimensional plane(x, y) and $u_{xmax},\ u_{ymax} \in \mathbb{R}_+,\ u_{xmin},\ u_{ymin} \in \mathbb{R}_-$. The values are taken in such a manner so that the quadratic program of (5.22) remains feasible. For a team of n_r robots, the rectangular components based velocity vectors u_{min} and u_{max} are given by:

$$u_{i}{}_{min} = [u_{1xmin},\ u_{1ymin},\ u_{2xmin}, ..., u_{n_r xmin},\ u_{n_r ymin}]^T \tag{5.23}$$

$$u_{i}{}_{max} = [u_{1xmax},\ u_{1ymax},\ u_{2xmax}, ..., u_{n_r xmax},\ u_{n_r ymax}]^T \tag{5.24}$$

5.3.5 Deriving General Expressions for Obstacle and Inter-Robots Collision Avoidance

The goal of the obstacle and collisions avoidance strategy is to allow the robots to execute a task while ensuring that they maintain minimum threshold distance with the obstacles and robots while coming in their vicinity, allowing the task to be finally executed [24, 27]. The desired behavior is enforced through separate constraints formulated by introducing two additional control barrier functions denoted as $o(q)$ and $c_2(q)$ such that $o(q) \geq 0$ and $c_2(q) \geq 0$. The zero super level sets S_3 and S_4 are additional safe sets in addition to S defined for optimization problem of (5.6) where $S_3 = \{x \mid o(q) \geq 0\}$, $S_4 = \{x \mid c_2(q) \geq 0\}$. The functions $o(q)$ and $c_2(q)$ are continuously differentiable functions and $o,\ c_2\ :\ \mathbb{R}^n \to \mathbb{R}$. If the following conditions are satisfied then o and c_2 are called control barrier functions.

$$\dot{o}(q) = \frac{\partial o(q)}{\partial q} u \geq -\eta(o(q)) \tag{5.25}$$

$$\dot{c_2}(q) = \frac{\partial c_2(q)}{\partial q} u \geq -\kappa(c_2(q)) \tag{5.26}$$

The designed control barrier functions describe the sets in the state space in which the robots are able to avoid collisions and obstacles when the need arises. Here $\eta,\ \kappa\ :\ \mathbb{R} \to \mathbb{R}$ are extended class k_∞ functions [50]. In order to ensure the avoidance of collisions and obstacles, these sets S_3 and S_4 are required to be kept forward invariant [50]. To avoid obstacles and inter-robots collisions, a robot i has to maintain minimum safe distances d_o from obstacles and d_c from its neighborhood robots such that d_o, $d_c \in \mathbb{R}_+$ [5]. For robot i, the obstacle avoidance cost is given by $o_i(q)$ and collision avoidance cost is given as $c_{2i}(q)$. The associated safe sets K_{io} and K_{2ij} are zero super level sets of function $o(q)$ and $c_2(q)$ respectively as given by:

$$K_{io} = \{o_i(q) = ||q_i - o_o||^2 - d_o^2 \geq 0\} \tag{5.27}$$

$$K_{2ij} = \{c_{2i}(q) = ||q_i - q_j||^2 - d_c^2 \geq 0\} \tag{5.28}$$

where d_o in equation (5.27) denotes minimu threshold distance from an obstacle and if n_o denotes maximum number of obstacles then subscript $o \in \{1, ..., n_o\}$ and $i,\ j \in \{1, ..., n_r\}$, $i \neq j$.

The Fig. 5.8 shows four robots r_1, r_2, r_3 and r_4 are assigned tasks T_1, T_2, T_3 and T_4 respectively for surveillance and delivery of goods at destinations d_1, d_2, d_3 and d_4 while avoiding inter-robot collisions and obstacles. These robots have traction ability for crossing marshy places, camera for surveillance, goods carrying capacity, etc. Under the environment effect such as intense rain at $t = t_1$, r_2 and r_3 can not make progress towards d_2 and d_3 due to loss of traction ability on marshy place caused by rain. The corresponding environment variables e_2 and e_3 updates environment module E. Environment-aware resilient algorithm ensures tasks switching that is, T_1, T_2, T_3 and T_4 are now performed by r_2, r_1, r_4 and r_3 respectively.

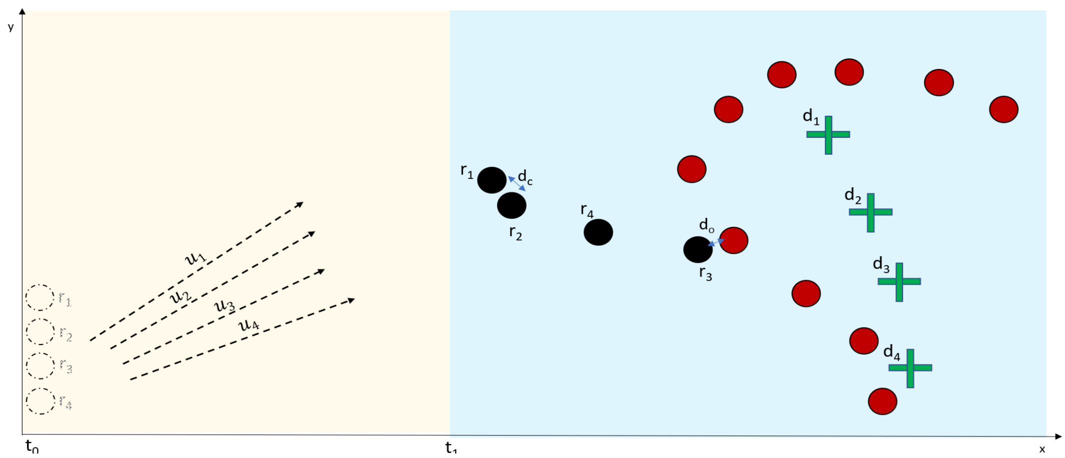


FIGURE 5.8: Environment-aware dynamic tasking. Four robots (black circles) r_1, r_2, r_3 and r_4 in a plane start heading towards destinations d_1, d_2, d_3 and d_4 at $t = t_0$. d_c and d_o manifest minimum threshold distances at which the collision avoidance operation with robots and obstacles (red circles) starts. Unforeseen change in environment (blue region) at time $t = t_1$ causes robots to switch the tasks.

5.3.6 Task Re-issuance and Specialty Reassignment with Obstacle and Collisions Avoidance

Multiple control barrier functions based constraints can be combined in an optimization problem as formulated in [24] and [91]. Thus the control barrier functions based constraints given in optimization program of (5.22) can be combined with equations (5.25) and (5.26) to get the mathematical optimization program of (5.29) for constrained execution of tasks with tasks re-issuance and specialty reassignment strategies. The optimization based formulation of (5.29) enable the robots to avoid collisions and obstacles on way while accomplishing the desired objectives optimally under constrained velocities.

$$\min_{u,\sigma,\phi} \tau \, ||\psi^{'} - \psi^{*}||^2 + \sum_{i=1}^{n_r} ||u_i||^2 + ||\sigma_i||^2_{C_i^E} \tag{5.29a}$$

subject to:

$$\frac{\partial f_{i,n}(q)}{\partial q_i} u_i \geq -\rho(f_{i,n}(q)) - \sigma_{i,n} \tag{5.29b}$$

$$\frac{\partial o_i(q)}{\partial q_i} u_i \geq -\eta(o(q)) \tag{5.29c}$$

$$\frac{\partial c_{2i}(q)}{\partial q_i} u_i \geq -\kappa(c_2(q)) \tag{5.29d}$$

$$\frac{\sigma_{i,m}}{\zeta} \geq \sigma_{i,n} - \sigma_{max}(1 - \phi_{i,n}) \tag{5.29e}$$

$$\mathbf{1}^T \phi_i = 1 \tag{5.29f}$$

$$u_{i_{min}} \leq u_i \leq u_{i_{max}} \tag{5.29g}$$

$$||\sigma_i|| \leq \sigma_{max}$$

The collisions and obstacles avoidance operation starts as soon as the robots either reach within a predefined minimum threshold distance from the robots in their vicinity or obstacles they encounter where they stop executing the task, perform collisions and obstacles avoidance operations. Once the robots avoid the collisions and escape the obstacles, they return to accomplish issued tasks to complete the desired objectives while being constrained by defined control barrier functions. This is achieved by executing the controller from the following quadratic program [5]. The term optimal refers to the dynamic assignment and prioritization of multiple tasks across a team of robots in a way that optimizes three key objectives: 1) Minimizing energy consumption that is, robots optimize their control inputs (u_i), 2) Optimize slack variables (σ_i) which allow robots to relax task constraints, enabling flexible prioritization that is, smaller $\sigma_{i,n}$ means robot i prioritizes task n, 3) Optimize vector of task preferences ϕ_i, that is, $\phi_{i,n} = 1$ means robot i prioritizes task n, 4) Satisfying task re-issuance matrix ψ', that is the allocation must match a current distribution of tasks ψ^*.

5.3.7 Quadratic Form for Environment Aware Resilient Framework

As already mentioned that the single integrator dynamics is considered where each robot can be controlled through velocity [6]. If q is the position of robot then we have the following first order dynamics of each robot in a multi robot system:

$$\dot{q}_i = u_i \tag{5.30}$$

Here i, $q_i \in \ R^2$ and $u_i \in R^2$ denote the robot i, position of robot i and velocity of robot i respectively, in a multi robot system consisting of n_r robots and $i \in \{1, 2, 3, ..., n_r\}$. If $f_{i,n}(q)$ is one-to-one function between robot i and destination d_n where d_n is point in $(x,\, y)$ plane [16].

$$f_{i,n}(q) = -||q_i - d_n||^2 \tag{5.31}$$

The derivative of $f_{i,n}(q)$ is given by:

$$\frac{\partial f_{i,n}(q)}{\partial q_i} \quad = -2(||q_i - d_n||) \tag{5.32}$$

Here u_{ix} and u_{iy} manifest the components of velocity u_i in two dimensional space (Fig. 4.2). The above minimization problem (5.29) becomes a mathematical optimization problem, further categorized as quadratic program [50, 92]. Standard form of a quadratic program based controller for solving the optimization problem of (5.29) is rewritten in the quadratic program (5.33). In (5.33), H is a symmetric matrix, F is a real valued vector, A and b are matrix and vector from equation (5.29). Moreover in constraints of (5.29g) u_{min} and u_{max} are lower and upper bounds for rectangular velocity components (u_x,u_y) against each robot's velocity vector u in two dimensional plane (x, y). Both u_{min} and u_{max} are $1 \times 2n_r$ real valued vectors.

$$
\begin{aligned}
&\underset{u}{\text{argmin}}\ \frac{1}{2}u^T H u + F^T u \\
&\text{subject to:} \\
&Au \leq b \\
&A_{eq}u = b_{eq} \\
&u_{min} \leq u \leq u_{max} \\
&0 \leq \sigma \leq \sigma_{max}
\end{aligned}
\tag{5.33}
$$

Here, A and b are from optimization problem of (5.29) and given by as follows:

$$
A = \begin{bmatrix} -2(||q_i - d_n||) + \sigma_{i,n} \\ -2(q_i - o_b) \\ -2(q_i - q_j) \\ -\frac{\sigma_{i,m}}{\zeta} + \sigma_{i,n} + \sigma_{max}\phi_{i,n} \end{bmatrix}, \quad \forall i \neq j
$$

and

$$
b = \begin{bmatrix} \rho(-||q_i - d_n||^2) \\ \eta(||q_i - o_n||^2 - d_b^2) \\ \kappa(||q_i - q_j||^2 - d_c^2) \\ \sigma_{max} \end{bmatrix}, \quad \forall i \neq j
$$

Here, A_{eq} and b_{eq} from equation (5.29f) are given by:

$$
A_{eq} = \begin{bmatrix} \mathbf{1}^T \phi_i \end{bmatrix}
$$

and

$$
b_{eq} = 1
$$

The bounds on u_i and σ_i are given by as $u_{i_{min}} \le u_i \le u_{i_{max}}$ and $0 \le \sigma \le \sigma_{max}$. The above mentioned optimization problem of (5.33) renders $2n_r$ number of control inputs (u) for a multi robot system consisting of n_r robots to perform task re-issuance and specialty reassignment constrained by multiple control barrier functions, slack variables and $2n_r$ rectangular velocity components. In the next section, the detailed simulation results are presented.

5.4 Simulation Results and Discussions

In this section, the simulations results for an environment aware resilience of multi robot tasks are presented, control strategy for which already developed in section 5.3.2. The quadratic program of equation (5.29) gives a framework for the environment resilience through the dynamic tasking for a multi robot system under optimal strategy in constrained conditions. A Matlab© routine for the optimization problem of equation (5.29) can be programmed in which the n_r robots can be defined with their initial positions taken within an area of 3×3 square meter i.e., extending from x_{min} to x_{max} along x-axis and y_{min} to y_{max} along y-axis, where values for x_{min}, y_{min}, x_{max} and y_{max} are -1.5, -1.5, 1.5 and 1.5 respectively. The initial positions of the robots are initialized in $\mathbb{R}^2$ using Matlab© command *rand*. For solving the quadratic program of equation (5.33), the values for the variables are as $\tau = 1000$, $\zeta = 1000000$, $\sigma_{max} = 12$, $w_r = 0.04\,m$, $w_o = 0.04\,m$, $\rho = 10$, $k = 10$, $\eta = -8$, $d_o = 0.05\,m$ and $d_c = 0.1\,m$ [16]. By selecting an *ith* robot out of the n_r robots and calculating its positions q_{ix}, q_{iy} alongside positions of the neighborhood robots q_{jx}, q_{jy}, the A and b for the quadratic program of the equation (5.33) can be calculated. The quadratic problem can be solved at each time step to get u_{ix}, u_{iy}, u_{jx} and u_{jy} for each robot in order to move the robots in a plane while calculating the updated positions q_{ix}, q_{iy}, q_{jx} and q_{jy} using the equations (5.34) and (5.35) until desired objective is achieved as of Fig. 5.8. As mentioned in previous chapter, formulae for calculating updated positions q_{ix}, q_{iy} of robot i in x and y directions where t and dt denote as time and time step respectively, are restated below:

$$q_{ix}^{t+dt} = q_{ix}^{t} + u_{ix}^{t} dt, \;\; i \in \{1, ..., n_r\} \tag{5.34}$$

$$q_{iy}^{t+dt} = q_{iy}^{t} + u_{iy}^{t} dt, \;\; i \in \{1, ..., n_r\} \tag{5.35}$$

Here the numerous simulation results showcase the plotting of resultant velocity u_i for each robot where $u_i = \sqrt{u_{ix}^2 + u_{iy}^2}$ and $i \in \{1, ..., n_r\}$. The case I includes resilient tasks execution with 2, 3 and 4 robots whereas the cases II and III are considered with the maximum no. of robots i.e., 3 and 4 robots respectively. The presented simulation results, in terms of the optimization variables' behavior, are found to be in agreement with the existing work of [16].

5.4.1 Case I: Simultaneous Task Re-issuance/Withdrawal and Specialties Activation/Deactivation

For performing the simulations, different scenarios can be considered here such as that the robots in a team have the specialties which equip them to perform all the tasks. However as explained in the section 5.3.2, the task issuance/withdrawal/re-issuance to every robot in a team of the robots and further the specialty reassignment among these robots can be carried out by manipulating the environment coefficients against the specialties of every robot in the team. The environment coefficients given by the equation (5.16) are part of the environment module vector E as given by the equation (5.14). Here for simplicity the proposed idea is solved for two robots whereas for three and four robots the corresponding results are showcased without getting into the details. Multiple tasks are created for the varying number of robots and numerous obstacles in order to prove the efficacy of optimization problem of equation (5.29). The solution for two robots is as mentioned below:

- No. of robots: $n_r = 2$
- No. of tasks: $n_t = 2$
- No. of obstacles: $n_o = 4$

- No. of probable inter-robots collisions: 1
- The environment module E (equation (5.14)) against two robots is given by as follows:

$$E = [\ e_{11} \quad e_{12} \quad e_{21} \quad e_{22}\ ]^T \tag{5.36}$$

- The task re-issuance matrix (equation (5.20)) for two robots is given by:

$$\psi' = \begin{bmatrix} \frac{\psi_1^\theta}{\psi_1^\theta+\psi_2^\theta} & 0 & \frac{\psi_1^\theta}{\psi_1^\theta+\psi_2^\theta} & 0 \\ 0 & \frac{\psi_2^\theta}{\psi_1^\theta+\psi_2^\theta} & 0 & \frac{\psi_2^\theta}{\psi_1^\theta+\psi_2^\theta} \end{bmatrix} E^T \tag{5.37}$$

- The environment specialty matrix (C_i^E from equation (5.17)) against robot 1 and 2 is given by:

$$C_1^E = diag\left((e_1)^T\right) C_1 = \begin{bmatrix} e_{11}c_{11} & 0 \\ 0 & e_{12}c_{12} \end{bmatrix} \tag{5.38}$$

$$C_2^E = diag\left((e_2)^T\right) C_2 = \begin{bmatrix} e_{21}c_{21} & 0 \\ 0 & e_{22}c_{22} \end{bmatrix} \tag{5.39}$$

- The current task specification ψ^* (equation (5.21)) is given by:

$$\psi^* = \frac{1}{2}[C_1^E C_1^{E^+} C_2^E C_2^{E^+}] \begin{bmatrix} \phi_{11} \\ \phi_{12} \\ \phi_{21} \\ \phi_{22} \end{bmatrix} \tag{5.40}$$

where the term $C_i^{E^+} = ((C_i^E)^T C_i^E)^{-1}(C_i^E)^T$.

- Velocities vector:

$$u_{2n_r\times 1} = [u_{1x},\ u_{1y},\ u_{2x},\ u_{2y}]^T$$

- Minimum velocity bound for $2n_r$ rectangular velocity components:

$$u_{min} = -[u_{1xmin},\ u_{1ymin},\ u_{2xmin},\ u_{2ymin}]^T$$

- Maximum velocity bound for $2n_r$ rectangular velocity components:

$$u_{max} = [u_{1xmax},\ u_{1ymax},\ u_{2xmax},\ u_{2ymax}]^T$$

5.4.1.1 Scenario I: Resilient Execution of 2 Tasks with 2 Robots Avoiding 4 Obstacles and 1 Probable Inter-robot Collision.

Two robots r_1 and r_2 are issued with the tasks T_1 and T_2 to reach certain destinations d_1 and d_2 in a plane while avoiding the multiple obstacles and inter-robots collisions on way. Using the aforementioned equations (5.36) and (5.37) the tasks are withdrawn for sometime and the specialties deactivated simultaneously, thereafter tasks are re-issued and the specialties are reactivated, again simultaneously.
Solution. At the start of experiment the robots r_1 and r_2 are assigned the tasks T_1 and T_2. However due to certain unpredictable situation at certain time (say $t = 2$ sec), using environment coefficients vector e_i (equation (5.16)), the tasks are withdrawn from both robots (say for $t = 1$ sec) while also deactivating their specialties causing the ceasing of robots movements. However, for the time interval $2 \leq t < 3$, these tasks are re-issued to robots but this time their tasks are switched that is, the environment module E re-issues tasks and activates different specialties to those of earlier, through each robots environment coefficients e_i to accomplish the alternate objectives that is, T_1 to be accomplished by r_2 and T_2 to be accomplished by r_1. For $t < 2$, the environment module vector from equation (5.36) is given by as follows:

$$E^{t<2} = [\ 1 \quad 0 \quad 0 \quad 1\ ]^T \tag{5.41}$$

where environment coefficients vectors for r_1 and r_2 in the light of equation (5.16) are given as $e_1^{t<2} = [\ 1 \quad 0\ ]^T$ and $e_2^{t<2} = [\ 0 \quad 1\ ]^T$. For the task issuance/withdrawal purpose, in equation (5.37), the value of $\psi_1^\theta = \psi_2^\theta = 1$ that the both tasks are kept issued for the entire simulation time. Thus the task re-issuance matrix of equation (5.37) for the two robots takes the form as follows:

$$\psi'^{t<2} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} E^{t<2} \tag{5.42}$$

Here it is considered that all the robots can perform all the tasks that is, $C_1 = C_2 = \begin{bmatrix} 1 & 0; & 0 & 1 \end{bmatrix}$. Thus in the light of equations (5.38) and (5.39), the environment aware specialty matrix (C_i^E) against r_1 and r_2 for $t < 2$ takes the following form:

$$C_1^{E^{t<2}} = diag\left((e_1^{t<2})^T\right) C_1 \tag{5.43}$$

$$C_2^{E^{t<2}} = diag\left((e_2^{t<2})^T\right) C_2 \tag{5.44}$$

The equation (5.41) manifests that as $e_{11} = e_{22} = 1$, the environment module E can activate issuance of tasks to r_1 and r_2 while also activating their corresponding specialties through each robot environment coefficients vector to perform tasks T_1 and T_2. The robots r_1 and r_2 start their journey and meanwhile, the environment module E (equation (5.41)) is looked up and gets updated by its members e_i against each robot at each time-step dt. Thereafter for time interval $2 \leq t < 3$, the environment module E (equation (5.36)) changes its environment coefficients (e_i) as follows:

$$E^{2\leq t<3} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{5.45}$$

Where environment variable vector for r_1 in the light of equation (5.16) is $e_1^{2\leq t<3} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ and for r_2 is $e_2^{2\leq t<3} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$. The corresponding task re-issuance matrix (equation (5.37)) becomes as follows:

$$\psi'^{2\leq t<3} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} E^{2\leq t<3} \tag{5.46}$$

and the environment aware specialties of equations (5.38) and (5.39) become as given by:

$$C_1^{E^{2\leq t<3}} = C_2^{E^{2\leq t<3}} = \begin{bmatrix} 0 & 0; 0 & 0 \end{bmatrix} \tag{5.47}$$

The above equation depicts the withdrawal of tasks from robots as well as the deactivation of specialties causing ceasing of advancements by robots r_1 and r_2. From time 3 to 10 secs the environment module E (equation (5.48)) energizes different environment coefficients against the robots specialties to those of earlier environment coefficients that is, the environment module re-issues tasks T_1 and T_2 while activating corresponding environment aware specialties to execute the tasks T_1 by r_2 and T_2 by r_1.

$$E^{3\leq t\leq 10} = \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}^T \tag{5.48}$$

Where the environment variable vector for r_1 in the light of equation (5.16) is $e_1^{3\leq t\leq 10} = [\begin{array}{cc} 0 & 1 \end{array}]^T$ and for r_2, $e_2^{3\leq t\leq 10} = [\begin{array}{cc} 1 & 0 \end{array}]^T$. For the task re-issuance purpose, as mentioned above the value of $\psi_1^\theta = \psi_2^\theta = 1$. The task re-issuance matrix can be calculated as follows:

$$\psi^{'3\leq t\leq 10} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} E^{3\leq t\leq 10} \tag{5.49}$$

The environment aware specialty matrix (C_i^E) against r_1 and r_2 are recalculated for time interval $3 \leq t \leq 10$ as follows:

$$C_1^{E^{3\leq t\leq 10}} = diag\left((e_1^{3\leq t\leq 10})^T\right) C_1 \tag{5.50}$$

$$C_2^{E^{3\leq t\leq 10}} = diag\left((e_2^{3\leq t\leq 10})^T\right) C_2 \tag{5.51}$$

Similarly the current task vectors $\psi^{*^{t<2}}$, $\psi^{*^{2\leq t<3}}$ and $\psi^{*^{3\leq t\leq 10}}$ are too solved for each time interval by following equation (5.40) through a Matlab© routine. The terms E, ψ' and ψ^* are evaluated and optimization problem of equation (5.29) is solved through Matlab© *quadprog* and the corresponding simulation results are showcased in the Fig. 5.9. It is clear from the results that robots accomplished the issued tasks as per scenario described in above paragraph. In the Fig. 5.9, at $t = 0$ sec, the environment module (equation (5.41)) issues tasks (equation (5.42)) T_1 and T_2 to the robots r_1 and r_2 to reach destinations d_1 and d_2 respectively with the appropriate specialties activated (equations (5.43) and (5.44)) by setting $e_{11} = e_{22} = 1$ in the equation (5.41). As soon as robots start moving to perform the issued tasks, the corresponding optimization variables $\phi_{11}^{t<2}$ and $\phi_{22}^{t<2}$ as per equation (5.29f) start increasing whereas corresponding slack variables $\sigma_{11}^{t<2}$ and $\sigma_{22}^{t<2}$ as per equation (5.29e) start lessening. The velocities u_1 and u_2 constrained by (5.29g) also start increasing as shown in the Fig. 5.9. At $t = 2$ sec, the environment module (equation (5.45)) withdraws these tasks (equation (5.46)) from robots and specialties are deactivated (5.47) for 1 sec meanwhile $\phi_{11}^{2\leq t<3}$, $\phi_{22}^{2\leq t<3}$, $\sigma_{11}^{2\leq t<3}$ and $\sigma_{22}^{2\leq t<3}$ start maintaining the constant values, while the velocities u_1 and u_2 become zero. At $t = 3$ sec the environment module (equation (5.48)) re-issues tasks (equation (5.49)) with switching i.e., $e_{12} = e_{21} = 1$, activating the appropriate specialties (equations (5.50) and (5.51)) forcing r_1 to perform T_2 by reaching d_2

and r_2 to perform T_1 by reaching d_1. Finally $\phi_{12}^{3\leq t\leq 10}$ and $\phi_{21}^{3\leq t\leq 10}$ reach value of 1 indicating accomplishment of T_2 and T_1 whereas $\sigma_{12}^{3\leq t\leq 10}$ and $\sigma_{21}^{3\leq t\leq 10}$ reach nearly zero value indicating that T_2 and T_1 are performed with highest preferences by robots r_1 and r_2 respectively. The corresponding velocities of robots u_1 and u_2 can be drawn as they start increasing until reaching zero for time interval $2 \leq t < 3$ secs and start increasing again, finally become zero as soon as robots reach the intended destinations. Here for the simulation purpose, the following values against u_{min} and u_{max} are taken:

$$u_{min} = -[0.1,\ 0.1,\ 0.1,\ 0.1] \tag{5.52}$$

$$u_{max} = [0.1,\ 0.1,\ 0.1,\ 0.1] \tag{5.53}$$

The Figs. 5.10(a) to 5.10(c) are the snapshots taken during the simulation ex-

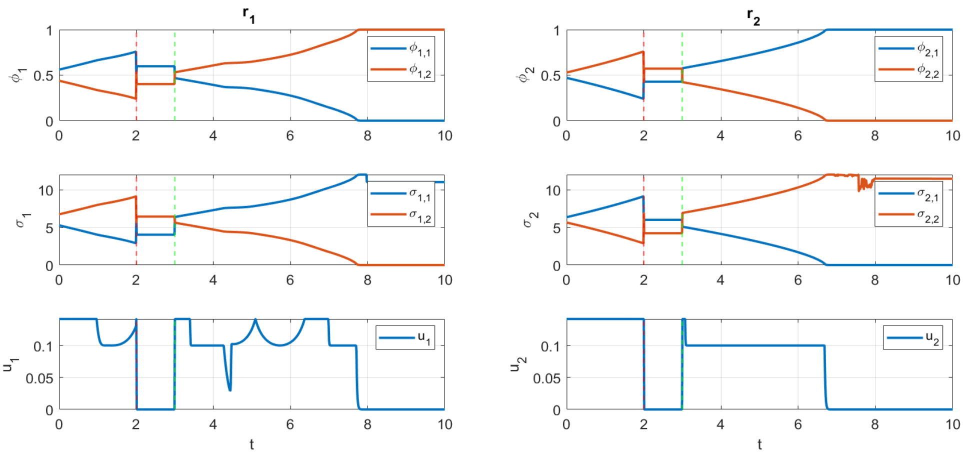
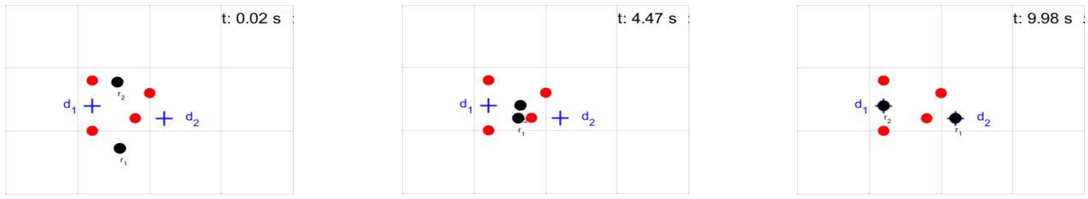


FIGURE 5.9: Behavior of optimization variables w.r.t time. At $t = 0.0$, environment module (equation (5.41)) issues (equation (5.42)) tasks T_1 and T_2 to robots r_1 and r_2 to reach destinations d_1 and d_2 with environment-aware specialties activated (equations (5.43) and (5.44)). Optimization variables ϕ_{11}, ϕ_{22} start increasing and slack variables σ_{11} and σ_{22} start lessening. At $t = 2$ sec (red dashed line), environment module (equation (5.45)) withdraws tasks (equation (5.46)) from robots and specialties gets deactivated (equation (5.47)) for 1 sec whereas ϕ_{11}, ϕ_{22}, σ_{11} and σ_{22} also attain the constant values. At $t = 3$ sec (green dashed line) environment module (equation (5.48)) re-issues tasks (equation (5.49)) with switching also activating appropriate robots specialties through environment coefficients (equations (5.50) and (5.51)), finally r_1 performs T_2 reaching d_2 and r_2 performs T_1 reaching d_1. ϕ_{12} and ϕ_{21} reach value of 1 indicating accomplishment of T_2 and T_1 whereas σ_{12} and σ_{21} reach nearly zero value indicating T_2 and T_1 performed with top preferences by r_1 and r_2. Velocities u_1 and u_2 start increasing, finally becoming zero while robots avoid collisions causing abrupt changes in velocities as indicated.

periment. In the Fig. 5.10(a) the robots r_1 and r_2 start the tasks while avoiding



(a) Robots r_1 and r_2 at the start of simulation. Red circles denote obstacles. + signs depict destinations d_1 and d_2 in space.

(b) r_1 and r_2 carryout collisions and obstacles avoidance.

(c) r_1 and r_2 reach destinations d_2 and d_1 indicating accomplishment of tasks T_2 and T_1 respectively.

FIGURE 5.10: Simultaneous task re-issuance and environment-aware specialties activation/deactivation (scenario I).

obstacles and collisions in the Fig. 5.10(b), finally accomplishing the tasks as shown in the Fig. 5.10(c). A task is said to be accomplished if the optimization variable ϕ for the each robot of the team attains the value of 1 and the variable σ becomes nearly of zero value.

5.4.1.2 Scenario II: Resilient Execution of 3 Tasks with 3 Robots Avoiding 6 Obstacles and 6 Probable Inter-robots Collisions.

Three robots r_1, r_2, r_3 are issued with the tasks T_1, T_2, T_3 to reach certain destinations d_1, d_2, d_3 respectively in a plane while avoiding the multiple obstacles and inter-robots collisions on way as shown in the Fig. 5.12(a) however due to certain situation, the tasks are withdrawn from all the robots for sometime while also deactivating their specialties through each robots environment coefficients vector e_i, thus stopping the robots in mid-course. However, thereafter tasks are re-issued to robots but this time their tasks are swapped among robots causing environment module E to re-issue tasks T_1, T_2, T_3 to r_3, r_1, r_2 respectively and simultaneously deactivating different specialties to those of earlier, to accomplish the swapped tasks. All tasks are said to be accomplished if the optimization variable ϕ for each robot of the team attains the value of nearly 1 and σ becomes nearly of zero value for all the participating robots accomplishing the assigned tasks. Fig. 5.11 showcase the corresponding results.

Solution. For above mentioned scenario, the environment module E is given by:

$$E^{t<0.2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}^T \tag{5.54}$$

Supposedly at some point in time during the mission course, E takes the form causing deactivation of robots specialties and withdrawal of tasks for sometime as given by:

$$E^{0.2\leq t<0.3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{5.55}$$

After a certain time, E takes the following form causing tasks re-issuance and environment-aware specialty reassignment:

$$E^{0.3\leq t<4.0} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}^T \tag{5.56}$$

We can calculate environment aware specialties $C_i^{E^{t<0.2}}$, $C_i^{E^{0.2\leq t<0.3}}$, $C_i^{E^{0.3\leq t<4.0}}$, (equation (5.17)) task re-issuance matrix $\psi'^{t<0.2}$, $\psi'^{0.2\leq t<0.3}$, $\psi'^{0.3\leq t<4.}$ (equation (5.20)) and current tasks vectors $\psi^{*t<0.2}$, $\psi^{*0.2\leq t<0.3}$, $\psi^{*0.3\leq t<4.0}$ (equation (5.21)) accordingly, in order to solve the optimization problem of (5.29) at each time step dt. Term $\psi_n^\theta = 1$ (equation (5.20)) which manifests that task issuance is active for the entire simulation time. Here for the simulation purpose, the following values against u_{min} and u_{max} are taken as given by:

$$u_{min} = -[1.5,\ 1.5, \ldots, 1.5]_{1\times 6} \tag{5.57}$$

$$u_{max} = [1.5,\ 1.5, \ldots, 1.5]_{1\times 6} \tag{5.58}$$

The Figs. 5.12(a) to 5.12(c) are the snapshots taken during the simulation experiment. In Fig. 5.12(a) the robots r_1, r_2 and r_3 start the tasks while avoiding obstacles and collisions (Fig. 5.12(b)), finally accomplishing the tasks as shown in Fig. 5.12(c).

5.4.1.3 Scenario III: Resilient Execution of 4 Tasks with 4 Robots Avoiding 10 Obstacles and 12 Probable Inter-robots Collisions.

Four robots r_1, r_2, r_3, r_4 are issued with the tasks T_1, T_2, T_3, T_4 to reach certain destinations d_1, d_2, d_3, d_4 respectively on a plane while avoiding multiple obstacles and inter-robots collisions on way as shown in Fig. 5.14(a). However thereafter the tasks are withdrawn for sometime and then re-issued to robots but this time

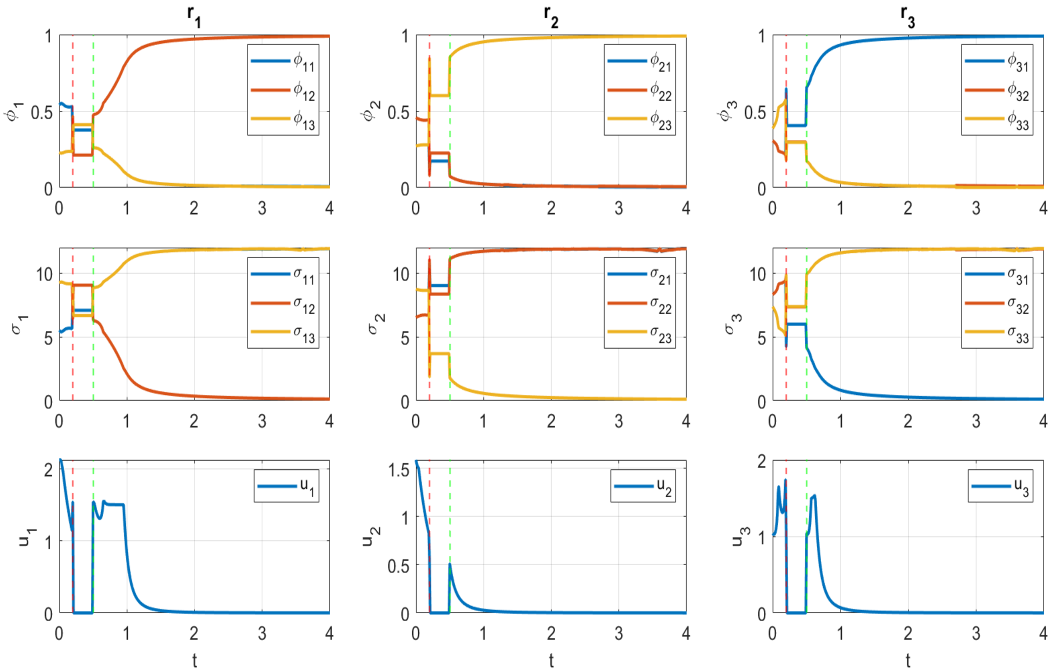
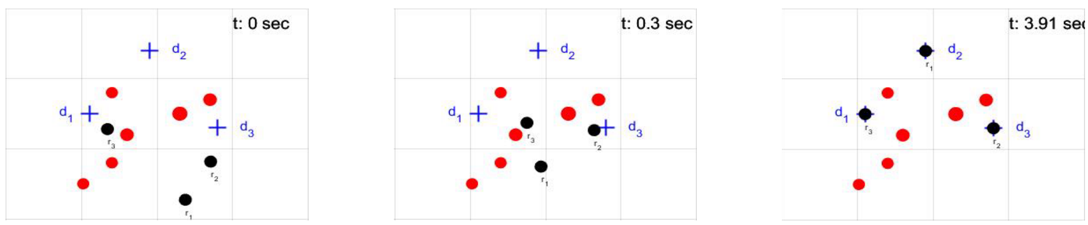


FIGURE 5.11: Behavior of optimization variables w.r.t time. At $t = 0.0$ sec, environment module (equation (5.54)) issues tasks T_1, T_2 and T_3 to robots r_1, r_2 and r_3 to reach destinations d_1, d_2 and d_3 respectively with appropriate specialties activated through environment coefficient vector. Terms ϕ_{11}, ϕ_{22} and ϕ_{33} start increasing and slack variables σ_{11}, σ_{22} and σ_{33} starts lessening. At $t = 0.2$ (red dashed line) environment module (equation (5.55)) withdraws tasks from robots and environment aware specialties gets deactivated for 0.1 whereas ϕ_{11}, ϕ_{22}, ϕ_{33}, σ_{11}, σ_{22} and σ_{33} also attain the constant values. At $t = 0.3$ (green dashed line) environment module (equation (5.56)) re-issues tasks with switching, also activating appropriate robots specialties through environment coefficients vectors, finally r_1 performs T_2 reaching d_2, r_3 performs T_1 reaching d_1 and r_2 performs T_3 reaching d_3. Associated optimization variables ϕ_{12}, ϕ_{23} and ϕ_{31} reach value of 1 indicating accomplishment of T_2, T_3 and T_1 whereas σ_{12}, σ_{23} and σ_{31} reach nearly zero value indicating T_2, T_3 and T_1 are performed with top preferences by robots r_1, r_2 and r_3. Velocities u_1, u_2 and u_3 start increasing, finally becoming zero while robots avoid collisions and obstacles causing abrupt changes in velocities as indicated.



(a) r_1, r_2 and r_3 at the start of simulation. Red circles denote obstacles.

(b) Collision and obstacles avoidance

(c) r_1, r_2 and r_3 accomplish tasks T_2, T_3 and T_1 by reaching destinations d_2, d_3 and d_1.

FIGURE 5.12: Simultaneous task re-issuance and environment-aware specialties activation/deactivation (scenario II).

their tasks are swapped among robots causing the environment module E to re-issue tasks T_1, T_2, T_3, T_4 to r_4, r_3, r_2, r_1 respectively. All tasks are said to be accomplished if the optimization variables ϕ for each robot of the team attain value of 1 and variables σ become nearly of zero value for all the participating robots (Fig. 5.13).

Solution. For above mentioned scenario, the environment module E (equation (5.14)) for simulation time $t < 0.2$ is given by:

$$E^{t<0.2} = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]^T \tag{5.59}$$

For simulation time windows i.e., 0.2 to 0.3 and 0.3 to 4.0 the environment module E (equation (5.14)) takes the following forms respectively:

$$E^{0.2\leq t<0.3} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \tag{5.60}$$

$$E^{0.3\leq t<4.0} = [0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]^T \tag{5.61}$$

The environment aware specialties $C_i^{E^{t<0.2}}$, $C_i^{E^{0.2\leq t<0.3}}$, $C_i^{E^{0.3\leq t<4.0}}$ (equation (5.17)) task re-issuance vectors $\psi'^{t<0.2}$, $\psi'^{0.2\leq t<0.3}$, $\psi'^{0.3\leq t<4.0}$ (equation (5.20) with $\psi_n^\theta = 1$) and current tasks vectors $\psi^{*t<0.2}$, $\psi^{*0.2\leq t<0.3}$, $\psi^{*0.3\leq t<4.0}$ (equation (5.21)) can be calculated accordingly, in order to solve the optimization problem of (5.29) at each time step dt. Here for the simulation purpose, the following values against u_{min} and u_{max} are considered:

$$u_{min} = -[1.0,\ 1.0, \ldots, 1.0]_{1\times 2n_r} \tag{5.62}$$

$$u_{max} = [1.0,\ 1.0, \ldots, 1.0]_{1\times 2n_r} \tag{5.63}$$

The Figs. 5.14(a) to 5.14(c) are the snapshots taken during the simulation experiment. In Fig. 5.14(a) r_1, r_2, r_3 and r_4 start the tasks and after nearly avoiding collisions (Fig. 5.14(b)), finally accomplishing the tasks as shown in Fig. 5.14(c).

5.4.2 Case II: Midcourse Tasks Withdrawal with Active Environment Coefficients

Scenario IV: Non Resilience of 3 Tasks with 3 Robots Avoiding 6 Obstacles and 6 Probable Inter-robots Collisions. Three robots r_1, r_2, r_3

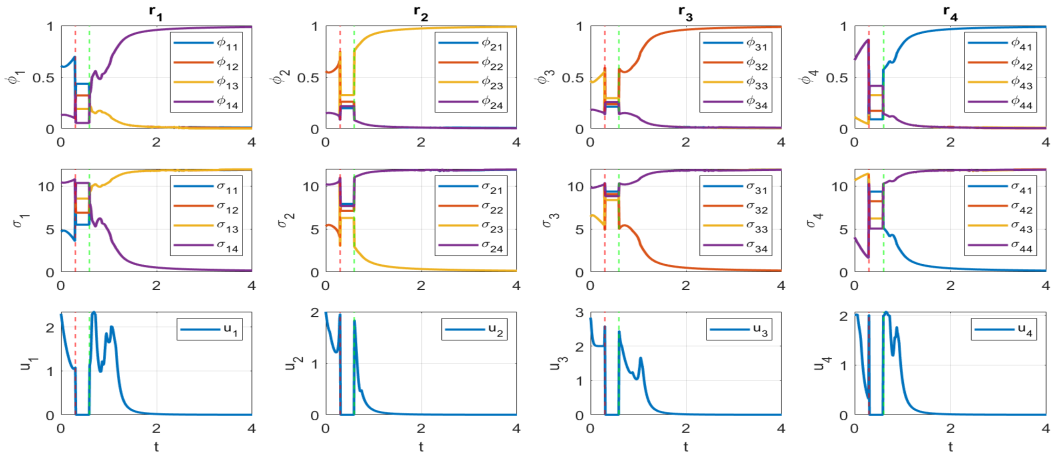
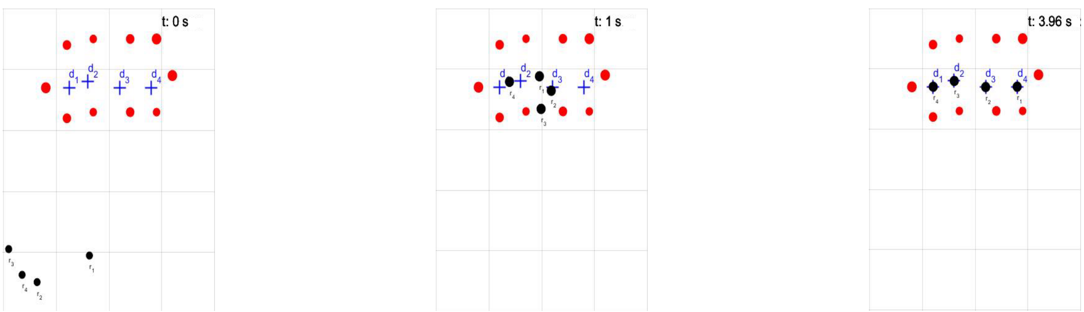


FIGURE 5.13: Behavior of optimization variables w.r.t time. At $t = 0.0$ sec, environment module (equation (5.59)) issues tasks T_1, T_2, T_3, T_4 to robots r_1, r_2, r_3, r_4 to reach destinations d_1, d_2, d_3, d_4 respectively with appropriate specialties (equation (5.17)) activated through each robots environment coefficient vector e_i (equation (5.16)). Variables ϕ_{11}, ϕ_{22}, ϕ_{33}, ϕ_{44} start increasing and slack variables σ_{11}, σ_{22}, σ_{33}, σ_{44} start lessening. At $t = 0.02$ sec (red dashed line), environment module (equation (5.60)) withdraws tasks from robots and environment-aware specialties gets deactivated for 0.03 sec whereas ϕ_{11}, ϕ_{22}, ϕ_{33}, ϕ_{44}, σ_{11}, σ_{22}, σ_{33}, σ_{44} also attain the constant values. At $t = 0.06$ sec (green dashed line) environment module (equation (5.61)) re-issues tasks with switching, also activating appropriate robots specialties through environment coefficients vectors, finally r_1 performs T_4 by reaching d_4, r_3 performs T_2 by reaching d_2, r_2 performs T_3 by reaching d_3 and r_4 performs T_1 by reaching d_1. Associated optimization variables ϕ_{14}, ϕ_{23},ϕ_{32}, ϕ_{41} reach value of 1 indicating accomplishment of T_4, T_3, T_2 and T_1 whereas σ_{14}, σ_{23}, σ_{32} and σ_{41} reach nearly zero value indicating T_4, T_3, T_2 and T_1 are performed with top preferences by robots r_1, r_2, r_3 and r_4. Velocities u_1, u_2, u_3 and u_4 start increasing, becoming zero in the interval and then increasing again, while robots avoid collisions and obstacles causing abrupt changes in velocities as indicated, all velocities finally becoming zero.



(a) Robots r_1, r_2, r_3 and r_4 at the start of simulation. Red circles denote obstacles.

(b) Robots are near their destinations after probable collisions and obstacles avoidance.

(c) Tasks T_1, T_2, T_3 and T_4 accomplishment by r_4, r_3, r_2 and r_1 by reaching destinations d_1, d_2, d_3 and d_4.

FIGURE 5.14: Simultaneous task re-issuance and specialties activation/deactivation (scenario III).

are issued with tasks T_1, T_2, T_3 to reach certain destinations d_1, d_2, d_3 respectively in a plane while avoiding multiple obstacles and inter-robots collisions on way as shown in Fig. 5.16(a). However due to certain situation, the tasks are withdrawn from all the robots however their specialties are kept activated through each robots environment coefficient vectors e_i, thus stopping the robots on way.

Solution. For above mentioned scenario, the environment module E for simulation time $t < 4.0$ is given by:

$$E^{0<t<4.0} = [\ 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1\ ]^T \tag{5.64}$$

The term ψ_n^θ in re-issuance vector (equation (5.20)) takes the following values for intervals $t \leq 0.2$ and $0.2 \leq t < 4.0$:

$$\psi_1^{\theta^{t<0.2}} = \psi_2^{\theta^{t<0.2}} = \psi_3^{\theta^{t<0.2}} = 1 \tag{5.65}$$

For withdrawal of tasks:

$$\psi_1^{\theta^{0.2\leq t<4.0}} = \psi_2^{\theta^{0.2\leq t<4.0}} = \psi_3^{\theta^{0.2\leq t<4.0}} = 0 \tag{5.66}$$

We can calculate environment aware specialties $C_i^{E^{0<t<4.0}}$ and current tasks vectors $\psi^{*^{t<0.2}}$, $\psi^{*^{0.2\leq t<0.3}}$, $\psi^{*^{0.3\leq t<4.0}}$ from equations (5.17) and (5.21) respectively, in order to solve the optimization problem of (5.29). Fig. 5.15 shows the simulation result the scenario as mentioned in the preceding paragraph. Fig. 5.16(a) and Fig. 5.16(b) are the snapshots taken during the simulation experiment. In Fig. 5.16(a) r_1, r_2 and r_3 start the tasks but couldn't complete due to withdrawal of tasks ($\psi_1^\theta = \psi_2^\theta = \psi_3^\theta = 0$) in midcourse as shown in Fig. 5.16(b).

5.4.3 Case III: Midcourse Degradation of Environment Coefficients with Active Task Issuance

Scenario V: Resilient Execution of 2 Mandatory Tasks with 4 Robots Avoiding 10 Obstacles and 12 Probable Inter-robot Collisions. Four robots r_1, r_2, r_3, r_4 are issued with tasks T_1, T_2, T_3, T_4 to reach d_1, d_2, d_3, d_4 respectively with active tasks issuance strategy. The tasks T_1 and T_2 are mandatory to be executed but T_3 and T_4 are not mandatory. At a certain time, the

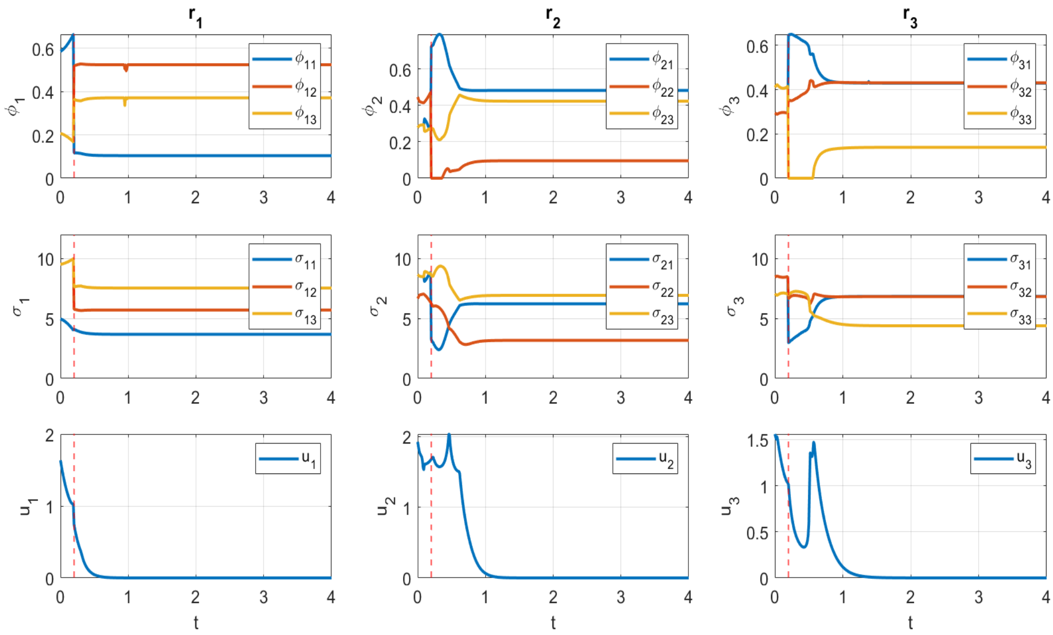


FIGURE 5.15: Behavior of optimization variables w.r.t time. At $t = 0.0$ sec, environment module (equation (5.64)) issues tasks T_1, T_2, T_3 to robots r_1, r_2, r_3 with active tasks issuance strategy (equation (5.65)) to reach destinations d_1, d_2, d_3 respectively with appropriate specialties activated through each robots environment coefficient vector e_i. Variables ϕ_{11}, ϕ_{22}, ϕ_{33} start increasing and slack variables σ_{11}, σ_{22}, σ_{33} start lessening. At $t = 0.02$ sec (red dashed line), tasks are withdrawn (equation (5.66)) from robots and environment-aware specialties (equation (5.55)) are kept activated. Variables ϕ_{11}, ϕ_{22}, ϕ_{33}, σ_{11}, σ_{22}, σ_{33} to attain the constant values. Finally, no robot could be able to reach the destinations till the end of the experiment, also making robots velocities reach zero value.

(a) Robots r_1, r_2, r_3 and r_4 at the start are issued with tasks ($\psi_1^\theta = \psi_2^\theta = \psi_3^\theta = 1$) to reach destinations d_1, d_2, and d_3 in space.

(b) Tasks are withdrawn ($\psi_1^\theta = \psi_2^\theta = \psi_3^\theta = 0$) causing robots to stop though e_i against each robot are kept activated for assigned tasks performance.

FIGURE 5.16: Case II: Midcourse withdrawal of tasks with active environment coefficients (scenario IV).

environment coefficients e_{11} and e_{22} against the robots r_1 and r_2 gets depreciated (say more than 30%) due to certain reasons i.e., sharp degradation in required specialty (equation (5.18)) as indicated by the corresponding coefficient vector (equation (5.16)) thus stopping the both robots on the way. The environment module E updates the corresponding robot status as inactive robot as per rules mentioned in equations (5.18) and (5.19). Since T_1 and T_2 are mandatory to be executed, the situation prompts E to invoke r_3 to perform T_2 and r_4 to perform T_1. For this special scenario, E is not accounted for in task re-issuance vector of equation (5.20). The modified task re-issuance vector for the current scenario is given by as:

$$\psi(t)' = \frac{1}{\psi_1^{\theta}(t) + \psi_2^{\theta}(t) + ... + \psi_{n_t}^{\theta}(t)}(diag((\psi^{\theta}(t))^T), ..., diag((\psi^{\theta}(t)^T))_{1\times 2n_t} \quad (5.67)$$

The environment module E for $t < 0.3$ is given by:

$$E^{t<0.3} = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]^T \quad (5.68)$$

For simulation time window from 0.3 to 4.0 the environment module E takes the following form as given below:

$$E^{0.2\leq t<4.0} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]^T \quad (5.69)$$

For modified task re-issuance vector of equation (5.67), the $\psi_n^{\theta} = 1$ for the entire simulation time. The Figs. 5.18(a) and 5.18(b) are the snapshots taken during the simulation experiment. In Fig. 5.17 the robots r_1, r_2, r_3, r_4 start performing the tasks and the Fig. 5.18(b) shows the accomplishment of the mandatory tasks by two robots r_3 and r_4 only.

5.4.4 Miscellaneous Observations

a. Convergence Time: The Fig. 5.19 shows time plot against number of robots (n_r) with respective constraints considered in scenarios I, II and III. As the number of robots increases as part of team, the time to accomplish tasks also increases. The blue, green and orange regions show time windows for scenarios I, II and III as presented in previous sections.

b. Computational Complexity: The computational complexity $O(n_r n_t)$ calculated for the optimization problem of (5.29) with $n_r = 4$, $n_t = 4$ and $n_o = 10$ is 15.253×10^6 operations [16, 92]. With the processor speed of 2.9 Ghz and

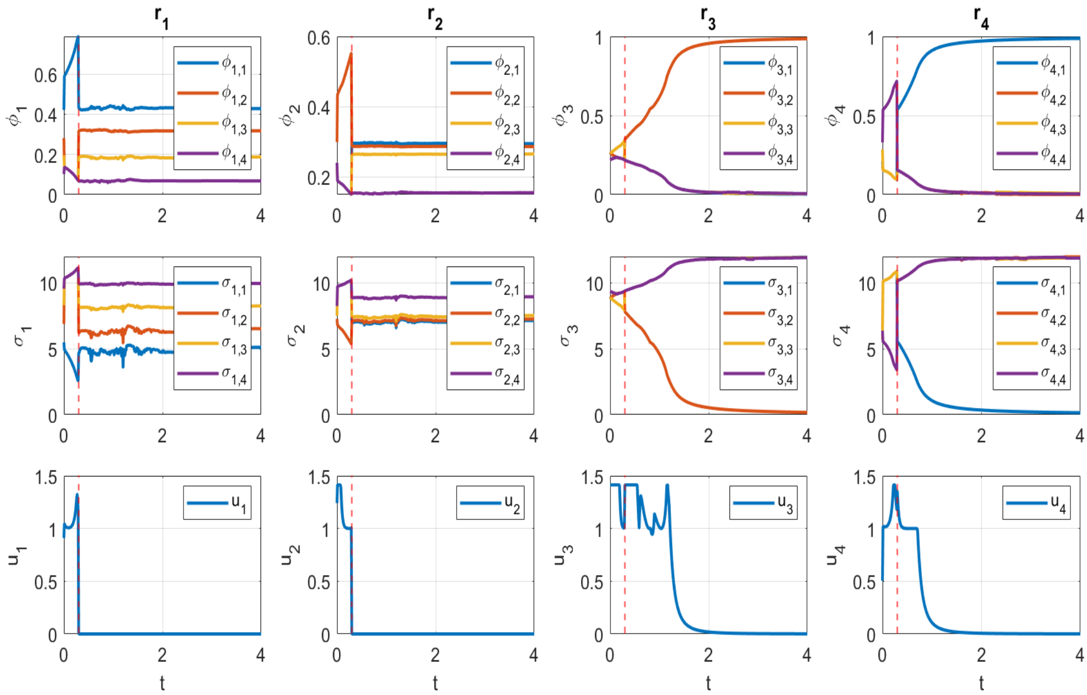


FIGURE 5.17: Behavior of optimization variables w.r.t time. At $t = 0.0$ sec, environment module (equation (5.68)) issues tasks T_1, T_2, T_3, T_4 to robots r_1, r_2, r_3, r_4 to reach destinations d_1, d_2, d_3, d_4 respectively with appropriate specialties activated through each robots environment coefficient vector e_i. Variables ϕ_{11}, ϕ_{22}, ϕ_{33}, ϕ_{44} starts increasing and slack variables σ_{11}, σ_{22}, σ_{33}, σ_{44} start lessening. At $t = 0.3$ sec (red dashed line), due to degradation of corresponding e_i, environment module E (equation (5.69)) deactivates r_1 and r_2, activating environment aware specialties for only r_3 and r_4 to perform T_2 and T_1 respectively that is, ϕ_{32} and ϕ_{41}, reach value of 1. Finally only two robots could be able to reach the destinations till the end of the experiment, also making velocities of r_1 and r_2 to reach zero value.

(a) r_1, r_2, r_3 and r_4 at the start are issued with tasks to reach destinations d_1, d_2, d_3 and d_4.

(b) r_3 and r_4 performed mandatory tasks T_2 and T_1 respectively but degradation of corresponding environment coefficients of r_1 and r_2 caused E to stop the robots.

FIGURE 5.18: Case III: Midcourse degradation of environment coefficients with active task issuance (scenario V).

computations of 15.253×10^6, the minimum distance maintained during collision avoidance can be calculated using standard formulae that is, $distance = robot's\ velocity \times computational\ time = 0.525\ mm$ which is less than minimum threshold distance of $0.1\ m$ for each robot's velocity of $0.1\ m/s$.

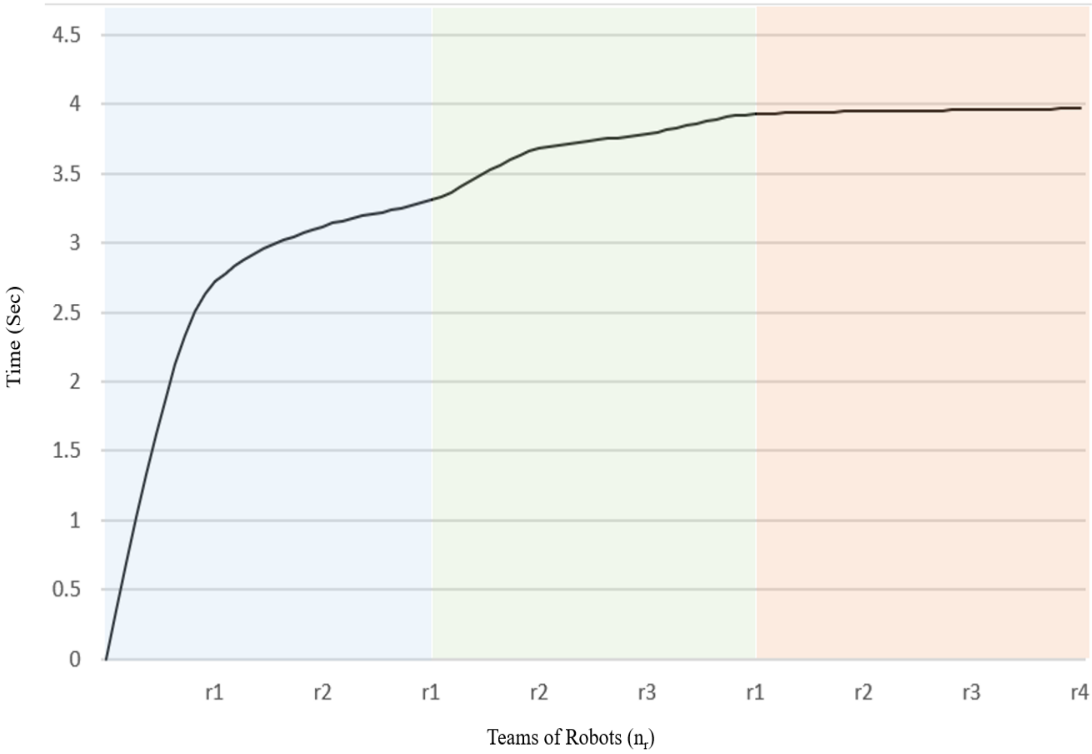


FIGURE 5.19: Task accomplishment time with number of robots (n_r). Figure shows time plot against (n_r) with respective constraints. As the number of robots increases as part of team, the time to accomplish tasks also increases. Blue, green and orange regions show time-windows for scenarios I, II and III.

5.4.5 Limitations of the Research

Few limitations of the thesis research work are discussed in this section as mentioned below.

a. Communication: In this research, we have assumed that communication between robots in swarm is ideal while no delays and packet losses are there. This may outcome a worst consequence as in real world situations such as disaster zones, communication may be unreliable and consequently robots might compute incorrect priorities due to outdated information. To mitigate such a scenario, the integration of robust estimation techniques are required to be incorporated.

b. Scalability: The scalability for large robot teams may pose a challenge for real-time Quadratic Programming (QP) due to increased computational demands. These higher computational requirements can lead to memory or CPU bottlenecks, causing standard solvers (such as Matlab©'s *quadprog*) to fail. In such scenarios, task clustering can be employed, where similar tasks are grouped or the memory and CPU capacity can be upgraded accordingly.

5.5 Chapter Summary

In this chapter the system modeling and control design technique for the conduct of thesis research are discussed. The relationships between robots and robot tasks are identified. For solving the problem statement of this thesis, it is found that the environment and specialty relationship can lead us to the development of an environment-aware resilient task framework. The notion of allocating tasks to the robots keeping in view the evolving environment and the required specialties to perform the allocated tasks is considered. The control design for an environment-aware resilience of multi robot tasks is developed through the scenario of dynamic tasking for a multi robot system under optimal strategy within constrained conditions and further simulated in Matlab©. In the simulation results section, an environment aware resilient framework is simulated for a multi robot system to carryout the dynamic task re-issuance and environment-aware specialty reassignment in an optimal strategy under the constrained environment including obstacles avoidance, inter-robots collisions prevention and bounded velocities. A simultaneous mechanism for the tasks issuance/withdrawal/re-issuance coupled with the robot specialties is developed and simulated as well. Multiple scenarios encompassing three different cases that is, the simultaneous task re-issuance and environment-aware specialties activation/ deactivation, mid-course withdrawal of the tasks with active environment coefficients and mid-course degradation of environment coefficients with active task issuance, are devised and simulated. In order to achieve these outcomes, a novel dynamic task-specialty matrix based mechanism termed as environment module is proposed and further incorporated into an optimization technique. The environment module in optimization program let the robots issue/re-issue, active/inactive the specialties in an online way leading to accomplishment of tasks optimally in an evolving environment. In addition to minimization between task re-issuance and current task preference, the proposed optimization technique-based problem also minimized robots constrained velocities and task related slack variables. This optimization program is subject to the control barrier function based constraint encoding the desired objective cost function along with other constraints primarily required for optimal task execution. The capabilities of robot-to-robot collision avoidance and robot-to-obstacle avoidance are also incorporated through the separate control barrier functions. The presented simulation results show the efficacy of optimization technique based decentralized controllers for solving the problem statement of this thesis by accomplishing the respective tasks thus testifying our established notion. Moreover, these results, in terms of the optimization variables' behavior, are found to be in agreement with the existing work.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this research a methodical approach has been adopted to achieve the desired objective of an environment-aware resilience of the multi robot tasks by a team of robots under the dynamically evolving environment. The detailed literature survey manifested that the control barrier function has an extensive strength and can be leveraged to execute the different multi robot tasks. In order to prove this, a control barrier function based control technique was developed for the ground vehicles to achieve an adaptive cruise control on the different surface conditions under the weather effect. The firm base for achieving an environment aware resilience was constructed through the development of decentralized controllers for the multi robot formation control using control barrier functions. The term “firm base” refers to successful utilization of control barrier functions for task accomplishment in terms of attainment of formation control - a concept further employed for the environment aware resilience of tasks. These efforts led to attain an environment aware resilience of the multi robot tasks through solving the dynamic tasking problem as presented in this thesis. The thesis work first presented the practical concept of the control barrier function as explored from a real life example of the adaptive cruise control. By leveraging on the control barrier function, a control technique was devised for the ground vehicles to achieve an adaptive cruise control on the weather affected surface conditions. The control technique utilized a quadratic program and unified the control barrier function and control Lyapunov function with the surface condition taken as the hard constraint defined through the control barrier function. The simulated results found through this technique exhibited the intended safety and stability objectives on different weather based surface conditions. The presence of the control barrier function ensured the forward invariance of a set defined through the hard constraint whereas the control Lyapunov function helped to reach a desired velocity state. The stability and safety objectives acted as two separate constraints of the designed quadratic program. Through the manifestation of the presented Matlab© based simulation results, the proposed control technique was found correct as the adaptive cruise control was successfully achieved under different surface conditions.

The results as an outcome of adopted approach showed the sudden drop in value of u' to avoid collision as v_{ego} reached the desired velocity of 30 m/s at 6.7 *secs* with corresponding invocation of hard and soft constraints (CBF and CLF), finally maintaining constant values of $v_{ego} = 20\ m/s$ and $z = 36\ m$. The simulation results for other surfaces wet, slush, soft and compact snow had also been provided and compared. The v_{ego} tries to reach 30 m/s at 3.2 *secs* for compact snow, 4 *secs* for soft snow, 5.3 *secs* for slush, 5.8 *secs* for wet and 6.7 *secs* for dry surface with earlier decrease in control inputs, increase in slack variables and invoking of CBF, CLF until the distance z reached a safe value. These simulation results also demonstrated the simultaneous achievement of the control objective and the safety objective. This manifested that the control barrier function has the potential to achieve a control design which paved the way for achieving the multi robot tasks accomplishment through a formation control problem.

The notion of control barrier function based control design was extended and the decentralized controllers based on the control barrier functions were formulated to control the robots in a multi robot system in order to achieve the accomplishment of the assigned tasks. The control barrier functions were employed for attaining the desired objective of the formation forming. These control barrier functions were also utilized for the inter-robots collision avoidance while the desired formation takes place. During the attainment of desired multi robot formation, the constraints were enforced on the rectangular components of velocities and thus the desired formation was achieved with and without the constraints on the robots velocities. The cost function for achieving the desired formation was embedded as the control barrier function based constraint of an optimization problem. The inter-robots collision avoidance among the robots was attained through a separately designed control barrier function as constraint of the same optimization problem. The formation control of different robots and their inter-robots collision avoidance in a team of multiple robots was then carried out by solving the formulated optimization problem based on the control barrier functions. The developed quadratic program based formation controller for each robot was simulated in Matlab© and the robots in the team successfully attained the desired formation in a leaderless consensus by deploying themselves in a plane under the different values of the constrained rectangular velocities without colliding with each other. Several simulation examples with different scenarios were presented to illustrate and compare the working of the constrained controllers providing testimony to the working of our proposed notion. The techniques proposed in this research can be employed when a multi robot system was required to make the desired formation under the confined conditions.

The environment-aware resilience of multi tasks was a special case of multi robot task allocation as explored in this thesis. This led to investigate the tasks allocation and their accomplishments by a multi robot system. The slack variables were used for the optimal allocation of these tasks. These slack variables were incorporated in the control barrier functions based constraints designed for achieving the desired objective costs encoding the assigned tasks. In order to achieve the desired resilience, the proposed environment module was made part of the task re-issuance strategy and environment-aware specialties of the robots. The environment module has environment coefficients vectors associated with the specialties of the robots. The association of the task re-issuance strategy with the environment coefficients ensured the robots participation simultaneously changing with the active/in-active status of robots specialties. The control barrier function based constraint ensured the accomplishments of the assigned tasks through the manipulation of the associated slack variables in an optimal way. The optimality was achieved during the dynamic assignment and prioritization of multiple tasks across a team of heterogeneous robots in a way that optimized the key objectives that is, (1) the developed framework minimized energy consumption as robots optimized their control inputs (u) within selected constrained velocities, (2) the task relaxation constraints (σ) enabled task prioritization by nearly getting value of *zero* with corresponding value of 1 for the vector of task preferences (ϕ). The environment-aware resilient framework was incorporated with the inter-robots collision avoidance and robot-obstacle avoidance through the separate control barrier functions based constraints. The optimization problem overall minimized the robots input velocities and associated slack variables. The manifestation of the resilient execution of multiple tasks was successfully carried out through the three different cases such as the simultaneous task re-issuance and specialties activation/deactivation, the mid-course withdrawal of the tasks with the active environment coefficients and the mid-course degradation of the environment coefficients with the active task issuance. The presented results, in terms of the optimization variables' behavior, were found to be in agreement with the existing work. The overall research thus manifested, the successful achievement of an environment-aware resilience through the development of a framework ensuring dynamic tasking for a multi robot system under optimal strategy in constrained conditions.

6.2 Future Directions

The future research directions in multi robot systems are expanding speedily, as the advancements in the field of robotics, artificial intelligence and communication technologies are giving birth to new challenges. Here are some key areas for future research directions in the field of multi robot systems and their control. Few broader research directions are described in succeeding subsections.

6.2.1 Intelligent Multi Robot Systems

As the quantity of robots in a multi robot system rises more, guaranteeing the active coordination and an effective decision-making becomes complex. In such scenario future research direction can be producing such algorithms that empower thousands of robots in teams to work in an organized fashion without needing central control which is a typical idea stimulated from natural bio systems such as flock of the birds.

6.2.2 Robustness in Multi Robot System

Multi robot systems can face different challenges such as noise, parametric uncertainties, communication delays due to packet loss or link failure, control input delay, interruptions or environmental disturbances. Future research direction can be producing such self-curative systems with plug and play features where robots can negotiate such issues and further familiarize them to failures seamlessly meanwhile ensuring dispensing roles or revamping the collective function of the swarm intact.

6.2.3 Autonomy in Decision Making of Multi Robot Systems

There can be a challenge of making real-time decisions in dynamic and indeterminate environments. In such a scenario, the future research direction can be utilizing modern concepts of reinforcement learning and game theory which can boost decision-making autonomy and collaboration in multi robot systems, permitting them to proficiently accomplish tasks in the real-world circumstances without the human involvement.

6.2.4 Easy Human-Swarm Interaction

For the challenge of effective collaboration of human operatives with multi robot systems and examining more intuitive technological edges; the natural language processing and human-in-the-swarm approaches can permit non-skilled users to control the huge fleets of swarm systems.

6.2.5 Robotic Swarm based Data Fusion

Multi robot systems can gather, amalgamate and process data from various sources to build a precise prototype of the environmental conditions. This application of swarms can help us to explore distributed detection and awareness techniques, including simultaneous localization and mapping, empowering robots to form a cooperative understanding of their surroundings through the data fusion from different sources.

6.2.6 Energy Constraints of Multi Robot Systems

Multi robot systems functioning in severe or isolated environments encounter noteworthy energy related constraints. Research into energy harvesting, effective power administration and optimizing task allocation to range the effective time of robot swarms can be carried out effectively.
The above research guidelines indicate towards a future where multi robot systems are more smart, independent and pliable, influencing industries ranging from farming and logistics to space exploration and catastrophe response. The future research directions in the field of control and optimization specifically in the context of multi robot systems can include numerous promising areas as given in succeeding subsections.

6.2.7 Control of Heterogeneous Team of Robots having Higher Order Dynamics with Experimental Validations

In this thesis research work the robots with first-order dynamical models are considered; all operating in the two dimensional scenarios, however for the future research heterogeneous team of robots consisting of wheeled, climbing and aerial robots can be considered with higher-order dynamics and diverse tasks. Furthermore the developed algorithms as an outcome of research on heterogeneous team of robots will be deployed on physical platforms for practical implementation and experimental validation of simulated results presented in this thesis. With higher order dynamics, the constraining of accelerations can also be considered for future research as a high change in velocities is observed in the simulation results of this thesis. While resilient task execution has been demonstrated for systems with two, three, and up to four tasks performed by two, three, or four robots, the concept can be extended to large-scale systems. As the number of tasks, robots, and scenarios increases, greater resilience will be required. In such cases, resilience testing can be conducted, and a resilient index can be determined to quantify performance.

6.2.8 Improved Decentralized Optimization based Algorithms

Coordinating the robots in a multi robot system with decentralized control while ensuring the accomplishment of multi robot objectives are met, is difficult. Communication delays, packet losses or inadequate channel bandwidth can worsen overall performance in a decentralized multi robot system. To handle this challenge, a deep research into decentralized optimization based algorithms using local information for informed decision-making is required with guaranteed global coordination. This would enhance robustness while plummeting reliance on constant communication between robots in a multi robot system.

6.2.9 Mixed Optimization Approaches

Designing such multi robot system algorithms that proficiently unify universal and local optimizations is complex, especially when the desired objectives at diverse levels change dynamically. In such challenges, generating hybrid optimization frameworks become essential. The assigned tasks across many layers of control, where universal objectives such as multi robot task allocations are optimized at advanced levels, while each robot in a multi robot team concentrate on local tasks like formation control or obstacle avoidance.

6.2.10 Data-driven Multi Robot Control

Learning-based techniques like reinforcement learning typically involve huge quantities of data as well as training time, which can prove unfeasible in real-world applications of multi robot systems. Moreover, guaranteeing protection during the learning stage is too much important. In such challenge, developing such algorithms in which learning algorithms instead of CBF based approaches can be integrated with optimization techniques, can allow robots to learn complex control policies over time meanwhile confirming safety and proficiency. This can collectively improve malleability in indeterminate and dynamic environments.

6.2.11 Real-time Fast and Reliable Advanced Computation

Numerous optimization technique based problems in multi robot systems are NP-hard, denoting that they entail noteworthy computational resources to resolve. In real-world and real-time scenarios, optimization technique based algorithms go computationally expensive and complex, especially as the system expands. In such challenges, developing computationally effective optimization techniques that work in real time, conceivably through estimates, heuristics or ubiquitous processing, can empower efficient, speedy and consistent decision-making even in large-scale multi robot systems.

6.2.12 Multi Robot Systems in Uncertain and Stochastic Environments

Catering for uncertainty in robot motion, environment detection or inter-robot communications needs more complex models, which may make optimization problems inflexible or excessively conventional. In such challenge, encompassing control and optimization approaches to handle uncertainty explicitly utilizing stochastic optimization techniques can enhance performance and dependability particularly when robots function in volatile or dynamically fluctuating environments.

6.2.13 Multi Robot Systems with Multi-objective Optimization

Challenge of harmonizing multiple objectives such as task accomplishment, energy efficacy and safety can cause contradictory requirements during a multi robot based mission, making it tough to discover an optimum resolution that gratifies all standards in real-time. In such challenging scenario designing control systems that can manage trade-offs between multiple objectives in a dynamic way is well-needed. This approach utilizes multi-objective optimization theory based algorithms that provide an efficient set of solutions, permitting real-time adaptation to fluctuating mission priorities.

6.2.14 Mixed Criticality Multi Robot Tasks

For future research work, tasks with varying priorities (e.g., emergency vs. routine) can be introduced and tested to check that how the framework prioritizes them under resource constraints (e.g., limited robots or time). Though resource constraints in terms of execution of mandatory or emergent tasks have been simulated in this thesis. However, such a priority-aware task in case of complex resource constraints like limited time is a compelling extension for future work.

6.2.15 Parameter Robustness

A critical extension of this research involves systematic investigation of parameter sensitivity in the QP framework is to bridge theoretical guarantees with practical performance optimization. Future work can characterize how key CBF parameters (α, ρ, τ) govern the velocity overshoot and task completion efficiency, presented through quantitative heatmaps.

Bibliography

[1] M. Oprea, "Agent-based modelling of multi-robot systems," *IOP Conference Series: Materials Science and Engineering*, vol. 444, p. 052026, 11 2018.

[2] A. Rahmani, "Maritime and multi-agent autonomy (347n) jet propulsion lab (jpl)." https://www-robotics.jpl.nasa.gov/what-we-do/applications/teaming/, (accessed on Nov. 30, 2024).

[3] S. Duan, Z. Jiang, Y. Ren, and H. Wei, "Coordination of multi-robot systems," in *2022 International Conference on Electronics and Devices, Computational Science (ICEDCS)*, pp. 361–367, IEEE, 2022.

[4] H. Kabir, M. Tham, Luen, and C. Chang, Yoong, "Internet of robotic things for mobile robots: Concepts, technologies, challenges, applications, and future directions," *Digital Communications and Networks*, vol. 9, no. 6, pp. 1265–1290, 2023.

[5] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt, "The robotarium: A remotely accessible swarm robotics research testbed," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, no. 1, pp. 1699–1706, IEEE, 05 2017.

[6] G. Notomista, S. Mayya, Y. Emam, C. Kroninger, A. Bohannon, S. Hutchinson, and M. Egerstedt, "A resilient and energy-aware task allocation framework for heterogeneous multirobot systems," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 159–179, 2022.

[7] J. Cortes and M. Egerstedt, "Coordinated control of multi-robot systems: A survey," *SICE Journal of Control, Measurement, and System Integration*, vol. 10, pp. 495–503, 11 2017.

[8] P. G. F. Dias, M. C. Silva, G. P. Rocha Filho, P. A. Vargas, L. P. Cota, and G. Pessin, "Swarm robotics: A perspective on the latest reviewed concepts and applications," *Sensors*, vol. 21, no. 6, 2021.

[9] H. Chakraa, F. Guérin, E. Leclercq, and D. Lefebvre, "Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art," *Robotics and Autonomous Systems*, vol. 168, p. 104492, 2023.

[10] L. Parker, *Multiple Mobile Robot Systems.* Springer, 01 2008.

[11] E. Stump and N. Michael, "Multi-robot persistent surveillance planning as a vehicle routing problem," in *2011 IEEE International Conference on Automation Science and Engineering*, pp. 569–575, IEEE, 2011.

[12] A. English, P. Ross, D. Ball, and P. Corke, "Vision based guidance for robot navigation in agriculture," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1693–1698, IEEE, 2014.

[13] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems," *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 939 – 954, 2004.

[14] G. A. Korsah, A. Stentz, and M. B. Dias, "A comprehensive taxonomy for multi-robot task allocation," *The International Journal of Robotics Research*, vol. 32, pp. 1495 – 1512, 2013.

[15] M. Egerstedt, J. N. Pauli, G. Notomista, and S. Hutchinson, "Robot ecology: Constraint-based control design for long duration autonomy," *Annual Reviews in Control*, vol. 46, pp. 1–7, 2018.

[16] G. Notomista, S. Mayya, S. Hutchinson, and M. Egerstedt, "An optimal task allocation strategy for heterogeneous multi-robot systems," in *2019 18th European Control Conference (ECC)*, pp. 2071–2076, IEEE, 2019.

[17] L. E. Parker, *Heterogeneous Multi-Robot Cooperation.* PhD thesis, Massachusetts Institute of Technology, 10 1994.

[18] L. Iocchi, D. Nardi, M. Piaggio, and A. Sgorbissa, *Distributed Coordination in Heterogeneous Multi-robot Systems*, vol. 15. Autonomous Robots, 2003.

[19] A. Prorok, M. A. Hsieh, and V. Kumar, "The impact of diversity on optimal control policies for heterogeneous robot swarms," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 346–358, 2017.

[20] R. K. Ramachandran, J. A. Preiss, and G. S. Sukhatme, "Resilience by reconfiguration: Exploiting heterogeneity in robot teams," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6518–6525, IEEE, 2019.

[21] B. Gerkey and M. Mataric, "Multi-robot task allocation: analyzing the complexity and optimality of key architectures," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 3, pp. 3862–3868, IEEE, 2003.

[22] G. Notomista and M. Egerstedt, "Constraint-driven coordinated control of multi-robot systems," in *2019 American Control Conference (ACC)*, pp. 1990–1996, IEEE, 2019.

[23] Y. Emam, S. Mayya, G. Notomista, A. Bohannon, and M. Egerstedt, "Adaptive task allocation for heterogeneous multi-robot teams with evolving and unknown robot capabilities," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7719–7725, IEEE, 2020.

[24] G. Notomista, S. F. Ruf, and M. Egerstedt, "Persistification of robotic tasks using control barrier functions," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 758–763, 2018.

[25] G. Notomista and M. Egerstedt, "Persistification of robotic tasks," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 2, pp. 756–767, 2021.

[26] H. Fouad and G. Beltrame, "Energy autonomy for resource-constrained multi robot missions," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7006–7013, IEEE, 2020.

[27] S. Wilson, P. Glotfelter, L. Wang, S. Mayya, G. Notomista, M. Mote, and M. Egerstedt, "The robotarium: Globally impactful opportunities, challenges,

and lessons learned in remote-access, distributed control of multirobot systems," *IEEE Control Systems Magazine*, vol. 40, no. 1, pp. 26–44, 2020.

[28] E. Nunes, M. Manner, H. Mitiche, and M. Gini, "A taxonomy for task allocation problems with temporal and ordering constraints," *Robotics and Autonomous Systems*, vol. 90, pp. 55–70, 2017.

[29] L. Lin and Z. Zheng, "Combinatorial bids based multi-robot task allocation method," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 1145–1150, IEEE, 2005.

[30] F. Tang and L. E. Parker, "A complete methodology for generating multi-robot task solutions using asymtre-d and market-based task allocation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3351–3358, IEEE, 2007.

[31] M. Otte, M. J. Kuhlman, and D. Sofge, "Auctions for multi-robot task allocation in communication limited environments," *Autonomous Robots*, vol. 44, no. 3, 2020.

[32] M. Irfan and A. Farooq, "Auction-based task allocation scheme for dynamic coalition formations in limited robotic swarms with heterogeneous capabilities," in *2016 International Conference on Intelligent Systems Engineering (ICISE)*, pp. 210–215, IEEE, 2016.

[33] W. Mather and M. A. Hsieh, "Macroscopic modeling of stochastic deployment policies with time delays for robot ensembles," *The International Journal of Robotics Research*, vol. 30, no. 5, pp. 590–600, 2011.

[34] S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar, "Optimized stochastic policies for task allocation in swarms of robots," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 927–937, 2009.

[35] S. Mayya, S. Wilson, and M. Egerstedt, "Closed-loop task allocation in robot swarms using inter-robot encounters," *Swarm Intelligence*, vol. 13, no. 2, pp. 115–143, 2019.

[36] W. Abbas and M. Egerstedt, "Characterizing heterogeneity in cooperative networks from a resource distribution view-point," *Communications in Information and Systems*, vol. 14, no. 1, pp. 1–22, 2014.

[37] T. Balch, "Hierarchic social entropy: An information theoretic measure of robot group diversity," *Autonomous Robots*, vol. 8, no. 3, 2000.

[38] H. Ravichandar, K. Shaw, and S. Chernova, "Strata: Unified framework for task assignments in large teams of heterogeneous agents," *Autonomous Agents and Multi-agent Systems*, vol. 34, no. 2, p. 38, 2020.

[39] K. Lerman, C. Jones, A. Galstyan, and J. Matarić, Maja, "Analysis of dynamic task allocation in multi-robot systems," *The International Journal of Robotics Research*, vol. 25, no. 3, pp. 225–241, 2006.

[40] N. Palmieri, X.-S. Yang, F. De Rango, and A. Santamaria, Francesco, "Self-adaptive decision-making mechanisms to balance the execution of multiple tasks for a multi-robots team," *Neurocomputing*, vol. 306, pp. 17–36, 2018.

[41] S. S. Fatima and M. Wooldridge, "Adaptive task resources allocation in multi-agent systems," in *Proceedings of the Fifth International Conference on Autonomous Agents*, (New York, NY, USA), p. 537–544, Association for Computing Machinery, 2001.

[42] N. Iijima, A. Sugiyama, M. Hayano, and T. Sugawara, "Adaptive task allocation based on social utility and individual preference in distributed environments," *Procedia Computer Science*, vol. 112, pp. 91–98, 2017.

[43] Y. Nomura, "Environment-aware robotics," *NTT Technical Review*, vol. 15, no. 10, pp. 22 – 24, 2017.

[44] K. Saulnier, D. Saldaña, A. Prorok, G. J. Pappas, and V. Kumar, "Resilient flocking for mobile robot teams," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1039–1046, 2017.

[45] F. Bullo, C. J., and S. Martinez, *Distributed Control of Robotic Networks: a Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press, 2009.

[46] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multi-Agent Networks*. Princeton University Press, 07 2010.

[47] R. Arkin, *Behavior-based Robotics*. The MIT press, 1981.

[48] S. I. Roumeliotis and M. J. Mataric, “”small-world” networks of mobile robots,” in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, p. 1093, AAAI Press, 2000.

[49] W. Ren and R. W. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. Communications and Control Engineering, Springer-Verlag London, 2008.

[50] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *2019 18th European Control Conference (ECC)*, pp. 3420–3431, IEEE, 2019.

[51] C. Tomlin, G. Pappas, and S. Sastry, “Conflict resolution for air traffic management: a study in multiagent hybrid systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.

[52] S. Prajna, A. Jadbabaie, and G. J. Pappas, “A framework for worst-case and stochastic safety verification using barrier certificates,” *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415–1428, 2007.

[53] A. D. Ames, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs with application to adaptive cruise control,” in *53rd IEEE Conference on Decision and Control*, pp. 6271–6278, IEEE, 2014.

[54] A. Kordani, O. Rahmani, A. Nasiri, and S. Boroomandrad, “Effect of adverse weather conditions on vehicle braking distance of highways,” *Civil Engineering Journal*, vol. 4, p. 46, 02 2018.

[55] P. A. Ioannou and C.-C. C. Chien, “Autonomous intelligent cruise control,” *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 657–672, 1993.

[56] C.-Y. Liang and H. Peng, “Optimal adaptive cruise control with guaranteed string stability,” *Vehicle System Dynamics*, vol. 32, no. 4-5, pp. 313–330, 1999.

[57] C.-Y. Liang and H. Peng, "String stability analysis of adaptive cruise controlled vehicles," *JSME International Journal Series C*, vol. 43, no. 3, pp. 671–677, 2000.

[58] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 429–436, 2006.

[59] A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp. 143–153, 2003.

[60] A. Musa, M. Pipicelli, M. Spano, F. Tufano, D. N. Francesco, G. Di Blasio, A. Gimelli, A. D. Misul, and G. Toscano, "A review of model predictive controls applied to advanced driver assistance systems," *Energies*, vol. 14, no. 23, p. 7974, 2021.

[61] E. D. Sontag, "A lyapunov-like characterization of asymptotic controllability," *SIAM Journal on Control and Optimization*, vol. 21, no. 3, pp. 462–471, 1983.

[62] R. Freeman and P. Kokotović, *Robust Nonlinear Control Design: State-space and Lyapunov Techniques.* Springer, 1996.

[63] P. M. Shiroma and M. F. M. Campos, "Comutar: A framework for multi-robot coordination and task allocation," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4817–4824, IEEE, 2009.

[64] G. Lefranc, "Multirobot allocation in a flexible manufacturing system, using reinforcement learning for decision-making, case of study," *Procedia Computer Science*, vol. 221, pp. 41–48, 2023.

[65] K. Garg, n. Zhang, Songyuan, O. So, and C. Dawson, Charlesand Fan, "Learning safe control for multi-robot systems: Methods, verification, and open challenges," *Annual Reviews in Control*, vol. 57, p. 100948, 2024.

[66] G. Chiriatti, G. Palmieri, C. Scoccia, M. C. Palpacelli, and M. Callegari, "Adaptive obstacle avoidance for a class of collaborative robots," *Machines*, vol. 9, no. 6, 2021.

[67] M. Rubagotti, I. Tusseyeva, S. Baltabayeva, D. Summers, and A. Sandygulova, "Perceived safety in physical human-robot interaction-a survey," *Robotics and Autonomous Systems*, vol. 151, p. 104047, 2022.

[68] L. Scalera, I. Palomba, E. Wehrle, A. Gasparetto, and R. Vidoni, "Natural motion for energy saving in robotic and mechatronic systems," *Applied Sciences*, vol. 9, no. 17, 2019.

[69] J. Zeng, *Challenges of Control Barrier Functions: Optimization, Control, Planning and Navigation.* PhD thesis, University of California, Berkeley, 2022.

[70] H. Choset, K. M. Lynch, S. Hutchinson, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms and Implementations.* The MIT Press, 05 2005.

[71] N. Correll, B. Hayes, C. Heckman, and A. Roncone, *Introduction to Autonomous Robots Mechanisms, Sensors, Actuators and Algorithms.* The MIT Press, 2022.

[72] P. Lin, W. Ren, C. Yang, and W. Gui, "Distributed consensus of second-order multiagent systems with nonconvex velocity and control input constraints," *IEEE Transactions on Automatic Control*, vol. 63, no. 4, pp. 1171–1176, 2018.

[73] J. Fu, Y. Lv, G. Wen, X. Yu, and T. Huang, "Velocity and input constrained coordination of second-order multi-agent systems with relative output information," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 3, pp. 1925–1938, 2020.

[74] J. Fu, G. Wen, X. Yu, and T. Huang, "Robust collision-avoidance formation navigation of velocity and input-constrained multirobot systems," *IEEE Transactions on Cybernetics*, vol. 54, no. 3, pp. 1734–1746, 2024.

[75] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[76] L. N. Tan, "Distributed h-infinity optimal tracking control for strict-feedback nonlinear large-scale systems with disturbances and saturating actuators,"

IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 50, no. 11, pp. 4719–4731, 2020.

[77] M. Dias, R. Zlot, N. Kalra, and A. Stentz, “Market-based multi-robot coordination: A survey and analysis,” *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257–1270, 2006.

[78] D. Rachael N. and B. Marvin K., “A review on multi-robot systems categorised by application domain,” in *2017 25th Mediterranean Conference on Control and Automation (MED)*, pp. 701–706, IEEE, 2017.

[79] J. Chen, J. Wang, Q. Xiao, and C. Chen, “A multi-robot task allocation method based on multi-objective optimization,” in *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 1868–1873, IEEE, 2018.

[80] H. Huang, A. V. Savkin, M. Ding, and C. Huang, “Mobile robots in wireless sensor networks: A survey on tasks,” *Computer Networks*, vol. 148, pp. 1–19, 2019.

[81] S. Saravanan, K. Ramanathan, Chetty, R. MM., and M. Nilakantan, “Review on state-of-the-art dynamic task allocation strategies for multiple-robot systems,” *Industrial Robot*, vol. 47, pp. 929–942, 2020.

[82] C. Wang, S. Zhu, B. Li, L. Song, and X. Guan, “Time-varying constraint-driven optimal task execution for multiple autonomous underwater vehicles,” *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 712–719, 2023.

[83] G. Notomista, S. Mayya, M. Selvaggio, M. Santos, and C. Secchi, “A set-theoretic approach to multi-task execution and prioritization,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9873–9879, IEEE, 2020.

[84] K. Warnakulasooriya, R. G. Benton, and A. Segev, “Priority basis task allocation for drone swarms,” in *2023 AAAI Association for the Advancement of Artificial Intelligence*, AAAI Press, 02 2023.

[85] L. Wang, T. Qiu, Z. Pu, J. Yi, and J. Zhu, "Hedonic coalition formation for distributed task allocation in heterogeneous multi-agent system," *International Journal of Control, Automation, and Systems*, vol. 22, no. 2, p. 1212–1224, 2024.

[86] S. Mayya, D. S. Dantonio, D. Saldaña, and V. Kumar, "Resilient task allocation in heterogeneous multi-robot systems," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1327–1334, 2021.

[87] Y. Emam, G. Notomista, P. Glotfelter, and M. Egerstedt, "Data-driven adaptive task allocation for heterogeneous multi-robot teams using robust control barrier functions," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9124–9130, IEEE, 2021.

[88] N. Correll, B. Hayes, C. Heckman, and A. Roncone, *Introduction to Autonomous Robots Mechanisms, Sensors, Actuators and Algorithms.* The MIT Press, 12 2022.

[89] F. Gulzar, Y. A. Butt, and A. Iqbal, "Adaptive cruise control for ground vehicles using control barrier function under weather based surface conditions," in *2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pp. 573–578, 2022.

[90] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54–61, 2015.

[91] M. Rauscher, M. Kimmel, and S. Hirche, "Constrained robot control using control barrier functions," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 279–285, IEEE, 2016.

[92] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 03 2004.

[93] M. Egerstedt, "Graph-theoretic methods for multi-agent coordination," in *ROBOMAT 07*, pp. 3–7, Centro Internacional de Matemática, 2007.

[94] M. Hale and M. Egerstedt, "Convergence rate estimates for consensus over random graphs," in *2017 American Control Conference (ACC)*, pp. 1024–1029, IEEE, 05 2017.

[95] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[96] F. Gulzar, N. M. Khan, Y. A. Butt, and A. I. Bhatti, "Constraint-oriented formation control of multi-robot system in leaderless consensus under confined conditions," *Systems Science & Control Engineering*, vol. 12, no. 1, p. 2436666, 2024.

[97] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multi-vehicle cooperative control," *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.

[98] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," vol. 1, pp. 59–65, IPCB: Instituto Politécnico de Castelo Branco, 01 2009.

[99] H. Sira-Ramirez, A. Luviano-Juárez, M. Ramirez-Neria, and E. Zurita-Bustamante, *Active Disturbance Rejection Control of Dynamic Systems: A flatness based approach.* Butterworth-Heinemann, 10 2018.