Titre: Title:	Control Barrier Functions For Energy Sufficiency In Mobile Robot Systems
Auteur: Author:	Hassan Fouad
Date:	2023
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Fouad, H. (2023). Control Barrier Functions For Energy Sufficiency In Mobile Robot Systems [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/55835/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/55835/
Directeurs de recherche: Advisors:	Giovanni Beltrame
Programme: Program:	Génie informatique

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Control Barrier Functions For Energy Sufficiency In Mobile Robot Systems

HASSAN FOUAD
Département de génie informatique et génie logiciel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*
Génie informatique

Septembre 2023

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée :

Control Barrier Functions For Energy Sufficiency In Mobile Robot Systems

présentée par **Hassan FOUAD**
en vue de l'obtention du diplôme de *Philosophiæ Doctor*
a été dûment acceptée par le jury d'examen constitué de :

Tarek OULD BACHIR, président
Giovanni BELTRAME, membre et directeur de recherche
David SAUSSIÉ, membre
Gennaro NOTOMISTA, membre externe

DEDICATION

To my mother...

ACKNOWLEDGEMENTS

First, I would like to praise Allah the Almighty, the most gracious and the most merciful, and thank him for giving me the strength and courage to keep pushing through the hard times till the end of my PhD.

Second, I would like to say that a PhD is an extremely intense and rich experience, not only because of the various technical aspects or the never ending challenges and hurdles that one has to overcome, but also because of the people one interacts with and learn from.

I would like to express my deep and heartfelt gratitude to my supervisor, Prof. Giovanni Beltrame, and I am happy to say that it was a true pleasure working with him for the past years during my PhD. He not only provided me with consistently excellent technical advice and guidance through the years, but also he kept believing in me, even in times when I was not believing in myself. Moreover, he gave me a perfect example of how a true leader should be like, through his immense patience, top notch communication skills, and true passion for his work. I gladly admit that I learned a lot from this great display of leadership and I am happy to imitate it in the future.

Next, I would like to thank all my friends and colleagues in MISTLab, and it is safe to say that my interactions with my colleagues over the years was everything I could ask for. The numerous fruitful discussions and the consistent, and often times selfless, help I received from many of my colleagues greatly shaped the evolution of my PhD, and made it take the shape it has today.

I would like also to thank all my friends that I got to know during my PhD, as well as all my old friends. The power of true friendship provided me with an immense support, and the feeling of comradery made it easier to push through hard times. This power of friendship is a true asset for me, and I am really privileged to know such great friends.

Last but definitely not least, I would like to thank my family for all the support and motivation through the years. The sacrifices that my family made during my PhD can not be overstated, and this provided me with enough motivation and resolve. Quitting was not even an option for me because of my family and their unrelenting support through the highest of highs, and the lowest of lows. I am privileged to have such a family, and no amount of words will be able to express my gratitude towards my family.

RÉSUMÉ

De nos jours, la robotique moderne progresse à un rythme effréné et, au fil du temps, de nouvelles applications robotiques apparaissent. Dans ce contexte, de nouvelles applications robotiques ne cessent d'émerger, notamment en ce qui concerne les robots mobiles et les systèmes multi-robots.

En effet, Les nouveaux systèmes robotiques permettent non seulement d'étendre les capacités humaines, mais donnent également la possibilité à l'homme d'accomplir des tâches plus ardues dans des environnements plus difficiles et plus dangereux. Ceci engendre des conséquences sociales et économiques importantes et potentiellement dangereuses, créant des effets socio-économiques autant importants que ces conséquences.

Par conséquent, la capacité des robots mobiles et des systèmes multi-robots à atteindre une autonomie à long terme et à survivre dans leur environnement de travail est essentielle. L'un des éléments clés dans la quête de l'autonomie à long terme est la suffisance énergétique ainsi que l'autonomie énergétique, c'est-à-dire la capacité des robots à maintenir une charge suffisante tout au long de l'exécution de leur mission, quelles que soient les circonstances, indépendamment de la nature de la mission ou l'environnement.

Le maintien de la suffisance énergétique pose de nouveaux défis, en particulier lorsque les ressources de recharge disponibles dans l'environnement sont limitées, ce qui impose la nécessité de coordonner les activités de recharge entre les différents robots. Un autre défi se pose lorsque les robots opèrent dans des environnements inconnus et complexes, où la consommation d'énergie des robots peut ne pas être uniforme dans tout l'environnement, et/ou l'environnement lui-même peut être encombré et non structuré. Cela nécessite de réfléchir à des alternatives plus prudentes dans la recherche d'une station de recharge, tout en prenant en considération ces aspects environnementaux. En outre, assurer une suffisance énergétique pour les scenarios mentionnés représente un défi en soi, car ces garanties pourraient non seulement trop restreindre la capacité des robots à se déplacer, mais également nuire à leur capacité à bien mener leurs missions.

À la lumière de ces défis, nous présentons dans cette thèse deux cadres différents qui s'attaquent à ces problèmes en fournissant des garanties sur la capacité des robots à effectuer leurs missions. Le premier cadre garantit l'énergie pour les robots dans un système multi-robots ayant une station commune qu'ils partagent. Notre cadre utilise des fonctions de barrière de contrôle (CBF) pour formuler deux types de contraintes : l'une d'elles garantit que les robots s'approcheront de la station de recharge et arriveront avant que leur tension ne descende en dessous d'un certain niveau fixé, et l'autre garantit l'utilisation mutuellement exclusive de la station de recharge par les robots.

Au cours du développement de ce cadre, nous formalisons et décrivons la capacité d'une station de recharge, à savoir le nombre de robots qu'une station peut desservir, compte tenu de la capacité de la station, ainsi que des propriétés de décharge des robots et de l'écart souhaité entre les heures d'arrivée des robots et les heures de départ d'une part, et de la séparation souhaitée dans les temps d'arrivée des robots à la station d'autre part. Nous montrons la validité de ce cadre à travers les résultats d'une campagne de simulations, et par des résultats expérimentaux utilisant des robots KheperaIV et des drones Cognifly.

Le second cadre garantit la suffisance énergétique des robots opérant dans des environnements inconnus et complexes. Nous tirons parti de la capacité d'un planificateur d'exploration à créer des chemins sûrs et viables dans une carte construite incrémentalement par un robot au fur et à mesure qu'il se déplace dans l'environnement, et nous utilisons les CBF pour fournir des garanties de suffisance énergétique sur ces chemins, c'est-à-dire que le robot peut parcourir les chemins produits par le planificateur de chemins de façon à ce qu'il arrive à la station avant que son budget énergétique ne soit épuisé. Pour définir les contraintes sur les chemins linéaires par morceaux du planificateur de chemins, nous proposons une méthode de lissage de trajectoire basée sur la sigmoïde qui produit une trajectoire paramétrée continue plus facile à intégrer dans les contraintes de suffisance énergétique. En effet, Nous montrons que notre cadre est adaptable à des robots avec des modèles dynamiques plus élaborés, en l'adaptant à un robot monocycle. Nous démontrons également la validité de notre cadre par le biais d'une campagne de simulation dans un ensemble d'environnements labyrinthiques, et par des résultats expérimentaux qui mettent en évidence l'application de notre cadre sur un mini rover AgileX Scout. En outre, Nous fournissons une discussion où nous passons en revue certaines des principales caractéristiques des cadres proposés, et où nous signalons certains problèmes liés au développement de cette thèse. Finalement, nous concluons en discutant de certaines limitations et en donnant des suggestions pour les travaux futurs.

ABSTRACT

Modern-day robotics is advancing at an astounding rate, and as time goes by new robotic applications keep emerging, especially considering mobile robots and multi-robot systems. New robotic systems promise to extend human abilities and enable humans to deal with more arduous tasks in harsher and more dangerous environments, with potentially significant social and economic effects.

Therefore, the ability of mobile robots and multi-robot systems to achieve long-term autonomy and survive in their work environments for extended periods of time is becoming more and more necessary, with the main aim of unlocking the full potential of these robotic systems, especially in real-life deployments. One key component in the quest for long-term autonomy is energy sufficiency, namely the ability of robots to maintain sufficient charge throughout their mission execution, regardless of what the mission or environment may be.

Maintaining energy sufficiency poses new challenges, especially when the available charging resources in the environment are limited, thus imposing the need to coordinate charging activities among different robots. Another challenge is when robots operate in unknown and complex environments, where robots' power consumption may not be uniform across the environment, and/or the environment itself could be cluttered and unstructured, which calls for more careful ways of seeking recharge, taking these environment-related aspects into account. Moreover, guaranteeing energy sufficiency for the aforementioned scenarios is a challenge in itself, that is because these guarantees could overly restrict the robots' ability to carry out their missions.

In light of these challenges, we present in this thesis two different frameworks that tackle these problems by providing guarantees on energy sufficiency for robots in two different scenarios. The first framework ensures energy sufficiency for robots in a multi-robot system having a common charging station that they share. Our framework uses Control Barrier Functions (CBFs) to formulate two types of constraints: one that ensures that robots will approach the charging station and arrive before their voltage drops below a certain set level, and one to ensure mutually exclusive use of the charging station by the robots. During the development of this framework, we formalize and describe the capacity of a charging station, namely the number of robots a station can serve, given the robots' discharging properties and the desired separation in arrival times of robots to the station. We show the validity of this framework through the results of a simulations campaign, and through experimental results using KheperaIV robots and Cognifly drones.

The second framework ensures energy sufficiency for robots operating in unknown and complex environments. We leverage the ability of an exploration planner to create safe and viable paths in a map that is being incrementally built by a robot as it moves in an environment, and we use

CBFs to provide guarantees on energy sufficiency over these paths, i.e. the robot can traverse the paths produced by the path planner so that it arrives to the charging station by the time its energy budget is depleted. To define the energy sufficiency constraints over piecewise linear paths from the path planner, we propose a sigmoid-based path smoothing method that produces a continuous parameterized path that is easier to integrate into the energy sufficiency constraints. We show that our framework is adaptable to robots with more elaborate dynamic models, and we do so by adapting our framework to a unicycle robot. We demonstrate the validity of our framework through a simulation campaign in a set of maze environments, and through experimental results of applying our framework on an AgileX Scout Mini rover. We provide a discussion where we review some of the main features of our proposed frameworks, and report some issues related to the development in this thesis. We conclude by discussing some limitations and giving suggestions for future work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

CBF	Control Barrier Functions
CLF	Control Lyapunov Function
CLBF	Control Lyapunov Barrier Function
RBF	Reciprocal Barrier Function
RCBF	Reciprocal Control Barrier Function
HOCBF	Higher Order Control Barrier Function
NBF	Non-smooth Barrier Function
NCBF	Non-smooth Control Barrier Function
BNCBF	Boolean Non-smooth Control Barrier Function
PACBF	Parameter Adaptive Control Barrier Function
RACBF	Relaxation Adaptive Control Barrier Function
HJ	Hamilton Jacobi
LTI	Linear Time Invariant
SMID	Set Membership IDentification
IQC	Integral Quadratic Constraints
ACLF	Adaptive Control Lyapunov Functions
ZBF	Zeroing Barrier Function
ZCBF	Zeroing Control Barrier Function
CS	Charging Station
SPC	Sequential Path Construction
ES	Energy Sufficiency
MRS	Multi-Robot System
QP	Quadratic Program
LTA	Long-Term Autonomy
RL	Reinforcement learning
KKT	Karush–Kuhn–Tucker
SLAM	Simultaneous localization and mapping
MC	Mission Controller

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

1.1 Context and motivation

Modern robotics have promising potential for becoming widespread in their use in everyday life. There is a plethora of applications where robots, especially multi-robot systems, can carry out arduous tasks and interact with dangerous environments. Some examples of such applications include: underground [2] and planetary exploration [3], mining [4], removal of landmines [5] and precision agriculture [6] among many others.

One important feature that such robot systems must have to maximize their benefit and potential is long-term autonomy, i.e. the ability to successfully and consistently run missions in different types of environments over extended periods of time. To this end, a multi-robot system must have a set of abilities and features that include the ability to maintain connectivity among robots and with ground stations, ensuring Energy Sufficiency (ES), and adapting to any given environment with time through learning.

A key element in achieving long term autonomy is energy sufficiency: the ability of robots to maintain a certain amount of energy during a mission, guaranteeing that they never run out of energy. Energy sufficiency can be achieved with the ability to recharge during mission execution when needed without compromising the mission.

To be effective, a method to maintain energy sufficiency has to be mission-agnostic (meaning that it is applicable to any mission), modular, and as little intrusive as possible to the robot mission. Moreover, the method has to be adaptable and applicable to a wide range of robot models and environments.

In this thesis, we deal with some central challenges related to energy sufficiency of mobile robot systems for long-term autonomy.

Resource access coordination in multi-robot systems

One central challenge for long-term autonomy is to provide guarantees that all robots have enough energy during mission execution, especially in situations where there is a limited number of charging stations available. In such cases, robots have to coordinate the use of an available charging station. This is in part because no more robots should arrive at a station for recharging more than the station can handle. Moreover, some types of missions carried out by multi-robot systems could be severely affected if several robots decide to opt out of the mission in favor of recharging, e.g.

coverage and patrolling missions. Consequently, the coordination behavior we seek needs to avoid robots waiting near the charging station and to incentive robots to arrive to an available station at different times. Therefore to achieve coordination, a robot's arrival time at a station needs to be adequately estimated and then manipulated to ensure no other robots have the same arrival time, thus achieving proper use of available charging resources.

Another issue that is tightly connected to the previous point is the need to understand the limits of a charging cycle with robots sharing a charging station. In other words, given a maximum arrival time of a robot to a station, we need to know how many robots can be serviced by this station. This idea is motivated by the observation that the number of robots that can be serviced by a single charging station cannot be arbitrary, as it depends on the discharging and charging characteristics of robots. This knowledge is important for the planning of field deployments and for successfully sizing the charging requirements of a multi-robot system in a field mission.

Operation in unknown and unstructured environments

Mobile robots often operate in unknown and unstructured environments. In these cases, robots have to generate maps and use a path planner to navigate the environment and return to their charging stations. Keeping track of the energy necessary to return to the charging station independently of the way the path is generated is essential for successful deployment in a mission-agnostic way. Ensuring ES on a path produced by a general path planner is challenging for a number of reasons. First, the amount of energy needed to traverse a given path needs to be adequately estimated, and this estimation is based on the path geometry, and having an accurate representation of power consumption along the path. Second, the fact that a path planner can change the path it produces in an abrupt manner at a certain frequency calls for the need to ensure that such path updates do not violate ES, i.e. the new path does not need more energy to traverse than is available. Third, the paths produced by path planners are often described in terms of waypoints in the environment, which may require employing path smoothing techniques to produce smooth paths that are more convenient to handle.

Translation of simple kinematic models to more realistic ones

Using simple kinematic models to express robot states has been a common practice in robotics, especially with applications that involve planning for robots. This is due to the fact that such models are relatively simple to use while being able to capture the essential states that concern a system designer in applications where planning is involved. However, the manner a robot consumes energy is often more accurately expressed with more elaborate models. For instance, the energy consumption of a unicycle-type robot could be more accurately expressed in terms of linear and

angular speeds, while a single integrator model does not have such states readily available. For this reason, there is a need to account for potential mismatches in energy consumption due to the model simplification.

1.2 Problem Statement

In this thesis we are concerned with fulfilling energy sufficiency to achieve long term autonomy of multi-robot systems. Endowing mobile robots, especially multi-robot systems, with such ability paves the way for more sustainable robot missions in realistic environments. In this thesis we attempt to tackle the following core challenges:

1. The lack of mission-agnostic, decentralized and flexible energy sufficiency frameworks that inherently handle coordination constraints among robots to access the available charging stations, and be adaptable to a wide range of environments;
2. The prevalent method of using single integrator models to describe robot states poses a challenge with modeling energy consumption with respect to the simplified set of states, which gives rise to additional challenges to adapt control actions derived using such a simplified model to actual robot kinematics;
3. Unclear understanding of the relationship between how robots coordinate, their individual properties, and the number of serviceable robots by a charging station.
4. The lack of simple and intuitive methods for path smoothing that produce parameterized continuous paths that are arbitrarily close to the piecewise continuous path produced by path planners.

1.3 Research Objectives

In the light of the previous challenges, and motivated by the positive potential impact long term autonomy can have socially and economically, the research objectives we adopt in this thesis are:

O1: Ensure ES for multi-robot systems while achieving the following goals:

- Be mission-agnostic: the method we seek should be compatible with any robot mission.
- Handle multi-robot coordination of Charging Station (CS) use in a decentralized manner.
- Provide guarantees on energy sufficiency.

O2: Mathematically define station capacity and provide the relationship between such capacity and robots' properties and mission properties.

O3: Ensure ES in unknown and unstructured environments, in a manner compatible with a wide range of path planners, while being mission-agnostic and compatible with a wide range of robot models.

1.4 Research Contributions

1. **FW1**: An ES framework based on CBFs that guarantees ES for robots in a multi-robot system, as well as proper coordination among robots to ensure proper coordinated access to a shared charging station. Moreover, we formulate station capacity for the framework and elaborate key relationships with robot and mission properties. This work is described in detail in Chapter 4 were we develop the base framework with a simplified battery discharge model, and Chapter 5 where we further develop the framework with a more accurate battery discharge model, enhanced coordination framework and more developed station capacity formulations. This work was published as

    * *H. Fouad and G. Beltrame, "Energy Autonomy for Resource-Constrained Multi Robot Missions," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 2020, pp. 7006-7013.*
    * *H. Fouad and G. Beltrame, "Energy Autonomy for Robot Systems With Constrained Resources," in IEEE Transactions on Robotics, vol. 38, no. 6, pp. 3675-3693, Dec. 2022.*

2. **FW2**: An ES framework based on CBFs that guarantees ES for a robot operating in an unknown environment. This work has three sub contributions: (a) A method that leverages the ability of an exploration planner to explore and construct maps for unknown environments, while using CBFs to provide guarantees on ES of a robot over paths produced by the path planner. (b) A simple and quick method for path smoothing that gives parameterized continuous paths closely tracking the original piecewise continuous paths. (c) A method for adapting the framework developed for a simple robot described by a single integrator model to a robot described by a unicycle model. These ideas are presented in more details in Chapter 6, and this work was published as

    * *H. Fouad, VS. Varadharajan, and G. Beltrame. "Energy Sufficiency in Unknown Environments via Control Barrier Functions." Submitted to the International Journal of Robotics Research, 2023.*

1.5 Research Impact

We envision that our proposed methods will have a tangible effect on the field of long term autonomy for multi-robot systems. The continuous operation of robot fleets is a topic of extreme importance, especially considering the current transition to industry 4.0 and the increased presence of robots in manufacturing, logistics, health. An example is a permanent drone show, where the UAVs representing "pixels" in the sky can be constantly replaced, as long as the overall power input from the available charging stations matches the overall energy used by the UAVs in the air. This approach can also be applied to fleets of monitoring robots (e.g. checking a perimeter) to extend the range and duration of operation of the whole system. **FW1** provides a solution to this class of problems, since it provides a method that enables multiple drones to share a charging station (that could be extended to multiple stations), and it provides sizing tools to properly estimates the amount of needed charging stations.

On a larger scale, consider that Electric vehicles (EVs) have gained much attention and favour by many companies, industries and even individuals due to their lower harmful impact on the environment. Moreover, this has been bolstered by recent advances in Li-Po batteries, their capacities and recharging technologies. One example of such a shift in interest is that at the time of writing this thesis, Amazon was planning to buy 100,000 EVs from Rivian, an EV manufacturer, a deal estimated to be valued at ten billion dollars. In this context ensuring that such delivery EVs have proper access to charging facilities when needed is an integral part of the process to ensure the sustainability of the system. **FW2** can be used to add a modular energy sufficiency layer to these EVs, along with a path planner, to seamlessly help guiding them to recharge when needed. Consequently, **FW2** could be useful for the transportation and logistics sector in general, as it is expected to depend more on EVs as time goes by.

Additionally, the ability of **FW2** to enforce energy sufficiency over arbitrary paths is a useful asset in applications involving exploration like cave or planetary exploration. Also, **FW2** can be applicable for humanitarian demining and precision agriculture, where it is often the case that robots need to move on specific paths through a map in the environment (e.g. irrigation patterns).

During the course of developing the methods in this thesis we took part in the development of the Cognifly drone, which was used extensively in the experiments of the multi-robot coordination of charging activities, which is described in more detail in Chapter 5. The Cognifly is currently one of the popular options for cheap and collision resilient drones, and has been used in several outreach programs for high school students in Montréal. The Cognifly development is published as

* de Azambuja, Ricardo, et al. "When being soft makes you tough: A collision-resilient quadcopter inspired by arthropods' exoskeletons." 2022 International Conference on Robotics and Automation (ICRA). IEEE, 2022.

CHAPTER 2 LITERATURE REVIEW

In this chapter we provide a literature review of the related body of work pertaining to the main themes in this thesis. The main topics we cover are:

- A brief discussion about long term autonomy in multi-robot systems: its signification and the potential impact it can have on society.
- Evolution of energy awareness in multi-robot systems.
- Control Barrier Functions (CBFs): which is the main tool we use in our framework to enforce energy related and coordination constraints. The main ideas we discuss pertaining to CBFs are
    * Chronological description of the development of several main ideas related to CBFs.
    * Discussion of some properties of CBFs.
    * Current development regarding using CBFs and different learning techniques.
    * Different applications of CBFs.

2.1 Long term autonomy mobile robot systems

Long Term Autonomy (LTA) is the ability of a robot or multiple robots to operate in their intended environments for extended periods of time. This can be seen as the natural progression of lab tested robots that although can do many types of tasks, they do not carry them out persistently for extended periods of time. Having robots that can consistently do a mission for a long time is a lucrative idea due to the potential socioeconomic gains they can bring about, specially with recent advances in sensory and computational abilities of robots that makes them more versatile and well equipped to take on complex tasks.

Effective long term autonomy would benefit several domains and types of missions that have direct economical benefit as in construction [7, 8] and mining [4, 9], can save human lives as in humanitarian demining [10, 11, 12] and search and rescue [13, 2], can provide assistance in daily life tasks like in service robots [14] and tour guide robots [15, 16], or help explore and understand different environments (potentially for future exploitation) as in cave exploration [17], space exploration [18, 19, 3] and studying oceans [20]. Therefore the motivation for achieving long-term

autonomy in different types of robot systems is strong owing to the benefits using robots persistently can bring in terms of saving human lives, achieving economic growth, and enhancing the living conditions of many people.

Several properties, thus, should exist in robots capable of doing long-term deployments. One such important quality is robustness and adaptability, which implies that a robot should be able to handle changes in its properties or environment during the course of its operation. Some examples include robustness in localization in indoor environments [21], outdoor environment [22], and dynamic environments [23]. Also, various machine learning techniques are employed to enable robots to learn from their environments, especially in situations where there is limited access to training data [24]. Moreover, they have to be self-sufficient, meaning that they should be able to seek the resources they need autonomously mid-mission. In the next section, we discuss one aspect of self-sufficiency needed for LTA, namely energy sufficiency.

2.2 Energy sufficiency in multi-robot systems

As discussed earlier, one key ingredient for the long-term autonomy of systems of individual or multiple robots is the ability to maintain enough energy to carry out their mission for extended periods of time, thus achieving energy sufficiency. Herein, we present a brief account of the historical development of energy awareness in single and multi-robot systems.

Energy awareness in robots and more generally dynamic systems has been a domain of interest for researchers for a long time. Early incarnations could be seen in the form of designing "energy efficient" systems that consume less energy. The domain of optimal control is full with such examples, in which a cost function can be designed and minimized such that it minimizes the energy consumption of the system. One example is the work in [25] in which a cost function is designed for a robotic manipulator that tends to minimize the execution time and energy expenditure of the system, and a numerical solution for the resulting two point boundary value problem was presented.

Another example can be found is [26] in which an approximate method was developed to produce sub-optimal but real time path planning for a manipulator mounted on a free flying space robot, so as to minimize the energy waste due to viscous friction in motors. In [27] an approximate method was presented to estimate energy-minimizing paths for a mobile robot on a terrain. The problem was tackled by converting the continuous space of the terrain into a discrete graph by introducing discrete nodes on boundary edges of terrain faces(patches of a map) and connecting them with edges which are weighted appropriately based on energy expended due to friction and gravity (according to the inclination of velocity vectors at each of these nodes on the original terrain), and

then a discrete optimization problem was formulated to find the discrete path with the least energy expenditure.

Node scheduling The previous examples more or less fall into the planning phase for a single robot. When it comes to multi-robot systems, this line of thought of optimally utilizing the available energy was extended. The field of sensor networks has been one of the initial areas in which such ideas were implemented. The main idea is having a group of sensors covering a certain region, and then it becomes a question of activating and deactivating sets of these sensors so as to utilize the redundancy in sensor coverage to minimize the total energy used by the network. This problem is known as node scheduling [28].

In [29] the authors tackled the problem of a stochastically placed sensor nodes in a sensor network with high redundancy in coverage (several parts of the map can be covered by many nodes). Their approach introduces a centralized algorithm for generating mutually exclusive sets of sensors, such that each set covers the map sufficiently, then operate each of these sets exclusively for a certain amount of time before switching to the next, leading to a decrease in energy consumption.

A similar concept was used in [30] in the sense that redundancy in the coverage of the sensors was used to save energy, but through using a more decentralized approach. The idea is simply to let each sensor node sleep and then wake up randomly with a certain probability. When a node wakes up it probes the neighbouring nodes to see if there are other active neighbours in which case it goes back to sleep, or starts working otherwise.

Another example of decentralized techniques to achieve the same idea was presented in [31], in which a decentralized algorithm was described for sensor networks with high redundancy. The idea is to ensure the existence of a minimal number of awake nodes that form a connected backbone that preserves the capacity of the original network, in the sense that no congestion in messages takes place at any of the nodes. A decentralized algorithm was proposed for electing these awake robots according to their connectivity to other nodes and their energy.

A slight variation in the theme of node scheduling is having nodes that can vary the sensing range as described in [32]. There, the basic idea is that random sets of nodes are selected at each round to be active, and then these active nodes apply certain rules to adjust their sensing ranges so as to maintain coverage, and then it was shown that using such rules for picking the sensing range can lead to an overall reduction in energy consumption compared to the case of using uniform sensing ranges like in [29].

The idea of using adjustable sensing ranges presented in [32] and the idea of finding set covers presented in [29] were mixed together in [33] to formulate what is called Adjustable Range-Set Covers (AR-SC) problem. In this problem it is required to find set covers (subsets of sensors) as

well as sensor ranges, such that the number of these sensor bundles is maximized, a group of M targets can be monitored by each of these set covers (sensor bundles) and that each sensor node can only consume a constrained amount of energy.

An integer programming formulation of this problem was given, as well as a linear programming approximation of the solution, all suited to centralized implementation. Moreover, they provided a greedy heuristic that can be done in a decentralized way to find these set covers.

With the development of mobile robots, much attention has shifted towards maximizing the utility of mobile robot networks for doing various types of missions, especially coverage and surveillance. For example, in [34] the problem of robot deployment of robots for search and rescue scenarios was investigated. The goal was to calculate the least number of mobile robots sufficient for dynamically covering a certain area given power constraints on each robot's energy consumption and time constraints about the deadline required for finishing the assigned task.

In [34] a policy for picking a fixed speed for each robot that maximizes its traversed distance was provided. Moreover, a deployment strategy was described that maximizes the covered area and determines the number of groups of robots and number of robots in each group sufficient to cover the area of concern.

Another interesting direction when it comes to utilizing multiple mobile robots is the idea of allocating tasks to robots such that robots with higher energy content have higher workload. One of the pioneering examples for using such approach for covering missions is [35]. In this work, a Lloyed like algorithm was utilized (robots chasing centroids of a voronoi diagram). The difference is that the metric used for defining the voronoi tessellations was modified such that it includes the energy content of each robot, resulting in robots with more energy having bigger tessellations to cover, leading to energy aware task re-partitioning.

Along similar lines goes the work in [36] where a model for the quality of a coverage sensor was provided, and then an energy aware waypoint selection method was described. The aim of this method is to target the portion of the mission when more time is spent in spacial distribution than in local coverage (when most of the desired area is covered and robots are "fine tuning" to cover the small missed patches). The waypoint to be selected by each robot should be such that it avoids previously covered areas, and the proposed approach then puts the majority of this redistribution load on robots with longest operational life and smallest history of actuation effort.

The same idea was also employed in different applications like energy aware rendezvous [37]. In this work an optimal control problem was formulated to let a group of robots decide when and where to meet, according to the initial voltage level of each robot. This rendezvous should be made in such a way to ensure that the voltage level is always non-negative. For that end, an optimal

control problem was formulated to calculate the optimal input acceleration profile for the moving robot that minimizes energy consumption and to derive a relation between voltage level and time of execution. Another optimization problem was then formulated that minimizes the spatial and temporal differences between robots, and constrained with the aforementioned relation and solved in a distributed manner in a very similar way to consensus problems.

Path planning techniques are also used for planning missions for multi-robot systems, but rather in a centralized manner. Much like the word done in [27], the authors in [38] the authors tackle the problem of path planning of multiple autonomous underwater vehicles in a general flow field. The authors provide cost functions for minimum time and energy (based on provided kinematic and energy models) to go from one point to another, and provide a graph search-based optimal trajectory planning (similar in the idea to that in [27]) so as to figure out the paths that minimize energy or time.

Charging stations In the previously presented ideas, the main theme was basically utilizing the available energy of a multi-robot system through proper planning or repartitioning of tasks. Another popular technique that has gained more attention recently is the possibility of having charging stations to actively add energy to the multi robot system. These charging stations can be stationary or moving.

Stationary charging stations The basic idea is to have a number of charging stations in the area where the robots execute their mission. Then at some point each robot goes to a charging station when energy runs critically low. One of the pioneering works in this direction is [39] that discusses the problem of several robots sharing a number of charging stations. This paper introduces some inequalities to approximately describe the capacity and the viability of the system (i.e. whether the number of charging stations is sufficient), as well as a simple algorithm that enables robots to share recharging resources using simple behaviors as building blocks, like activity, waiting, and searching.

One of the most important questions related to this direction is: how to determine the appropriate time to abandon the mission and go to recharge? It could be argued that the simplest solution is to use a fixed set threshold on energy or battery voltage [40], however, It was shown in [41] that using an adaptive threshold gives better results. In [42] the authors use an adaptive threshold for voltage that depends on the current battery state of charge, and then estimate the distance it can travel and compare it to its current distance from a charging station. Also [43] adopt an adaptive threshold for battery voltage to prompt returning to recharge, and this threshold is adapted by formulating a risk factor described by a probability distribution function that indicates the probability of having less

energy to return to the CS than the currently available energy, and their proposed solution balances between this risk and the gain the robot would get in terms of mission execution if it continues its mission.

One problem with the methods mentioned above is the fact that they do not provide hard theoretical guarantees that a robot will be sufficiently powered at all times. The introduction of Control Barrier Functions [44, 45, 46, 47], which is a method that mainly concerns providing guarantees on the safety of system execution, provided a solution for the lack of guarantees in [41, 43].

For example, in [44] and [47] a control barrier function was provided for energy persistence of a group of robots, each having a charging station. This proposed barrier function provides guarantees on having the voltage of each robot above a certain threshold. This barrier function is mixed with a nominal controller that does a coverage mission through quadratic programming, to result in a minimally invasive execution, meaning that the final control action will be the nominal as long as the states of the system are far away from the boundary of the safe set (i.e. as long as the energy is not near from the minimum allowable value).

The CBF approach is advantageous in the sense that it provides an answer to the question of required threshold to abandon the mission in favour of a recharge. Moreover, it is a flexible and versatile tool that can be used to "mix" different constraints to get more realistic behaviour from the system as in [46]. In this paper the authors propose an approach to compose CBFs by mixing smaller CBFs that server as building blocks.These behaviours include energy persistence, connectivity, collision avoidance and others.

The CBF approach was also employed in [45] to achieve energy persistence for a group of robots doing a coverage mission, with the difference that the recharging process doesn't happen in a specific charging station, but rather at certain parts of the map to be covered, similar to recharging by solar power, with having some areas with more sun light that others.

Although CBF methods provide guarantees on energy sufficiency, the works discussed so far do not handle situations in which multiple robots share a charging station, which could be a common scenario in many multi-robot missions. One solution was introduced in [48] where the authors provide a framework that enables robots in a multi-robot system, carrying out a frontier exploration mission, to choose a charging station among a set of available stations, and for the robots assigned to a certain station to coordinate so that they use the CS in a mutually exclusive manner. The authors adopted a market-based approach in which each robot places bids for using the CS depending on its available energy, distance from CS and distance from its nearest frontier.

A different approach was adopted in [49] where a group of robots carries out rendezvous on a regular basis so they can plan, and coordinate the use of a charging station. The problem was

formulated as an instance of a workforce scheduling problem which is NP, and aims to minimize the deprecation in mission quality due to robots needing to rendezvous, while at the same time properly using the available charging station. An evolution algorithm heuristic solution was then adopted, so that robots can all solve the problem in real time, broadcast their results, agree of plan and redo rendezvous later.

Another approach to tackle a similar problem was presented in [50] using game theory. This paper tackled the problem of having a number of charging stations and robots that have to share these stations. The problem is modelled as a special case of job-shop scheduling problem [51], with the assumption that all recharging processes take the same time. The authors propose a formulation of the resource allocation problem as a non-cooperative game, and propose an algorithm for calculating the unique pure strategy Nash Equilibrium point for this game (given that each robot knows positions of other robots), leading to a unique allocation of the charging stations.

From the previous methods discussed thus far [43, 48, 52], we can notice that they all do not provide hard theoretical guarantees on system performance regarding ES and they are often time very invasive in the sense that they are ES methods that are designed for a specific type of mission, making it less flexible and usable for other missions [48]. Moreover, some of these methods could be computationally demanding as in [49].

One interesting question that is also related to static charging stations is: how to place these charging stations to make sure that the robots will have access to them when needed? Authors in [53] tried to provide an answer to this question. Two problems were formulated: the first concerns finding the minimum sufficient number of charging stations given the number of robots and the minimum voltage threshold. The second is the inverse problem: given the locations of the charging stations, what is the minimum allowable voltage threshold. These minimization problems were solved with algorithms using satisfiable modulo theory.

The question of locating charging stations was also treated in [54] in the context of warehouse robots. The idea of this paper is to model the warehouse as network of nodes representing candidate locations for charging stations, with the links being the cost of travelling between these locations. The demand of recharge at each of these nodes varies with the operating conditions, and the goal is to determine a set of nodes (of the original set of nodes describing the warehouse) that creates a connected subgraph embedded in the subset of nodes with highest demands (solving an online online connected dominating set problem) to make sure there is a charging station with reach for all robots when needed.

Moving charging stations Another recharging modality that can be traced in the literature as well is using moving charging stations that can move to deliver a recharging service when needed.

Applications for such an approach vary widely from patrolling and surveillance [52, 55], area coverage [56], vehicle routing problems [57], transportation [58] and many others. The main motivation behind this direction is the possibility of reducing the detour the robots have to do to recharge compared to the case of static charging stations.

In [52] the problem of persistent robots doing surveillance mission was covered. Each robot goes in a predefined route and the main requirement is that these robots are able to replenish their energy without detouring from their specified routes. For that end, a group of recharging depots carrying batteries are tasked with dropping batteries at some locations along each route of each robot so that robots can collect them with no detour. The problem is then formulated as a generalized multiple depot travelling salesman problem in which a complete graph containing the robots and their potential replenishment points is defined, and then the main goal is to design cyclic routes for a number of these depots along this graph (the start node is the end node) such that the total distance covered by each depot is minimized, while assuring that at least one point of each robot's route is visited and all routes for all robots are visited. A heuristic algorithm called multiple depot random select was introduced to solve the problem, with both centralized and decentralized solutions.

A slightly different approach was adopted in [56]. The main goal was to do an energy persistent coverage mission by a group of robots in a GPS-deprived environment. The idea was to deploy the robots with some potential-based control law using only local interactions, and then from the resulting formations groups of virtual trees are formed for subgroups of the robots with charger robots deployed to each of these trees. Predecessor-successor relations are then established between robots, and these relations are used so that robots can exchange their positions to create an energy gradient towards the charging station (i.e. robots with lower energy nearer to station) without compromising the coverage.

Interestingly, the work in [55] can be seen as a prequel for the work in[52], as the same problem was tackled of having a group of robots doing surveillance tasks along predefined routes. However instead of having charging depots, there are charging robots that rendezvous with the robots at certain points along each robot's path, and it is required to design the path of the charging robots to meet with mission robots once to recharge them without detouring, while minimizing a certain cost function. A graph based representation for the problem including charging robots as well as potential charging locations as vertices was introduced and a mixed integer linear program was introduced and solved for small example case.

The work in [58] touches upon a very interesting point that can be seen as lying between the realms of static and dynamic charging station, which is: what is the effect of placing charging stations in specific locations in the field along the robot paths? what is the effect of the potential spacial interference on the performance of the mission? These question are interesting in the sense that

it can shed some light on the relation between charging station locations and mission execution quality. The authors argue that placing the charging stations along the robot paths (for a simple transportation mission at least) deteriorates the performance and the best way is to put it nearby. The authors then propose an algorithm, assuming that these charging stations are mobile, so that they adjust their locations and reduce the effect of spacial interference.

2.3 Control Barrier Functions

After giving quick glimpses of evolution and advancement in energy aware and energy persistent multi robot systems, we give in this section a brief account for the evolution of the idea of control barrier functions, which constitutes the workhorse for our energy persistence algorithms.

The presentation in this section will be divided into three main parts: (i) Discussion of the basics of control barrier functions and their evolution (ii) Examples of using control barrier functions in different applications (iii) Some of the relevant theoretical development.

2.3.1 Basics of control barrier functions

Control barrier functions (CBFs) have recently gained popularity as tools for ensuring system safety in safety critical systems [47]. What we mean by ensuring safety is providing guarantees that a system evolves in such a way that its states (or a subset of its states) stay in a safe set. Other tools have existed in the literature to verify safety for safety critical systems, and these methods mostly depend on estimating reachable sets of system states to verify that such sets don't intersect with unsafe sets. For example [59, 60] provide tools for estimating the reachable sets of linear hybrid systems for safety verification, [61] provides methods for checking the stability of piecewise affine systems by casting them as verification problems and [62] that introduces a Hamilton-Jaccobi formulation of reachable sets, provide algorithms to estimate them and use this knowledge to formulate a differential game for collision avoidance of two airplanes.

The main drawback of such methods is the need to explicitly know how system trajectories evolve, either approximately or numerically, which is not generally an easy task. Motivated by this, the notion of barrier certificates was introduced in [63] which defines Lyapunov-like functions, called barrier certificates, the existence of which guarantees the safety of the system. These functions are defined as [63]

$$B(x) > 0 \quad \forall x \in \mathcal{X}_u \tag{2.1a}$$

$$B(x) \leq 0 \quad \forall x \in \mathcal{X}_0 \tag{2.1b}$$

$$\frac{\partial B}{\partial x} f(x,d) \leq 0 \quad \forall (x,d) \in \mathcal{X} \times D \tag{2.1c}$$

where $\mathcal{X}_u$ is the set of unsafe states, $\mathcal{X}_0$ is the set of initial conditions, d is the system input and D is the space of all input values. The idea is simply to say if for a specific system we can provide a function B that is positive in the unsafe region, and ensure that B is always decreasing, then the system states never go to the unsafe set.

This work has been extended in [64] to stochastic systems, and much like converse Lyapunov theorems a converse theorem for the barrier certificates was introduced in [65]. The main advantage of such approach is that it provides guarantees on the safety of the system without the need of estimating its reachable sets [63].

A slight modification for the aforementioned barrier certificates (2.1) is to take its complement $B(x) = -h(x)$ [47], in which case the safe set would be defined for $h(x) > 0$ and the problem would be to ensure that this safe set is forward invariant. The author in [66] provide a review on the methods and applications of positive invariance of sets in control engineering, especially theorem 3.1 (Nagumo 1942) which was a first attempt to drive sufficient and necessary conditions on positive invariance of sets, and is essentially a similar result for what was introduced in [63] but using the complement of the barrier certificate.

Barrier certificates, much like Lyapunov functions, provide guarantees, but don't answer the question of how to choose the input in such a way that attains these guarantees. For that, the notion of control barrier functions was introduced in [67], which draws inspiration from control Lyapunov functions (CLFs) [68] and [69]. In [67] control affine systems were treated and an explicit formula of a control law was given that guarantees the safety of the system.

The next step to the work in [67] is to think about a way to mix the safety requirement (which has a control action respecting it) with another control action that satisfies a certain requirement. For that end several methods can be found in the literature to mix system requirements with safety guarantees.

For instance, in [70] this was done by introducing the control Lyapunov barrier function (CLBF) to study the stabilization of a control affine system with safety guarantees. The main idea behind this approach is to find a bounded continuous function W that 1. has a negative time derivative with respect to unforced system field outside the unsafe region 2. be positive in the unsafe set and then it can be proved that the system will be stabilized, without having the trajectories passing through the unsafe region. The authors also proposed a way for constructing such CLBF given that there is a CLF and a CBF available.

Another approach that has gained even more traction, and was almost developed around the same time, is the use of quadratic programming to mix control Lyapunov and control barrier functions

[71]. In this work, the focus was on control affine system defined as

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u \tag{2.2}$$

where $\mathbf{x} \in \mathbb{R}^n$ are system states (for n states), and $\mathbf{u}$ are system inputs (for m inputs). The proposed approach in [71] was that if there is a CLF for the system that stabilizes it in some sense (as defined in [69]) and a CBF that guarantees safety in some sense (as defined in [67]), then both can be mixed with a quadratic program that takes the following form

$$\begin{aligned} u^* = \underset{\bar{\mathbf{u}}=\begin{bmatrix}\mathbf{u}\\ \delta\end{bmatrix}}{\text{argmin}} \quad & \frac{1}{2}\bar{\mathbf{u}}^T H \bar{\mathbf{u}} + F\bar{\mathbf{u}} \\ \text{subject to} \quad & L_f V + L_g V^T \mathbf{u} + c_3 V \leq \delta \\ & L_f B + L_g B^T \mathbf{u} \leq \frac{\gamma}{B} \end{aligned} \tag{2.3}$$

where V and B are CLF and CBF of the system respectively, $L_f V = \frac{\partial V}{\partial \mathbf{x}}^T f$ and $L_g V = \frac{\partial V}{\partial \mathbf{x}}^T g$ are the Lie derivatives of the function V in direction of f and g , $c_3 > 0$ and $\gamma \geq 0$ are constants and δ is a relaxation parameter that will sacrifice the priority of system stabilization if it will cause the system states to go to the unsafe set.

According to [72], such approach has an advantage over the technique proposed in [70] in that it mediates between executing the mission (following the CLF) and ensuring safety (following the CBF), putting more emphasis on safety (through the relaxation of CLF condition when needed), while in [70] it would be difficult to construct CLBF if the mission and safety requirements are contradictory. Moreover, the technique in [71] is easier to construct, and algorithms for solving quadratic programs are quick and numerically efficient and suitable for online implementation.

Thus far, the control barrier functions defined were basically to ensure system safety, or in other words, to make sure that the system will not go to an unsafe region. Another way to look at the same problem is to ensure system safety by formulating the problem as a set invariance, in the sense that we need to ensure a system will stay in a defined safe set, rather than ensuring that it will not wander away to an unsafe set.

The authors in [73] tackled this point through defining the reciprocal barrier functions (RBFs) and the zeroing barrier functions (ZBFs). Both definitions were based on defining the safe set $\mathcal{C}$ as

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\} \tag{2.4a}$$

$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = 0\} \tag{2.4b}$$

$$Int(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\} \tag{2.4c}$$

where h is a continuously differentiable function. The main goal is to keep the interior of this safe set forward invariant (i.e. if the system starts safe, it stays safe for all times in the future). A ZBF is defined as being this previously mentioned h if there is an extended class $\mathcal{K}$ function α, and a superset $\mathcal{D}$ of the safe set $\mathcal{C}$ ($\mathcal{C} \subseteq \mathcal{D}$) such that

$$L_f h(x) > -\alpha(h(x)) \tag{2.5}$$

where $L_f h(x)$ is the lie derivative of h in the direction f (for a control affine system). It could be shown [73] that the presence of a ZBF for a system implies the forward invariance of $\mathcal{C}$. The RBF is similar in essence to the control barrier functions defined in [67], and its main purpose is to steer the system away from unsafe sets. However, in [73] it was defined such that it depends on h. More rigorously, a continuously differentiable function B is RBF if there exist class $\mathcal{K}$ functions $\alpha_1(h(x)), \alpha_2(h(x)), \alpha_3(h(x))$ such that for all $x \in \int(\mathcal{C})$

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))} \tag{2.6a}$$

$$L_f B(x) \leq \alpha_3(h(x)) \tag{2.6b}$$

These barrier functions so far don't depend on inputs, and they act as certificates in the sense that their existence guarantees the safety of the system. In [73] the definition of the control versions of the RBF and ZBF were given. Thus, a continuously differentiable function h is called zeroing control barrier function (ZCBF) if for some set $\mathcal{D}$ such that $\mathcal{C} \subseteq \mathcal{D}$, there exists an extended class $\mathcal{K}$ function α such that

$$\sup_{u \in U}[L_f h(x) + L_g h(x) u + \alpha(h(x))] \geq 0, \forall \in \mathcal{D} \tag{2.7}$$

and it was shown that having a ZCBF for the system implies that the safe set $\mathcal{C}$ is forward invariant. Likewise, for a continuously differentiable function B is called a RCBF if there exist class $\mathcal{K}$ functions $\alpha_1(h(x)), \alpha_2(h(x)), \alpha_3(h(x))$ such that for all $x \in Int(\mathcal{C})$

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))} \tag{2.8a}$$

$$\inf_{u \in U}[L_f B(x) + L_g B(x) u - \alpha_3(h(x))] \leq 0 \tag{2.8b}$$

and as mentioned earlier these control barrier functions can be a part of a QP problem that mediated between following a mission and ensuring safety as in equation (2.3).

2.3.2 Theoretical aspects

In this section, we present some of the relevant theoretical extensions and modifications to the basic idea, along with some applications for which these modifications are relevant.

Robustness of CBFs

In [72] the authors tackled the problem of establishing the robustness properties of zeroing barrier functions to bounded uncertainties to the system model. This was done by applying somewhat similar idea to [74](chapter 8) about the stability of periodic orbits. So the main theme for showing robustness is by first showing the asymptotic stability of the safe set $\mathcal{C}$ defined in (2.4), then based on this result it was shown that there is a neighbourhood around the safe set, in which the safe set is still asymptotically stable for the perturbed system. Moreover, it was shown that solving the QP problem for similar to (2.3) and takes the form

$$\begin{aligned} u^* = \underset{\mathbf{u}}{\text{argmin}} \quad & \frac{1}{2}\mathbf{u}^T H \mathbf{u} + F\mathbf{u} \\ \text{subject to} \quad & L_f V + L_g V^T + c_3 V \leq \delta \\ & L_f h + L_g h u + \alpha(h) \geq 0 \end{aligned} \tag{2.9}$$

for a zeroing control barrier function with relative degree one ($L_g h \neq 0$) results in having an optimal solution u^* which is locally lipcshitz continuous (which ensures the existence of unique solution for the system given this input).

It is also worth mentioning that the QP in equation (2.9) is well suited in cases where the access for a CLF V that fulfills the desired mission is available. A formulation similar to the QP in (2.9) is by solving the following constrained least squares (CLS) problem [44]

$$\begin{aligned} u^* = \underset{\mathbf{u}}{\text{argmin}} \quad & \frac{1}{2}||\mathbf{u} - \mathbf{u}_{nom}||^2 \\ \text{subject to} \quad & L_f h + L_g h u + \alpha(h) \geq 0 \end{aligned} \tag{2.10}$$

where $||.||$ is the Euclidean norm, and $\mathbf{u}_{nom}$ is the nominal control action that the robot executes related to its mission. It is to be noted that the QP in this case acts as an arbitrator, in the sense that it tries to adhere to the mission as much as possible, unless this leads to getting out from the safe set, in which case the mission is abandoned and the safety control action is adopted.

High relative degree control barrier functions

One of the problems that may take place while trying to find the "safe" control action that satisfies (2.7) is to have a ZCBF with a higher relative degree than one, meaning that $L_g L_f^{\rho-1} = 0, L_g L_f^{\rho} \neq 0$ (where ρ is the relative degree of the system), which means in other words that the control barrier function has to be differentiated ρ times till the control action u shows up. This is problematic because simply the term $L_g h$ in (2.7) will be equal to zero and the QP problem can't be solved this way.

Several solutions were proposed in the literature to deal with this problem [75, 76, 77]. In [75] reciprocal control barrier functions were used, and a safe set was defined to ensure proper gait motion for a bipedal robot. A backstepping approach was adopted to define a new RCBF that contains the sought control action, then it was shown that if this new control barrier function satisfies a condition similar to (2.8a), it can be guaranteed that the set $\mathcal{C}$ (safe set) will be forward invariant (i.e. safety will be guaranteed).

In [76] the problem was treated in a slightly different manner, although having a similar backstepping flavour. The authors in this work tackle the problem of using quadratic programmes to mix between control lyapunov functions and control barrier functions, but they use input output feedback linearization to render a nonlinear system to a linear one, which makes finding a CLF a straight forward task for the linearized system. Moreover, the notion of virtual input output linearization was introduced and utilized with the CBF, motivated by the fact that the original input has already been used to input output linearize the original system dynamics. As a consequence, the high relative degree CBF was transformed into a linear controllable canonical form system whose output is the original CBF.

Based on this linearized representation of the higher relative degree CBF, a linear controller can be designed to ensure that the dynamics of the stability of the resulting transformed system for the CBF. Such type of CBF for which a linear combination of its derivatives serve as the stabilizing (third) term in (2.8a) was called an exponential control barrier function in [76].

A similar, yet more general approach was adopted in [77] with the difference that a zeroing control barrier function was used. The main idea was to do a similar input output linearization step for the system with the barrier function being as a pseudo output, but functions α_i were added at each differentiation along the way. After that a group of sets C_i were defined for each of these differentiations , and the main idea is to pick each of these α_i functions to ensure forward invariance of each of these sets C_i. A special case would be taking these α_i functions as linear functions to give the same result in [76]. This sort of barrier function with generic α_i functions in their definitions was called the higher order control barrier function (HOCBF) in [77].

Feasibility of Quadratic Programs for CBFs

One of the central issues that come up while dealing with CBFs is the feasibility of the quadratic program. Feasibility issues while dealing with CBFs could be caused by conflict between constraints and/or conflicts between the constraints and control limits, leading to no solution satisfying all constraints simultaneously.

To solve the issue of contradicting constraints, the authors in [78] proposed a technique for mixing barrier functions through multiplication and addition, giving a similar effect of union and intersect operators (OR and AND operators) for barrier functions. The main advantage of such technique is its mathematical simplicity, but the fact that it isn't easily translatable to zeroing barrier functions [79] (as it depends on the definition of barrier functions in [67]) which makes it lose many of the robustness properties in [73]. The authors in [46] proposed an idea to compose such compound barrier functions using min and max operators to give the same effect of AND and OR operators proposed in [78]. The main goal in [46] is to propose a constraint based approach for robots to survive for long periods of time, drawing inspiration from animals that act within certain constraints for their survival, changing the priorities of each of these constraints depending on the surrounding conditions.

The main challenge in the idea proposed in [46] is dealing with the non-smoothness of the min and max operators. The authors in [79] tackled this issue by introducing the non-smooth barrier functions (NBFs) that depend on their definitions on tools from non-smooth analysis defined in [80]. Moreover, sufficient conditions for the validity of the barrier function were given and the proposed tools were used to introduce the min/max composition of barrier functions.

This work has been extended in [81] for non-smooth control barrier functions (NCBF) for control affine systems with possibly discontinuous inputs. For this, the conditions necessary for having a control barrier function were modified by generalizing the notion of inner product and defining it for the generalized derivative of the discontinuous barrier function. Based on these results, the boolean non-smooth control barrier functions (BNCBF) were defined as results of boolean operations on such NCBFs. The gradient conditions necessary for ensuring forward invariance were modified for QP application, and an example application of leader follower navigation with collision avoidance and connectivity maintenance was provided.

Another direction to study the issue of QP feasibility is through studying the properties of the constraints of the QP and how they may impede each other. One example is in [82] that studies conditions for existence of control actions satisfying multiple inequalities simultaneously, and these conditions express the need of having two CBF constraints to be pointing in the same direction (i.e. $L_g h_i(x)$ for $i = 1, 2$ having the same sign), and if not then further conditions should be satisfied to

ensure that the two CBF constraints have an intersection (i.e. control action u that satisfies both). A similar idea was presented in [83], in which CBFs with constraints "pointing" in the same direction were called non-interfering. Moreover, they offer conditions for which solutions exist in case of having non-interfering constraints, and in case of interfering constraints they provide an algorithm that restrains system states from evolving in the direction of state space regions where violations of non interference could take place (through introducing more constraints).

Studying QP stability is also concerned with studying the effects of input boundaries on performance. One popular way to deal with this issue is through using penalty functions and relaxation functions [84]. In this work the authors presented Parameter Adaptive CBF (PACBF) and Relaxation Adaptive CBF (RACBF). In PACBF a set of penalty functions are defined as a part of the formulation of the higher order CBF (multiplied by the equivalent of the $\alpha(.)$ in (2.10)), and auxiliary dynamics were defined for these functions and the CBF constraints are applied for the augmented system consisting of the original system plus the auxiliary dynamics, which provides an ability for the system to adapt with control action bounds. In RACBF a similar set of parameters, albeit less in number, are defined with auxiliary dynamics and augmented system, but the formulation is done in such a way they act like relaxation parameters rather than penalty parameters multiplied by the $\alpha(.)$ function in (2.10). A similar extension to this theme is presented in [85] through using Auxiliary Variable CBF with the aim of reducing the number of parameters that need to be tuned in PACBF and RACBF. In this method similar variables to penalty functions were used in the definition of the HOCBF, albeit with a different structure than PACBF, with auxiliary dynamics defined for these variables and constraints were constructed to the full augmented system.

Dealing with uncertainties

One main issue with our dicussion of CBFs so far is the fact that they need an accurate model of the system to validate the guarantees given by these CBFs. However, in reality it is often the case that system models are not accurate: system parameteres may not be correct, besides the fact that unmodelled dynamics may exist. Some of the most popular methods that emerged recently to deal with this issue use tools from robust control [86, 87, 88, 89, 90], adaptive control [91, 92, 93, 94], and machine learning. In this section we focus on robust and adaptive methods, and in the next section we give more details on learning methods.

The main idea behind robust CBF methods is to add an extra margin of conserativeness to the CBF constraint, through adding an extra uncertainty related term, in a manner that compensates for uncertainty in dynamics. For example in [88] a method was introduced to deal with uncertainties in state estimation through using an upper bound for state disturbance as the extra term in the CBF constraint and showing that enforcing such constraint on the state estimate will lead to forward

invariance of the safe set for the actual states. In [90] an extension to Hamilton Jaccobi reachability method was introduced through using CBFs as to reduce the conservativeness of original HJ methods. In this work the sense of robustness is essentially manifested in the formulation through minimizing the effect of disturbances in a CBF formulation, while using the CBF as a value function for the HJ optimization problem. One added advantage of this method is that it retrieves the biggest possible safe set in the state space, but it suffers from the curse of dimensionality.

Other works deal with robustness from the perspective of the effect of input dynamics. One such example is in [87] which deals with the problem of system input being affected by uncertain elements so that the actual control input affecting the system has uncertainty. The way the authors in [87] deal with this issue is through creating a modified set of dynamics a modified input (affected by uncertainty) and through upper bounding the effect of this uncertainty they could add an extra uncertainty related term to the QP constraint which adds enough conservativness to account for input uncertainty. The authors in [86] tackled a similar issue for input input uncertainty to model the effect of unmodelled dynamics and lags of actuators. The concept of solution is similar in essence to [87], but they use integral quadratic constraints (IQC) to formulate the uncertainty term in the CBF constraint. The authors consider that the actual system input has a component that passes by a filter F which has LTI dynamics. The IQC signifies roughly the difference in energy between the actual uncertain input that goes to the system and the expected filter F output. We note that in both [87, 86] the constraints of the quadratic program become quadratic instead of linear due to direct dependence of these constraints on the magnitude of the input values, which makes the solution of the optimization a bit more computationally expensive.

Another philosophy of dealing with uncertainty in systems is through adding some extra parameters to the system equation and adapting these parameters using system output, which gives rise to adaptive CBFs. One early example in this regard is [91] which uses a system model where there is a set of unknown parameters that are being adapted using an adaptation law in a manner similar to adaptive control Lyapunov functions (ACLF) [95]. The adaptation law is made such that the values of the parameters follow the gradient of a CBF to make it increase to ensure forward invariance of the safe set. A central issue with this work is the fact it was incapable of approaching the boundary of a safe set, making it very restrictive. This work has been enhanced and extended to the case of hybrid systems in [93] and in this work an attempt was made to reduce the conservativeness of the method in [91] by introducing a hybrid adaptation law with a hysteresis-like mechanism. Adaptive CBFs for fully actuated and under actuated nonlinear systems were discussed in [92]. In this work the control action for the system was considered a summation of two control actions, one that follows and CBF-CLF QP problem similar to (2.9) using a set of estimated states, and another control action that counts for uncertainty and tries to reduce the estimation error between true states and estimated ones.

Although robust and adaptive methods do a good job handling system uncertainty, some works tend to mix both and adopt a philosophy of starting with some form of a conservative system representation through high degree uncertainty, then adapting the system using system output data to reduce this uncertainty with time, which leads to more robust and easier formulations than adaptive techniques. One example is [96] that provided a solution to the issue of excess conservativeness of [91] by adopting a robust CBF formulation akin to [88, 89], by introducing an additional term to the CBF constraint formulation with parameters signifying uncertainty level, then adapting the system by using set membership identification technique (SMID) on system output data to find values of these parameters that reduce overall system uncertainty and enlarge the safe set. A similar essense could be found in [89] where the uncertainty was modeled in terms of a set of unknown parameters (in a manner similar to adaptive methods like in [91]) and then a minimization problem for the uncertainty gradient was used as the extra term in the CBF constraint, and a duality principle was used to modify the problem to a set of regular linear constraints that a QP can handle while accounting for uncertainty effects at the same time. A similar SMID approach to [96] was adopted to reduce the uncertainty through parameter adaptation.

CBFs and learning techniques

Over the past 3-5 years we can notice a noticeable growth in the literature with respect to works incorporating CBFs with learning techniques and vise versa. This merging serves many purposes like dealing with system uncertainty [97, 98, 99, 100], finding policies for non-control affine systems [101], provide safety guarantees to RL policies [102, 103], tune CBF parameters for better performance [104] and construction of CBFs from data [105, 106, 107] among many others. In this section we give a brief account for some of these applications.

One of the main issues of CBFs is the fact that it may be challenging to formulate, depending on the application, and arbitrary construction of such functions may lead to confining the system in a smaller safe set than the true one. Therefore construction of CBFs from data is a central theme when it comes to applications of machine learning in CBF. One example is [106] in which a method was presented to construct CBFs from expert demonstrations. The authors defined a subset of the state space where the validity of CBF is guaranteed and provided a set of conditions to form a loss function that ensures some desired continuity properties for the CBF and its derivatives. The CBF is obtained by parameterizing the CBF and minimizing the loss function wrt these parameters. In [107] the authors use Support Vector Machine (SVM) to construct a CBF from sensor data for navigation application. The idea of the method is to collect a dataset of points that are labelled as safe or unsafe, train a SVM classifier that gives posterior probability a data point is safe or not, and this probability could be converted to a CBF that is utilized in a QP to filter system's control

actions. The method presented in [105] also tackles a navigation task in a cluttered environment, but the CBF is expressed as a linear function with learnable parameters and they adopt an incremental learning technique to adjust these parameters.

A closely related work to construction of CBFs is the construction of barrier certificates, which can prove useful for non-control affine nonlinear systems [101]. In this work the authors aim to generate control actions driving a non-control affine nonlinear system to a target goal set without passing through unsafe sets. They use neural networks to represent a barrier function and a Lyapunov-like function, and provide a loss function that satisfies certain desired qualities for these two functions (positive definiteness and proper signs of derivatives). The network is trained and the policy is obtained by minimizing this loss function.

Another important application for using learning techniques is mitigating the effects of system uncertainty. One popular way to model uncertainty in nonlinear systems is by using Gaussian Processes (GP) [98, 103]. In [98] the authors use a GP to model nonlinear system uncertainty and the main idea is to start with a conservative estimate of disturbance with a small safe set, then expand this safe set as more data is collected. The CBF was parameterized and a maximization problem for the safe set was formulated, and a QP was adopted with the updated CBF constraints to maintain safety. Adaptive sampling was adopted to reduce numerical complexity, where the new sampled points are more dense around regions of state space that are more uncertain. Similarly in [103] disturbance was modelled as a GP, but the goal was to train a Reinforcement Learning (RL) policy with the CBF-QP framework incorporated in the training phase. This was done by introducing a differentiable optimization framework that can encorporate the QP as Karush–Kuhn–Tucker (KKT) conditions (linear conditions) to be part of the reward in the learning.

The use of CBFs in RL is one of the currently active areas of research that attracts a lot attention. The motivation behind this is the fact that RL does not have explicit guarantees on safety. One example of tackling this problem is [102] that aims to produce RL policies that are inherently safe using CBFs. The authors use GP to model disturbance, similar to [98, 103] and they formulate CBF constraints that provide stochastic safety guarantees. In order to guide the policy in safe direction, they modified the CBF constraint to include the history of safe control actions from previous episodes and they show that safety is still guaranteed.

We can find in the literature another body of work concerning using RL methods to enhance CBF performance and deal with system uncertainty. For example, [99] mitigates the effect of system uncertainty while providing safety and tracking guarantees. The main idea is to formulate CBF-CLF constraints while injecting an input dependant on a nominal (inaccurate) model which results in having extra error terms due to the model mismatch. Some more tunable parameters were added to the constraints and a RL agent was trained to drive these error terms to zero, thus negating

the effect of model mismatch. In [104] the authors formulate a regular CBF constraint with a set of tunable parameters (akin to the $\alpha(.)$ function in (2.9)). Parameters are intended to be tuned to reduce CBF constraint excess conservativeness, prevent the system from getting stuck under CBF and enhancing quality of trajectories. The authors provided two methods for CBF parameter tuning. The first is a brute force method in which trajectories are explored in state space to populate a dataset that is used later to train a neural network, and the second is to use RL to explore these parameter values. This could be argued to be similar in essence to other applications where RL is used for parameter tuning, e.g. [108].

2.3.3 Applications of CBFs

There is a great body of work in the literature related to the applications of control barrier functions. Many of these applications have to do with safety critical systems [71, 109, 110, 111, 112], as well as other applications like energy persistification [44, 45] and designing robot behaviours [113].

In [109] the problem of collision avoidance between two airplanes was tackled. The main property of such problem is the existence of actuator saturation. The main challenge was the fact that the combination of nonlinear dynamics and actuator saturation makes simple CBFs comprised of functions as distances between two aircrafts are not enough. For that a specific barrier function for this application that guarantees safety in a certain time horizon was introduced, and the control action can be calculated numerically.

In [110] a very interesting review of set invariance methods was given including tools from viability theory and Lyapunov stability theorem with the aim of designing control laws to keep the system in a safe set. The main goal of the paper is to introduce a robust QP formulation for a CBF problem, in which the system parameters are uncertain, and this concept was implemented to ensure safety of a segway robot, in the sense that it has to stand vertical, even if it takes a disturbance input.

One of the pioneering examples of using CBFs in real applications was [71], in which a modified version of the control barrier function defined in [67] was introduced, where a direct connection between the barrier function and set invariance was established. A CBF formulation was introduced for collision avoidance between two double integrator point masses representing two cars, and a QP was defined to alter the original behaviour of one of the cars in a minimally invasive manner to ensure it won't collide with another car if it moves with its original velocity.

Another popular application for using CBFs is collision avoidance in multi-robot systems [111]. In this paper a similar formulation for collision avoidance to what was in [71] was used. A centralized version of the problem was constructed, in which a central agent solves the QP problem related to the collision avoidance CBF, knowing all the positions of all robots. Moreover, a decentralized

version of the same problem was introduced, in which each agent solves the QP problem for its immediate neighbours without the need to know the locations of every other agent.

One of the interesting applications using CBFs is bipedal robot motion [114]. In this paper a CLF-CBF QP formulation (similar to what has been adopted in [71]) was used for controlling the gait motion of a bipedal robot, while enforcing some safety constraints as avoiding overhead obstacles and landing on precise footholds. Also in [75] a similar problem was tackled but the problem was formulated in such a way that the relative degrees of the barrier functions were higher than one, and thus a backstepping approach was proposed for deriving the control action for achieving set forward invariance (and thus safety) using the same QP formulation in [71].

Another interesting idea was addressed in [115] in which a generalized framework was presented for solving gradient descent problems using a control barrier function formulation. The main idea is to consider the negative of the potential function as being a zeroing barrier function and use the same QP formulation to minimize the potential function, but here while using the CBF formulation as being a constraint on the minimization in a decentralized manner. This idea has been implemented to achieve a formation task and a coverage task for a multi robot system.

Another interesting application of CBFs is the work in [113], which presents an idea for guaranteeing the fulfillment of the correct graph structure required for a multi-robot system to do a certain behavior or segment of a mission when the robots switch from one behavior to another, utilizing tools from CBFs. For that end, a finite-time control barrier function (FCBF) was defined that guarantees the convergence of the system states to a certain desired safe set in a finite time, drawing inspiration for that from [116]. The rest of the idea is to define the different safe sets corresponding to each desired graph structure and use a CBF based framework to make the robots converge to such desired graph structures in a minimally invasive manner, without the need to do an explicit rendezvous and picking the right graph from the resulting complete graph, which interferes with the mission execution.

CHAPTER 3 RESEARCH APPROACH AND THESIS ORGANIZATION

In this chapter, we give a brief description of the deliverables of the thesis, as well as a concise description of the main ideas in the thesis. and we link different deliverables to the research objectives defined in the Chapter 1. We conclude this chapter by giving a description of the thesis layout.

In this thesis, we develop two frameworks for ES:

(**FW1**) Energy sufficiency for multi-robot systems sharing a charging station.

(**FW2**) Energy sufficiency for robots in unknown environments.

FW1: Energy sufficiency for multi-robot systems sharing a charging station (O1 and O2)

In the development of this framework, we are interested in scenarios in which robots of a multi-robot system are carrying out a mission, with one shared charging station that they need to share for recharging. The purpose of this framework is twofold:

- Provide guarantees on the energy sufficiency of the robots in the system, meaning that no robot will have its energy running below a certain threshold.
- Provide guarantees on coordination of the charging station usage to ensure mutually exclusive use of the charging station.

The development of **FW1** should ensure that the robots are able to maintain the required energy levels irrespective of their missions (energy agnostic), and to achieve as little interference to the original mission being carried out by the robots as possible (minimally invasive), and to be easily applicable to any type of control action (flexible and modular). The use of CBFs aids with achieving these goals as we demonstrate in Chapters 4 and 5.

Another key idea that we explore during the development of **FW1** is the idea of charging station capacity, and by this we mean the number of robots serviceable by one charging station given their discharge properties, as well as the desired separation in robots' arrival times to the station. We also highlight the relation between this capacity and the coordination framework in **FW1**.

In our development of **FW1** we start by developing a basic version of the framework in Chapter 4, where we use a simple formulation in which the robot dynamics are modeled as a single integrator, and a linear battery voltage discharge model. We define the ES and coordination frameworks within **FW1** by introducing CBF constraints for the robot's position w.r.t. CS in relation to its battery

voltage (similar to [44]), and other CBF constraints for pairwise difference in robots' arrival times to the CS to achieve desired coordination. We also derive expressions for CS capacity based on the ES and coordination frameworks developed in Chapter 4. In Chapter 5 we develop the treatment presented in Chapter 4 by adopting a more accurate battery discharge model, modifying the ES and coordination CBF constraints, and providing a more developed treatment of the CS capacity that allows for a larger number of serviceable robots than that was presented in Chapter 4.

FW2: Energy sufficiency for robots in unknown environments

In the development of this framework, we tackle one of the central issues regarding the ES aspect of **FW1**, namely the limitation of robots moving only in straight lines. The main idea is to pair the capabilities of a path planner to find sound paths in an environment, and the ability of CBFs to provide performance guarantees to produce a framework that enables a robot to ensure ES in a complex and potentially unknown environment, irrespective of the mission it carries out. The development of **FW2** is presented fully in Chpater 6.

We present a path smoothing method based on sigmoid functions that can produce smooth parameterized paths from a set of waypoints, such that the smooth path is arbitrarily close to the piecewise linear path defined by these waypoints. Given the ability of the path smoothing method to create parameterized paths, we proceed with developing **FW2**, which aims to provide guarantees on ES of a robot over a path produced by a path planner. This framework has several favorable properties: compatibility with any mission type (mission agnostic), compatibility with any path planner (modularity), and adaptability to more elaborate robot dynamics (flexibility and extensibility).

The core idea behind **FW2** is to create a reference point along the path, which is facilitated by the nature of the smoothed path, and modify the energy sufficiency constraints from **FW1** to move this reference point along the path in a manner that ensures its arrival to the CS by the time the robots energy budget is depleted. The robot then follows this reference point in a stable manner, as discussed in Chapter 6.

Basic methods and tools used

Herein we give a brief description of the main methods and practices adopted during the development of **FW1** and **FW2**. Our development is divided into three main parts: theoretical development, simulation investigation and validation, and experimental validation.

In the theoretical development phase, we design our ES frameworks and provide proofs of validity and establish theoretical limits on performance, and this is manifested in various lemmas, propositions and theorems in Chapters 4, 5 and 6.

In the simulation validation phase, the main simulation tool we use is the ARGoS simulator [117], and we mostly use a KheperaIV robot model in simulation as a proof of concept. We use energy and voltage models for the robots similar to what is being used during the development of **FW1** and **FW2**. The various frameworks are implemented on the robots using the Buzz programming language [118] which facilitates interacting with multiple robots in a multi-robot system programming-wise. In the development of **FW1** and **FW2**, we run various simulation campaigns (using the described simulation setup) of batches of 30 simulations or more for each scenario under study. We run these simulation campaigns on computecanada clusters.

Following the simulation phase, we go through an experimental validation phase where we test the validity of our proposed methods on actual real-life robots to test the extent of efficacy of our methods. Throughout the development in this thesis, we use three main experimental setups (we provide more details about these setups in later Chapers)

- In Chapter 4, we carry out experiments using a set of three KheperaIV [119] robots in an indoor lab environment, and we use an indoor Optitrack [120] motion capture system for indoor localization. The robot runs Buzz codes using BzzKh4 [121], a modified version of Buzz for Khepera robots.
- In Chapter 5, we carry out experiments with a set of three Cognifly [122] quadcopters. We use the Optitrack system for indoor localization as well. The drones run BuzzCognifly [123], which is a Buzz implementation adapted for Cognifly drones.
- In Chapter 6, we use one AgileX Scout Mini rover, running an exploration software stack, and equipped with a set of sensors that enable the robot of doing frontier exploration tasks. We give more details about the system structure in Chapter 6. Moreover, we run the simulations on a docker container that runs an instance of ARGoS simulator, with the exploration stack and simulated sensors running in the background.

Document layout

Herein, we describe briefly the layout of the next chapters. In Chapter 4, we present the development of the basic version of **FW1** using a simplified battery discharge model, and introduce the idea of CS capacity. We further develop this treatment in Chapter 5, where we use a more accurate battery model and give a more developed treatment of CS capacity, as well as demonstrate the

results of experimental trials using Cognifly quadcopters [122]. In Chapter 6 we develop **FW2**, where we endow a generic path planner with ES property in a modular and energy-agnostic fashion, and we demonstrate the results of experimental runs using AgileX Scout Mini rover [124]. In Chapter 7, we provide a discussion of the work presented in the thesis, where we highlight some of the strong points of the work and give some remarks about some points of relevance. Moreover, we discuss some of the issues we attempted to tackle during the course of the development of **FW1** and **FW2**. Chapter 8 concludes the thesis and provides insights about current limitations and possible directions for future work.

CHAPTER 4 ARTICLE 1 : ENERGY AUTONOMY FOR RESOURCE-CONSTRAINED MULTI ROBOT MISSIONS

Preface: Robots in a multi-robot system need the ability to consistently recharge to achieve reliable long-term autonomy. In cases where robots share a charging station, they need to coordinate access to the station to avoid queuing and reduce degradation in mission quality. In this chapter, we tackle the energy sufficiency problem of a group of robots sharing one charging station. We propose a mission agnostic framework based on Control Barrier Functions (CBFs) that achieves two goals: 1. ensuring any robot's voltage stays above a certain desired threshold. 2. and ensuring mutually exclusive use of the charging station.
Another idea we study is the relationship between the number of serviceable robots, that carry out the coordination framework we propose, by one charging station and robots' discharge and recharge properties, as well as the sequence of arrival to the station. We believe that understanding this relationship is important for sizing charging demands for multi-robot systems.

Full Citation: Fouad, Hassan, and Giovanni Beltrame. "Energy autonomy for resource-constrained multi robot missions." 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020. Manuscript accepted for publication on July 2, 2020.

DOI: https://doi.org/10.1109/IROS45743.2020.9341455

Abstract: One of the key factors for extended autonomy and resilience of multi-robot systems, especially when robots operate on batteries, is their ability to maintain energy sufficiency by recharging when needed. In situations with limited access to charging facilities, robots need to be able to share and coordinate recharging activities, with guarantees that no robot will run out of energy. In this work, we present an approach based on Control Barrier Functions (CBFs) to enforce both energy sufficiency (assuring that no robot runs out of battery) and coordination constraints (guaranteeing mutual exclusive use of an available charging station), all in a mission agnostic fashion. Moreover, we investigate the system capacity in terms of the relation between feasible requirements of charging cycles and individual robot properties. We show simulation results, using a physics-based simulator and real robot experiments to demonstrate the effectiveness of the proposed approach.

4.1 Introduction

Long term autonomy is considered one of the key ingredients for the practical application of multi-robot system. When performing missions out of the comfort of the lab, limited battery capacity and the recharging ability of robots are one of the most important obstacles to deployment.

Many approaches can be found in literature that deal with this issue. Early efforts took directions as energy aware path planning [27] and node scheduling in wireless sensor networks. Later ideas have been integrated in multi-robot systems, as in [125] in which a mission was split in real time among participating agents depending on their energy level.

Another solution for energy maintenance is through the use of charging stations, whether being static or mobile. In [52], a group of charging robots plans routes to deposit batteries on predefined paths for robots doing surveillance so as to eliminate detours and assure energy sufficiency.

Notomista et al. [44] propose using static charging stations and a control barrier function (CBF) framework to assure energy persistence in a group of robots. This framework provides a constraint based behavioral layer that guarantees the survivability of robots by driving each robot to a dedicated charging station in a minimally invasive way (affecting their original mission as little as possible).

In this paper, we extend [44] by considering a group of robots doing a mission (e.g. coverage or waypoint navigation), but having a single charging station that they need to share. The contribution of this paper is twofold: 1) Augmenting the results in [44] with a CBF-based coordination framework that assures mutually exclusive use of the charging station, and 2) introducing some sufficient conditions that describe the system's capacity and assure the overall feasibility of the coordination.

4.2 Preliminaries

4.2.1 Control Barrier Functions (CBF)

A control barrier function (CBF) [47] is a tool that can be used to assure safety of control systems. In this context, safety means guaranteeing that the states of the system stay in a "safe set" and never wander off to "unsafe" regions.

The safe set is defined to be the superlevel set of a continuously differentiable function $h(x)$ such that [47]:

$$\begin{aligned} \mathcal{C} &= \{x \in \mathbb{R}^n : h(x) \geq 0\} \\ \partial\mathcal{C} &= \{x \in \mathbb{R}^n : h(x) = 0\} \\ Int(\mathcal{C}) &= \{x \in \mathbb{R}^n : h(x) > 0\} \end{aligned} \tag{4.1}$$

which means that assuring that $h(x) > 0, \forall t > t_0$ implies the safe set $\mathcal{C}$ is positively invariant and the system is safe. For a control affine system of the form

$$\dot{x} = f(x) + g(x)u$$

having a control action u that achieves

$$\underbrace{L_f h(x) + L_g h(x) u}_{\dot{h}(x)} \geq -\alpha(h(x)) \tag{4.2}$$

where $\alpha(h(x))$ is an extended type $\mathcal{K}$ function, assures that $\mathcal{C}$ is positively invariant. In this paper we use zeroing control barrier functions (ZCBF) [73] owing to their robustness and asymptotic stability properties [72].

Definition 1. *[73] For a region $\mathcal{D} \subset \mathcal{C}$ a continuously differentiable function $h(x)$ is called a zeroing control barrier function (ZCBF) if there exists an extended class $\mathcal{K}$ function $\alpha(h(x))$ such that*

$$\sup_{u \in U} \left(L_f h(x) + L_g h(x) u + \alpha(h(x))\right) \geq 0 \quad \forall x \in \mathcal{D} \tag{4.3}$$

We can also define the set K_{zcbf} [73] for the ZCBF $h(x)$ as

$$K_{zcbf} = \{u \in U : L_f h(x) + L_g h(x) u + \alpha(h(x)) \geq 0\} \tag{4.4}$$

which is the set containing all the "safe" control inputs. Choosing a Lipschitz continuous controller u from K_{zcbf} is sufficient to ensure that the safe set $\mathcal{C}$ is forward invariant [73].

Equation (4.2) represents a basic requirement on the control action to assure safety, but this control action is not necessarily that of an arbitrary mission. Quadratic Programming (QP) can be used to enforce (4.2) as a constraint that has to be respected by the mission's nominal control action u_{nom}

in the following manner [44]:

$$\begin{aligned} u^* = \min_u \quad & ||u - u_{nom}|| \\ \text{s.t.} \quad & L_f h(x) + L_g h(x) u \geq -\alpha(h(x)) \end{aligned} \tag{4.5}$$

4.2.2 Problem statement

Given a group of n robots, we:

- Ensure that the battery voltage of each robot never goes below a certain desired level (safety).
- Ensure that there is no more than one robot at the charging station at any time (coordination).
- Investigate the number of robots that can be accommodated by one charging station (capacity).

For the sake of simplicity of analysis, we use a robot model similar to [44]:

$$\begin{aligned} \dot{x} &= u \\ \dot{E} &= \begin{cases} -k_e, & ||x - x_c|| > \delta \\ k_{ch}, & \text{otherwise} \end{cases} \end{aligned} \tag{4.6}$$

where $x \in \mathbb{R}^2$ is the robot's position, $x_c \in \mathbb{R}^2$ is the position of the charging station, E is the battery voltage, $k_e > 0$ and $k_{ch} > 0$ are discharging and charging coefficients respectively and δ is an effective charging distance away from the charging station. To be conservative, we take k_e as the worst case discharge rate of the battery. The charging model in (4.6) is a linear approximation of a continuous charging process that can be obtained from wireless pads [126].

4.2.3 Overview of the strategy

In Figure 4.1 we present a schematic representation of our proposed framework. We build an energy sufficiency layer that ensures that the voltage of each robot stays higher than a given minimum level (to ensure sufficiency) and lower than a certain upper bound (by ejecting the robot from the charging region to avoid overcharging). We also implement a coordination layer that ensures mutually exclusive use of the charging station through proper manipulation of the robots' desired voltage thresholds, thus manipulate their arrival times to the charging station to respect a desired temporal separation δ_t. Moreover, we describe the system's capacity, namely the relationship between the battery discharge characteristics and the number of robots that can be served by a single charging station, as well as allowable values of δ_t, to ensure the overall feasibility of the system.

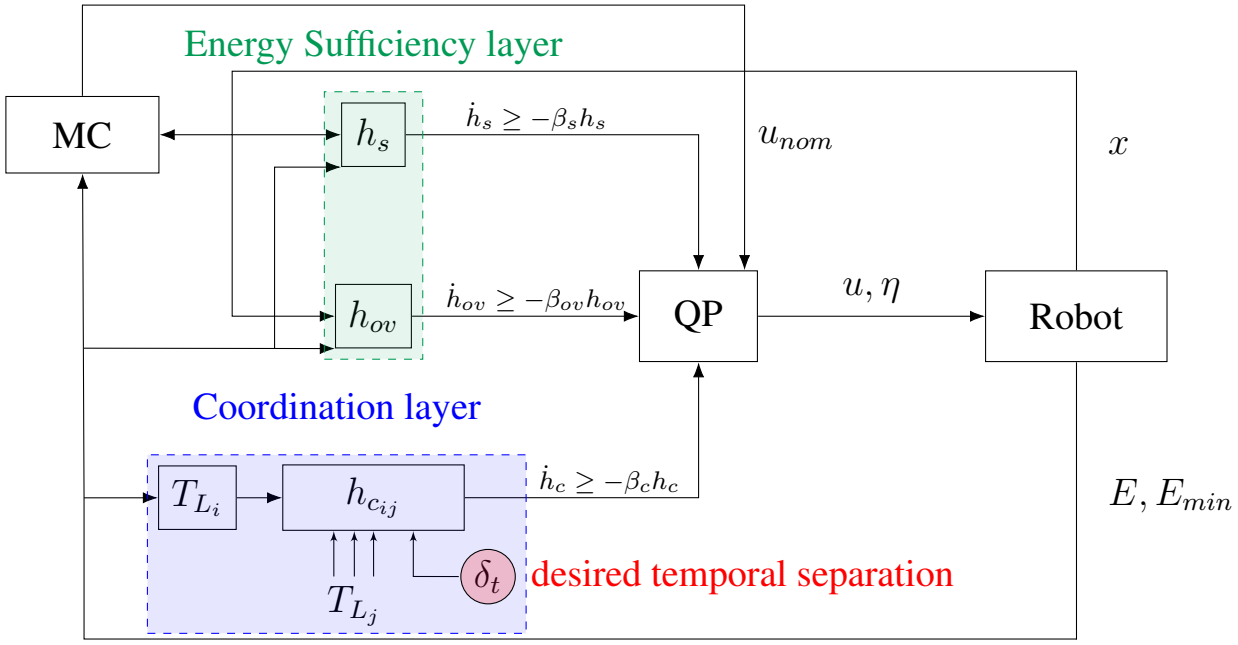


Figure 4.1 A schematic of the proposed framework. Both ES layer and coordination layer consist of CBFs that produce inequality constraints for the Quadratic Program (QP), that modifies a nominal control action u_{nom} coming from a Mission Controller (MC). The coordination layer aims to separate consecutive arrival times to the CS by at least an amount of δ_t.

4.3 Energy sufficiency framework

4.3.1 Energy sufficiency CBF

The candidate CBF we use for energy sufficiency is taken from [44] and is defined as

$$h_s = E - E_{min} - k_c \log \frac{||x - x_c||}{\delta} \tag{4.7}$$

where E_{min} is the desired minimum voltage, δ is the effective radius of the charging station and $k_c > 0$ is a constant such that the last expression approximates the amount of voltage needed to return back to the charging station.

Theorem 1 and 2 from [44] show that the CBF in (4.7) is a ZCBF if $||x - x_c|| > \delta$ and that $u \in U = \mathbb{R}^2$ (no restrictions on the control action moving the robots). We investigate the relation between the choice of the $\alpha(h)$ in (4.2) and the voltage level at the arrival at the charging station. In the following, we use a linear $\alpha(h)$ function of the form

$$\alpha(h) = \beta h \tag{4.8}$$

Lemma 1. *Given a robot with dynamics described in (4.6), and for the the QP in (4.5) using the ZCBF in (4.7), the voltage difference $E - E_{min}$ is bounded by zero from below and by a quantity inversely proportional to β from above.*

Proof. After some time $T > t_0$, the nominal control input u_{nom} won't be able to satisfy the constraint in (4.5), in which case the output of the QP problem will be a control input u that satisfies

$$\dot{h}_s = -\beta h_s \Rightarrow h_s(t) = h_s(T)e^{-\beta(t-T)} \tag{4.9}$$

but from (4.7)

$$E - E_{min} - k_c \log \frac{||x - x_c||}{\delta} = h_s(T)e^{-\beta(t-T)} \tag{4.10}$$

and on arrival at the charging station at time $t = t_a$ we have $||x - x_c|| = \delta$ thus

$$E(t_a) - E_{min} = h_s(T)e^{-\beta(t_a-T)} \tag{4.11}$$

to show that $E(t_a) - E_{min}$ is bounded from below, it suffices to mention that if the robot starts in the safe set (i.e. $h_s(t_0) \geq 0, \forall t \geq t_0$), then respecting the constraint in (4.5) for h_s assures that $h_s(T) \geq 0$ as well (by virtue of the fact that h_s is a ZCBF [44]), so the right hand side in (4.11) is not smaller than zero.

$E(t_a) - E_{min}$ is bounded from above by equation(4.11), which shows that the bound on this difference decays exponentially with β. □

Remark 1. *It is worth noting that Lemma 1 shows that the ZCBF h_s has a tracking property in the sense that the voltage at the time of arrival at the charging station is close to E_{min} with a margin, the tightness of which can be manipulated.*

Since the battery discharge is constant with time and that the difference $E - E_{min}$ is bounded (i.e. by proper choice of β the robot arrives at charging station with $E \approx E_{min}$), we can conclude that the arrival time at the charging station is approximately the time it takes the battery to discharge from its current voltage to E_{min}

$$T_L \approx \frac{E - E_{min}}{k_e} \tag{4.12}$$

with T_L being the arrival time at the charging station.

To recharge the battery, the robot needs to stay for a sufficient time inside the charging region:

Lemma 2. *Given a robot with dynamics described in (4.6), and for the the QP in (4.5) using the ZCBF in (4.7), choosing the $\alpha(h_s)$ function*

$$\alpha(h_s) = \begin{cases} \beta_h h_s & , ||x - x_c|| > \delta \\ \beta_l h_s & , \textit{otherwise} \end{cases} \tag{4.13}$$

with $0 < \beta_l \ll \beta_h$, leads to having $D \to \infty$ as $t \to \infty$ inside the charging region, where $D = ||x - x_c||$.

Proof. When a robot heads back to the charging station at $t \geq T$, the control action fulfilling (4.9) is

$$L_f h_s + L_g h_s u = -\beta h_s \Rightarrow -k_e - \frac{k_c}{D^2}(x - x_c)^T u = -\beta h_s$$

This equation can be written as

$$\begin{aligned} (x - x_c)^T u &= \frac{\beta h_s - k_e}{k_c} \overbrace{(x - x_c)^T (x - x_c)}^{D^2} \\ &\Rightarrow u = \frac{\beta h_s - k_e}{k_c}(x - x_c) \end{aligned} \tag{4.14}$$

plugging this control action into the robot dynamics, and noting that $\frac{d}{dt} D = \frac{(x - x_c)}{||x - x_c||}\dot{x}$, we have

$$\dot{D} = \frac{\beta h_s - k_e}{k_c} D \tag{4.15}$$

Inside the charging region, $\beta = \beta_l$ and $E - E_{min} = \Delta E_a + k_{ch}\Delta t$, with $\Delta E_a = E(t_a) - E_{min}$ and $\Delta t = t - t_a$, thus

$$\dot{D} = \frac{1}{k_c}\left(\beta_l \left(\Delta E_a + k_{ch}\Delta t - k_c \log \frac{D}{\delta}\right) - k_e\right) D \tag{4.16}$$

then if we substitute $\beta_l = 0$, which is the lowest value of β, in (4.16) we get

$$\dot{D} = -\frac{k_e}{k_c} D \Rightarrow D(t) = \delta e^{-\frac{k_e}{k_c}\Delta t} \tag{4.17}$$

which means that $D \to 0$ as $t \to \infty$. □

Lemma 3. *For the same conditions of Lemma 2, the choice of (4.13) assures that the robot stays inside the charging region for a sufficiently long period.*

Proof. When the robot approaches the charging station, $\dot{D} < 0$. However, there is a point at which $\dot{D} = 0$ as the battery recharges and starts to move out of the charging region. The time until this

reversal point can be calculated by setting $\dot{D} = 0$ in the above equation so

$$\Delta t_r = \frac{1}{k_{ch}} \left(\frac{k_e}{\beta_l} + k_c \log \frac{D_r}{\delta} - \Delta E_a \right) \tag{4.18}$$

where D_r is the distance from x_c at the reversal time. Since $D \to 0$ as $t \to \infty$ this means that that $D_r \neq 0$ in finite time so $\log \frac{D}{\delta} > -\infty$ and thus it can be seen that the reversal time can be set arbitrarily high by setting a low value for β_l, which in turn means that using the proposed switching in the value of β, the robot can stay for an arbitrary amount of time inside the charging region. □

4.3.2 Overcharge protection CBF

The purpose of this CBF is to ensure that the robot "escapes" the charging region before it overcharges (i.e. keeps charging beyond a desired maximum voltage) in a similar way to what was done in [45]. To this end we propose:

$$h_{ov} = E_{max} - E + k_{ov} \log \frac{D}{\delta} \tag{4.19}$$

The main intuition behind this choice is that when $E_{max} - E$ decreases as the robot recharges, the $k_{ov} \log \frac{D}{\delta}$ tends to become more positive (or rather less negative since the robot is inside the charging region during recharging) by escaping away from the charging region. In the following, we show that this proposed CBF is indeed a ZCBF, in a very similar way to Theorem 1 in [44].

Theorem 1. *The function $h_{ov} = E_{max} - E + k_{ov} \log \frac{D}{\delta}$ is a ZCBF if $U = \mathbb{R}^m$*

Proof. Since we are considering the case where $U = \mathbb{R}^m$ (i.e. no saturation on the control action), then showing that h_{ov} is a ZCBF follows from showing that $L_g h_{ov} \neq 0$. For the proposed function h_{ov}

$$L_g h_{ov} = \frac{k_{ov}}{D^2} (x - x_c)^T \tag{4.20}$$

so we need $||x - x_c|| > 0$ to have $L_g h_{ov}$ defined and not equal to zero which was shown in Lemma 2, and since recharging from E_{min} to E_{max} happens in a finite time, then $x \neq x_c$ in included in this period. □

Remark 2. *The CBFs for energy sufficiency and overcharge protection may be in conflict since one slows down the robot for it to recharge (i.e. h_s), while the other tries to push it outside the charging region (i.e. h_{ov}). To avoid infeasibility of the QP, the next two conditions have to be satisfied*

$$-k_e - \frac{k_c}{D^2} (x - x_c)^T u \geq -\beta h_c \tag{4.21a}$$

$$-k_{ch} + \frac{k_{ov}}{D^2}(x - x_c)^T u \geq -\beta_{ov} h_{ov} \tag{4.21b}$$

and eliminating u *we get*

$$\frac{\beta h_s - k_e}{k_c} - \frac{k_{ch} - \beta_{ov} h_{ov}}{k_{ov}} \geq 0, \tag{4.22}$$

so condition (4.21b) can be relaxed when the left hand side of (4.22) is equal to zero, leading h_{ov} *to take over and push the robot out of the charging region.*

4.4 Coordination framework

The purpose of the coordination framework is to ensure mutually exclusive use of the available charging station. The proposed strategy is through the manipulation of E_{min} for each robot to change their arrival times at the charging station (according to (4.12)). Our main assumptions are:

1. The underlying communication graph between the robots is a complete graph, meaning that each robot can receive information from all other robots.
2. Robots are homogeneous and have the same battery discharge rate.
3. All robots start at the maximum voltage.
4. The recharge rate of the battery is faster than the discharge rate. This can be a reasonable assumption for systems with powerful wireless charging pads along with capable lipo batteries or for battery swapping platforms[1].

The mechanism we propose to change the value of E_{min} is to assume that it changes according to the following dynamics

$$\dot{E}_{min} = \eta \tag{4.23}$$

where $\eta \in \Theta = \mathbb{R}$ with Θ being the set of all admissible values of η, and the nominal value of choice for η is zero (its default value is zero unless changed by other control laws to fulfill other constraints). Our main strategy is twofold: 1) Introduce a constraint to bound the value of E_{min} from below, as well as conditions that assures the feasibility of the scheduling with respect to the system's capacity, and 2) introduce a CBF $h_{c_{ij}}$ that keeps the difference in arrival times above a desired value δ_t through the manipulation of E_{min}.

[1] Although battery swapping is discrete process in nature, it can fit in our proposed framework by properly choosing k_{ch} so as to ensure all robots spend at least the amount of time needed for a battery swap process.

4.4.1 Bounds on E_{min}

Arbitrary manipulation of E_{min} does not necessarily comply with what could be physical bounds on its value. Asking for a too low value may cause permanent damage to the battery and a too high value cause h_s to be negative.

Lower bound on E_{min}

Lower bounding E_{min} corresponds to requiring having an acceptable voltage at the beginning of the charging process to avoid permanent damage to the battery. The proposed CBF for this purpose is

$$h_L = k_p(E_{min} - E_{lb}) \tag{4.24}$$

where $E_{lb} > 0$ is the desired lower bound voltage and $k_p > 0$ is a scaling gain. The constraint for the QP (4.5) is:

$$k_p\eta \geq -\alpha(h_L) \tag{4.25}$$

and we choose

$$\alpha(h_L) = \kappa \cdot \text{sign}(h_L) \cdot |h_L|^\rho \quad , \rho \in [0, 1) \tag{4.26}$$

with $\kappa > 0$.

Lemma 4. *For a robot with dynamics (4.6) and E_{min} obeying (4.23), h_L is a ZCBF.*

Proof. It is not hard to see that h_L is a ZCBF following the same argument in Theorem 2, as $\eta \in \Theta = \mathbb{R}$, so it is always the case that a value of η could be found that satisfies (4.3) for the constraint (4.25), so if $E_{min}(t_0) \geq 0$ then (4.25) ensures positive invariance of the safe set $\mathcal{C}_{E_{min}} = \{E_{min} \in \mathbb{R} : E_{min} \geq E_{lb}\}$. □

Upper bound on E_{min}

The upper bound on E_{min} is correlated with the capacity of the system and how many robots can be served by one charging station. Indeed, there is a relation between the feasible number of robots (that can be served with a minimum separation time δ_t) and certain robot parameters, like discharge and recharge rates and maximum and minimum voltage of the battery.

To demonstrate this, suppose we have n robots in the first recharging cycle (when all robots start at E_{max} and discharge with the same rate, as per assumptions 2 and 3) and that each robot has a specific E_{min}. We consider the first robot to recharge (with E_{min} yet to be determined) and the last one (with $E_{min} = E_{lb}$ in the extreme case). We require that once the first robot arrives at time t_1

and recharges at t_2, it won't recharge again until at least a time δ_t after the last robot has recharged. Thus at t_2

$$\frac{E_{max} - \bar{E}_m}{k_e} - \frac{(\bar{E}_m - k_e(t_2 - t_1)) - E_{lb}}{k_e} \geq \delta_t \tag{4.27}$$

where $t_2 - t_1 = \frac{E_{max} - \bar{E}_m}{k_{ch}}$ is the time needed to recharge the battery of the first robot, and $\bar{E}_m$ is the value of E_{min} of the first agent. The first expression in the above inequality is the time the first robot takes until it reaches the charging station again, and the second is the time the last agent take to reach the charging station for the first time starting from t_2. Substituting $t_2 - t_1$ in the above inequality:

$$\bar{E}_m \leq \frac{(1 + \frac{k_e}{k_{ch}})E_{max} + E_{lb} - \delta_t k_e}{2 + \frac{k_e}{k_{ch}}} \tag{4.28}$$

Noticing that the difference in arrival times of any two robots is $\Delta T_L = \frac{\Delta E - \Delta E_{min}}{ke}$ and considering the first recharging cycle where E is the same for all robots who have not recharged yet, this means that ΔE_{min} sets the difference in arrival times and uniformly separated arrival times imply uniform separation in values of E_{min}.

We can calculate the uniform step in E_{min} if we convert the last inequality to an equality and using

$$\Delta \bar{E}_m = \frac{\bar{E}_m - E_{lb}}{n - 1} = \frac{(1 + \frac{k_e}{k_{ch}})(E_{max} - E_{lb}) - \delta_t k_e}{(2 + \frac{k_e}{k_{ch}})(n - 1)} \tag{4.29}$$

What we require in this case is that the arrival times of the last two robots (without loss of generality) with $E_{min} = E_{lb}$ and $E_{min} = E_{lb} + \Delta E_{min}$ to be at least δ_t:

$$\frac{E_{max} - E_{lb}}{k_e} - \frac{E_{max} - (E_{lb} + \Delta E_{min})}{k_e} \geq \delta_t \tag{4.30}$$

and substituting (4.29) into the last equation we get

$$\frac{(1 + \frac{k_e}{k_{ch}})(E_{max} - E_{lb}) - \delta_t k_e}{(2 + \frac{k_e}{k_{ch}})(n - 1)} - \delta_t k_e \geq 0, \tag{4.31}$$

the critical value of δ_t at which equality is achieved, given system parameters ($k_e, k_{ch}, n, E_{max} - E_{min}$) is

$$\delta_{t_{cr}} = \frac{\left(1 + \frac{k_e}{k_{ch}}\right)(E_{max} - E_{lb})}{k_e \left[1 + \left(2 + \frac{k_e}{k_{ch}}\right)(n - 1)\right]}. \tag{4.32}$$

The last relation describes the feasible separation in arrival times for the robots given different system parameters, and considering that each robot has a distinct value of E_{min}, which are separated by multiples of $\bar{E}_m$.

Finally, we require $\delta_{t_{cr}}$ above to be more than the time taken to fully recharge a battery from E_{lb} to E_{max}

$$\delta_{t_{cr}} \geq \frac{E_{max} - E_{lb}}{k_{ch}}. \tag{4.33}$$

Definition 2. *For a group of n robots, each with a distinct fixed value of E_{min} and all applying the energy sufficiency and overcharge CBFs constraints (h_s and h_{ov}), a* charging cycle *is defined as the time window taken by the robot with the lowest E_{min} value (i.e. $E_{min} = E_{lb}$) to discharge from E_{max} to E_{lb} and then recharges again.*

Lemma 5. *For a group of n robots each with a distinct value of E_{min} that satisfies (4.32), (4.33) and (4.29), let z_i be the number of recharges that one robot can have in one charging cycle, then the maximum number of recharges for any robot is $\bar{z}_i = 2$.*

Proof. The number of recharges of robot $i = n$ (first robot to recharge) in one cycle is

$$z_n = 1 + \left\lfloor \frac{\frac{(E_{max} - E_{lb})\left(1 + \frac{k_e}{k_c h}\right)}{k_e}}{\frac{(E_{max} - \bar{E}_m)\left(1 + \frac{k_e}{k_c h}\right)}{k_e}} \right\rfloor \tag{4.34}$$

where the second expression on the right hand side is the floor of the quotient of the two periods. If we take this quotient and substitute (4.28) and (4.32) we get

$$\frac{\left(2 + \frac{k_e}{k_{ch}}\right) A}{A + \left(2 + \frac{k_e}{k_{ch}}\right)} = Q \tag{4.35}$$

where $A = \left(1 + \left(2 + \frac{k_e}{k_{ch}}\right)(n-1)\right)$. What we want to verify is that $1 < Q < 2$. Checking the difference between numerator and denominator

$$\left(2 + \frac{k_e}{k_{ch}}\right) A - A - \left(1 + \frac{k_e}{k_{ch}}\right) = (A-1)\left(1 + \frac{k_e}{k_{ch}}\right) > 0, \tag{4.36}$$

satisfying the first inequality. To check the second we need to make sure the numerator is less than twice the denominator

$$\begin{aligned} &2\left(A + \left(1 + \frac{k_e}{k_{ch}}\right)\right) - 2A - \frac{k_e}{k_{ch}} A \\ &= 2\left(1 + \frac{k_e}{k_{ch}}\right) - \frac{k_e}{k_{ch}}\left(1 + \left(2 + \frac{k_e}{k_{ch}}\right)(n-1)\right) \end{aligned} \tag{4.37}$$

but if we substitute (4.32) in (4.33) we have

$$\left(1+\frac{k_e}{k_{ch}}\right) \geq \frac{k_e}{k_{ch}}\left(1+\left(2+\frac{k_e}{k_{ch}}\right)(n-1)\right) \tag{4.38}$$

which renders (4.37) positive, meaning that the ratio is upper bounded by 2, which in turn means that the maximum number of recharges of the most needy agent is two per charging cycle. Since the last robot recharges only once in a cycle, this means that any other robot in between can recharge no more than twice per cycle, which completes the proof. □

Lemma 6. *For a group of n robots, if δ_t satisfies*

$$\frac{E_{max}-E_{lb}}{k_{ch}} \leq \delta_t \leq \delta_{t_{cr}} \tag{4.39}$$

as well as equation (4.29), then there exists $\mathbf{E}_m = \{E_{min_1}, \ldots, E_{min_n}\}$ such that the difference in arrival times between any two robots is at least δ_t (i.e. the scheduling problem is feasible).

Proof. The idea of the proof is to use the upper bound $\bar{z}_i$ and show that the possible difference between any two landing times is at least $\delta_{t_{cr}}$ even with this worst case scenario.

Based on Lemma 5, the possible number of recharges is $2(n-1)$ and thus the required number of spaces between these recharging events (taking the start and end recharging events of last agent) is $M = 2(n-1)+1 = 2n-1$. If we divide the length of a whole charging cycle by this quantity it gives the available time δ_{av} between any two recharging events in the worst case, which should be at least equal to $\delta_{t_{cr}}$. To check this

$$\delta_{av} - \delta_{t_{cr}} = \frac{(E_{max}-E_{lb})\left(1+\frac{k_e}{k_{ch}}\right)}{k_e}\left[\frac{1}{2n-1}-\frac{1}{A}\right] \tag{4.40}$$

and the difference of the numerator is $\frac{k_e}{k_{ch}}(n-1) > 0$, which in turn means that for the $2n-1$ intervals, each can be at least $\delta_{t_{cr}}$. This means that there exist values of E_{min_i} for each of these landings that are properly temporally separated.

To complete the proof, we consider two corner cases for E_i at the beginning of each cycle, to ensure the robots are able to adopt new E_{min} values leading to separate landings. The worst case is

$E_i = E_{min_i}$ at the beginning of each cycle. In this case, setting $E_{min_i} = E_{lb}$ gives

$$\begin{aligned} T_{L_2} &= \frac{E_{lb} + \Delta\bar{E}_m - E_{lb}}{k_e} = \delta_{t_{cr}} \\ T_{L_3} &= \frac{E_{lb} + 2\Delta\bar{E}_m - E_{lb}}{k_e} = 2\delta_{t_{cr}} \\ &\vdots \end{aligned} \tag{4.41}$$

the second corner case is if $E_i = E_{max}$ at the beginning of the cycle, which has a solution by design, as value of E_{min_i} separated apart by at least $\Delta\bar{E}_m$ assures having $\delta_{t_{cr}}$ between arrival times by design. □

4.4.2 Coordination CBF

This CBF aims at separating the arrival times of two robots with at least δ_t, and the core idea is very similar to the collision avoidance strategy proposed in [46], albeit it is collision of arrival times. For that we define a pairwise safe set $\mathcal{C}_{ij}$ as

$$\mathcal{C}_{ij} = \{(E_{min_i}, E_{min_j}) \in \mathbb{R}^2 | h_{c_{ij}} \geq 0\} \tag{4.42}$$

and we propose the following CBF $h_{c_{ij}}$ between two agents (i, j)

$$h_{c_{ij}} = \log \frac{|T_{L_i} - T_{L_j}|}{\delta_t} \tag{4.43}$$

where T_{L_i} is the arrival time of robot i as described in (4.11), and δ_t is the desired separation in arrival times between robots. The resulting constraint for the QP problem is

$$\frac{T_{L_i} - T_{L_j}}{|T_{L_i} - T_{L_j}|^2} \frac{\Gamma_i - \Gamma_j}{ke} \geq -\alpha(h_{c_{ij}}) \tag{4.44}$$

where $\Gamma_i = \frac{d}{dt} T_{L_i} = -k_{e_i} - \eta_i$.

It would be more practical to consider a decentralized version of equation (4.44) and to show that (4.43) is a ZCBF. The desired decentralization can be done by dropping the term η_j from Γ_j in (4.44) so we end up with (noticing that robots are assumed to have the same discharge rate)

$$-\frac{T_{L_i} - T_{L_j}}{k_e |T_{L_i} - T_{L_j}|^2} \eta_i \geq -\alpha(h_{c_{ij}}). \tag{4.45}$$

For the $\alpha(h_{c_{ij}})$ function in (4.45) we propose the form

$$\alpha(h_{c_{ij}}) = \gamma_{ij} \cdot \text{sign}(h_{c_{ij}}) \cdot |h_{c_{ij}}|^{\rho} \quad , \rho \in [0, 1) \,. \tag{4.46}$$

We define the value of γ_i for robot i as

$$\gamma_{ij} = \begin{cases} \gamma_h & \text{, if } ||x_i - x_c|| > \delta \text{ and } ||x_j - x_c|| > \delta \\ 0 & \text{, otherwise} \end{cases} \tag{4.47}$$

Theorem 2. *For a multi robot system with (4.6) as the dynamics of each robot, then for a pair of robots (i, j) satisfying $||x_i - x_c|| > \delta$ and $||x_j - x_c|| > \delta$, $h_{c_{ij}}$ is a ZCBF for $\eta \in \Theta = \mathbb{R}$, with (4.45) rendering the set $\mathcal{C}_{ij}$ forward invariant. Moreover, if $(E_{min_i}(t_0), E_{min_j}(t_0)) \notin \mathcal{C}_{ij}$, (4.45) leads $(E_{min_i}(t), E_{min_j}(t))$ to converge to $\partial\mathcal{C}_{ij}$ in finite time.*

Proof. Since $\eta \in \Theta = \mathbb{R}$ there exists a control action η that satisfies (4.3) (and keeps $\mathcal{C}_{ij}$ invariant), then to show that $h_{c_{ij}}$ is a ZCBF, we need to make sure that $|T_{L_i} - T_{L_j}| \neq 0$.

The only chance that this difference can be equal to zero is when one of the robots enters to the charging region. To show this, consider having two robots (i, j) applying (4.45) and without loss of generality suppose that robot j arrives at the charging station, so the difference in arrival times is

$$\Delta T_{L_{ij}} = \frac{1}{k_e} \left[E_i - E_{min_i} - (E_j(t_a) + k_{ch}(t - t_a) - E_{min_j}) \right] \tag{4.48}$$

Due to the choice of γ_{ij} in (4.47) the right hand side of (4.45) is equal to zero, so the choice of $\eta_{nom_i} = \eta_{nom_j} = 0$ as nominal values actually satisfies (4.45) for both robots and consequently E_{min_i} and E_{min_j} does not change. This means that (4.48) is

$$\Delta T_{L_{ij}} = \frac{1}{k_e} \left[E_i - E_{min_i} - (\delta_E + k_{ch}(t - t_a)) \right] \tag{4.49}$$

where δ_E is the difference between the voltage and minimum voltage at the time of arrival to the charging station, expressed in (4.11). Therefore, it can be seen from (4.49) that at some point this difference in arrival times is equal to zero (E_i decreases while $k_{ch}(t - t_a)$ increases), rendering (4.44) undefined. We can conclude that (4.43) is a ZCBF when both robots are out of the charging region.

The proof of the second part is the same as that of proposition III.1 in [113] and is omitted here for space limits. □

It is worth mentioning that the main motivation in the choice of (4.46) is the idea that each robot can start with a random estimate of its E_{min_i} and generating a control action η_i that satisfies (4.45) ensures safety in finite time.

In addition, the above theorem is applicable even if one robot i applies (4.45) with respect to another robot j while robot j is not doing the same. This follows the same argument that since $\eta_i \in \Theta = \mathbb{R}$ then there is always a value of η_i that satisfies (4.3), as long as both robots are away from the charging station.

In the coordination strategy we propose, we decouple the energy sufficiency behaviour and the coordination behaviour. This should not affect the arguments stated earlier about the ability of the energy sufficiency ZCBF to track E_{min}. This is because (4.10) and (4.11) do not put a constraint on the change of E_{min} as long as u is generated in such a way that causes $\dot{h}_s = -\beta h$.

To show this, the constraint in (4.5) for h_s, assuming that E_{min} changes, is

$$-k_e - \eta - \frac{k_c}{D^2}(x - x_c)^T u \geq -\alpha(h_s) \tag{4.50}$$

and if we consider E_{min} not to change in this constraint the second term of the LHS of the previous inequality drops out

$$-k_e - \frac{k_c}{D^2}(x - x_c)^T u \geq -\alpha(h_s) \tag{4.51}$$

If we consider that $u \in U = \mathbb{R}^2$ (which was the case for Theorem 1 in [44] to show that h_s is a ZCBF), then it can be argued that there will be always u' to be substituted in (4.51) such that

$$\frac{k_c}{D^2}(x - x_c)^T u' = \frac{k_c}{D^2}(x - x_c)^T u + \eta \tag{4.52}$$

and this indicates the possibility to decouple the coordination from the energy sufficiency CBFs.

4.4.3 Feasibility of QP

Our coordination strategy introduces two barrier functions: $h_{c_{ij}}$ which tends to keep two agents' arrival times separate through changing E_{min}, and h_L which bounds E_{min} from below. We need to assure that these constraints are admissible and lead to a feasible QP.

The main problem lies in the fact that the coordination effort is being done by changing only E_{min} which is only 1-D. This way potentially conflicting constraints in the QP (4.5) may render it infeasible. This issue has been dealt with in the context of control barrier function composition in [78] and [46] and we adapt the basic idea from the latter.

The main idea is that for a local agent, instead of applying the coordination constraint with all other agents, it only has to apply it with the agent that has the closest arrival time among all other agents (hence the need for assumption 1). If during the process the value of E_{min} is about to go below the lower bound, the agent stops the coordination and focuses only on keeping the lower bound of E_{min}.

This way, we make sure that by construction each agent only changes E_{min} in one direction, leading to a feasible QP. This idea is summarized in Algorithm (1).

Algorithm 1 Coordination algorithm

Require: $T_{L_k} \quad , \forall k \in \mathcal{N}_i$
  $h_{c_{min}} = h_0$
  $h_{L_i} = E_{min_i} - E_{lb}$
  while j in $\mathcal{N}_i$ **do**
    $h_{c_{ij}} = \log \frac{T_{L_i} - T_{L_j}}{\delta}$
    if $h_{c_{ij}} < h_{c_{min}}$ **then**
      $h_{c_{min}} = h_{c_{ij}}$
    end if
  end while
  if $h_{c_{min}} < h_{L_i}$ **then**
    $A_c = L_g h_{c_{min}}$
    $B_c = -L_f h_{c_{min}} - \alpha(h_{c_{min}})$ ... (eqn. 4.45)
  else
    $A_c = L_g h_{L_i}$
    $B_c = -L_f h_{L_i} - \alpha(h_{L_i})$ ... (eqn. 4.25)
  end if

The final QP is

$$\begin{aligned} \mathbf{u}^* = \min_{\mathbf{u} \in \mathbb{R}^3} \quad & ||\mathbf{u} - \mathbf{u}_{nom}|| \\ \text{s.t.} \quad & A\mathbf{u} \geq B \end{aligned} \tag{4.53}$$

where

$$A = \begin{bmatrix} A_s^T \\ A_{ov}^T \\ A_c^T \end{bmatrix} = \begin{bmatrix} -\frac{k_c}{D^2}(x - x_c)^T & 0 \\ \frac{k_{ov}}{D^2}(x - x_c)^T & 0 \\ \begin{bmatrix} 0 & 0 \end{bmatrix} & A_c^T \end{bmatrix},$$

$$B = \begin{bmatrix} B_s \\ B_{ov} \\ B_c \end{bmatrix} = \begin{bmatrix} k_e - \alpha_s(h_s) \\ k_{ch} - \alpha_{ov}(h_{ov}) \\ B_c \end{bmatrix}$$

while A_c and B_c are determined from Algorithm (1). In the following, we show that Algorithm (1) indeed achieves the desired coordination task.

Theorem 3. *For a multi robot system with dynamics defined in (4.6) and with the coordination and lower bound constraints defined in (4.43) and (4.24), and provided that the inequalities (4.28) and (4.39) are satisfied, then Algorithm (1) assures that the difference in arrival time between any two robots is at least δ_t (mutual exclusive use is assured).*

Proof. From Algorithm (1), each robot is either applying the coordination CBF $h_{c_{ij}}$ or the lower bound CBF h_L. For the robots which do not apply h_L, from Theorem (2), for a robot i the control action that respects the constraint (4.44) leads E_{min_i} into safe set $\mathcal{C}_{ij}$ with respect to its neighbor with the closest landing time. Each robot can apply this to its neighbor with the closest landing time $\{(i,j)|j \in \mathcal{N}_i \text{ and } h_{c_{ij}} = \min_{k\in\mathcal{N}_i} h_{c_{ik}}\}$, eventually leading to $E_{min_i} \in \mathcal{C} = \bigcap_{\forall i \neq j} \mathcal{C}_{ij} \quad , \forall i$.

Moreover, since we have established the feasibility of the scheduling problem in Lemma (6), then we know that the sets $\mathcal{C}_{ij}$ are nonempty and that a solution exists.

If a robot i is applying the lower bound h_L, then it can't push its arrival time any further. In this case The nearest robot j that applies the coordination CBF will have a control action η_j that will lead E_{min_j} to $\mathcal{C}_{ij}$ (noticing that $\mathcal{C}_{ij}$ is non empty), and then all other robots applying coordination CBF will coordinate in a pairwise fashion based on the neighbour of closest landing time as discussed in the previous point.

□

4.5 Results

4.5.1 Simulation results

We carried out simulations using ARGoS [117], a physics-based simulator designed to handle multi-robot and swarm systems. The code was written using the Buzz programming language [127].

The mission considered for this simulation is a coverage mission, as in [44], in which 7 robots spread over a given area.

The main requirement is to cover a square of dimensions $6m \times 6m$ with a charging region of radius $0.2m$ around the origin. The robots are required to arrive at the charging station with a separation of $\delta_t = 20$ sec (knowing that it takes $\frac{E_{max}-E_{lb}}{k_{ch}} = 6$ sec to recharge). Table 4.1 contains the parameters of the system.

Table 4.1 Values of parameters used in simulation

Parameter	k_e	k_{ch}	n	E_{max}	E_{lb}	$\delta_{t_{cr}}$
Value	0.012	0.5	7	13.2	10	$20.77 > \delta_t$

Figure 4.2a shows the separation in arrival times is as desired. We note that the voltage does not go below E_{lb}. It can be seen as well that the maximum voltage is exceeded in a small number of conditions because of the kinematic model in the simulation, which was for a differential drive robot, so the rotation the robot experiences to pursue the point mass velocity introduces some delay that can cause such peaks.

4.5.2 Experimental results

We performed a simple waypoint navigation mission with three Khepera IV robots (where $u_{nom} = -k_p(x - x_{target})$), where $k_p > 0$ is a proportional gain). In our experiment, a virtual battery simulated in code was used instead of a real battery for the sake of proving the concept and probing the effects of more realistic operating conditions on the proposed algorithm. The code had a running frequency of 10 Hz, and an optical tracking system was used for position feedback. Figures 4.2c and 4.2d show the evolution of E and E_{min} with time.

4.6 Conclusions

In this paper we present a control barrier function (CBF) based framework for long term autonomy of multi robot systems with limited charging resources. We started by highlighting some tracking properties of the energy persistence CBF in [44] and then we introduced a CBF based framework to achieve the necessary coordination for sharing the charging station.

As a future work we consider extending the current results by investigating double integrator robot models with disturbances and examine the effect of such disturbances on the coordination behavior. Moreover, we would like to accommodate our approach to the case of having multiple charging stations and possibly relaxing the assumption of having a complete communication graph.

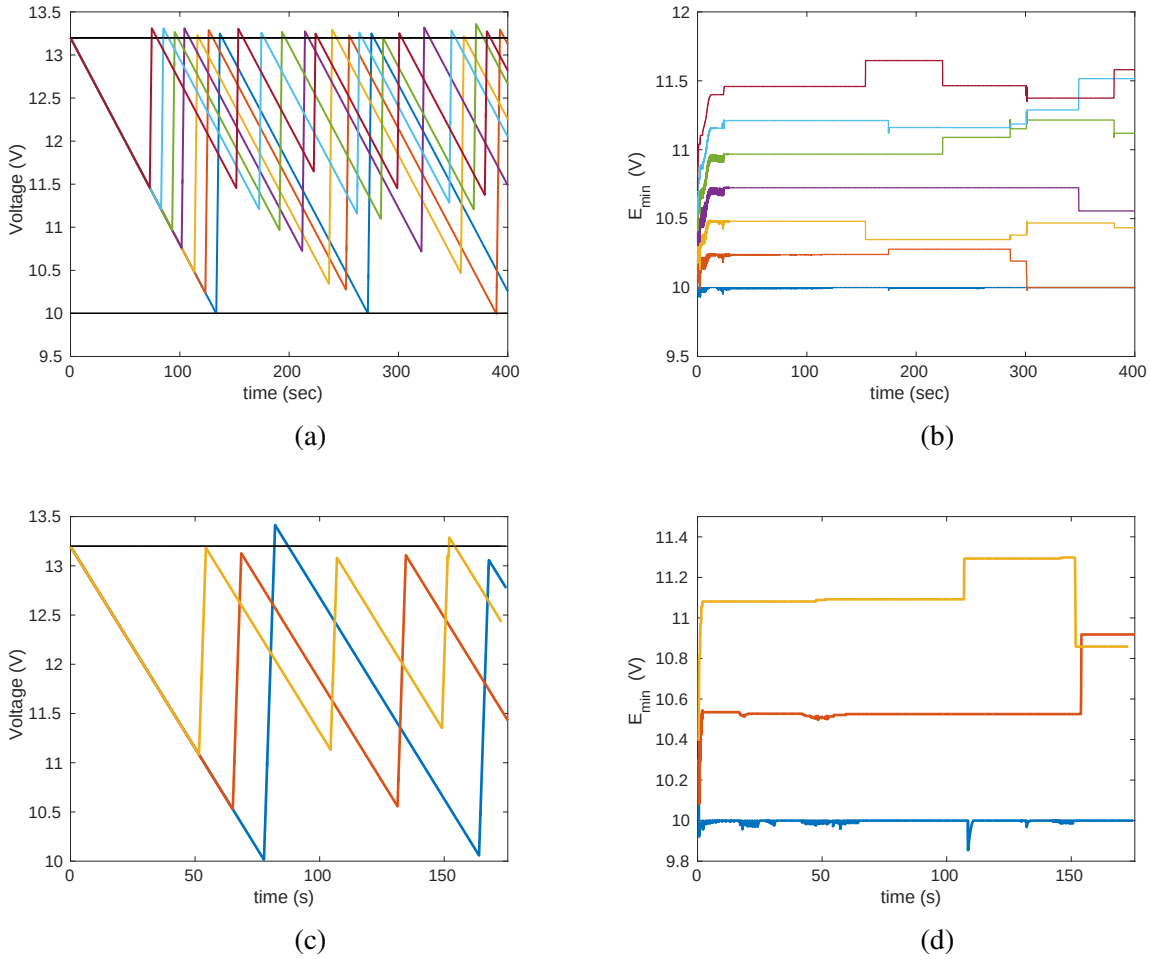


Figure 4.2 Evolution of voltage and E_{min} values for the coverage task in 4.5.1 (4.2a and 4.2b) and the waypoint navigation task in 4.5.2 (4.2c and 4.2d). The occasional overshoots of voltage can be mostly attributed to the difference between the single integrator kinematics and that of an actual robot. There is also some jitter in E_{min} due to the switching nature of Algorithm 1.

CHAPTER 5 ARTICLE 2 : ENERGY AUTONOMY FOR ROBOT SYSTEMS WITH CONSTRAINED RESOURCES

Preface: Long-term autonomy of multi-robot systems requires that robots are able to stay energized consistently. We are interested in situations where a group of robots, sharing a charging station, is carrying out a mission in an open environment.
In this chapter, we provide a mission-agnostic framework based on CBFs that gives guarantees on robots' voltages are above a certain threshold, while ensuring mutually exclusive use of the charging station. we extend the results of the previous chapter by adopting a more accurate voltage discharge model and we modify the coordination framework from the previous chapter to enable one station to serve more robots, and we discuss the station capacity for this setup, i.e. number of serviceable robots by one station, given robots' properties. Moreover, we present experimental validation results of carrying out the proposed framework on a group of three Cognifly drones.

Full Citation: Fouad, Hassan, and Giovanni Beltrame. "Energy Autonomy for Robot Systems With Constrained Resources." IEEE Transactions on Robotics 38.6 (2022): 3675-3693. Manuscript accepted for publication on April 14, 2022.

DOI: https://doi.org/10.1109/TRO.2022.3175438

Abstract: One of the key factors for extended autonomy and resilience of battery-powered multi-robot systems is their ability to maintain energy sufficiency by recharging when needed. In situations with limited access to charging facilities, robots need to be able to share and coordinate recharging activities, with guarantees that no robot will run out of energy. In this work, we present an approach based on Control Barrier Functions (CBFs) to enforce both energy sufficiency (ensuring that no robot runs out of battery) and coordination constraints (guaranteeing mutual exclusive use of an available charging station) in a mission agnostic fashion. Moreover, we investigate the system capacity in terms of the relation between individual robot properties and the limit on temporal separation requirements within charging cycles. We show physics-based simulation results as well as real robot experiments that demonstrate the effectiveness of the proposed approach.

5.1 Introduction

The field of multi-robot systems has seen extensive growth in the past years, as different applications have been explored, such as surveillance [128, 129], search and rescue [130], handling hazardous waste [131], coverage in mobile sensor networks [132], exploration and mapping [133] just to name a few. Although these applications highlight several merits of multi-robot systems, there is still a need to understand how to carry out such missions for an extended period of time in a robust and reliable manner.

Addressing this issue comprehensively is a multifaceted endeavour and calls for studying the effects of several variables that may change during the mission's extended lifetime. Some of these variables are related to the environment in which a mission takes place, while others pertain to robot states and characteristics. One important example of the latter is the robot's energy capacity and consumption, which essentially dictates how long a multi robot system can operate. Therefore, energy autonomy, or the ability of robots to seek recharging during the mission as needed, is one of the key ingredients of long-term autonomy for multi-robot systems.

In this paper we study the energy autonomy of a multi-robot system in scenarios where the recharging resources are limited. More specifically, a scenario in which multiple robots carry out a mission with one shared charging station that can only serve one robot at a time. The robots need to temporally coordinate in a manner that eliminates queueing outside of the charging station, while providing guarantees on the performance of the system. Such temporal coordination eventually leads to the robots alternating on the shared charging station with some difference in arrival times. This problem has been tackled in various ways in literature. For example, Munoz et al. [134] deal with a similar problem and introduce a finite state machine that makes robots alternate over the charging station. Michaud and Robichaud [39] propose a somewhat similar state machine to enable a group of robots to share a limited number of charging stations. These approaches are relatively simple and do not provide guarantees on the performance of the system.

Iftikar and Khan [135] used auctions to tackle the same problem to ensure mutual exclusive use of charging stations among a group of robots. One drawback of traditional auction based methods is the need for an auctioneer that collects bids and makes decisions, which means either a centralized system or one relying on an election process. Either way, the method does not provide guarantees.

A more rigorous approach was adopted by Notomista et al. [44], who present a method based on control barrier functions (CBFs) that gives guarantees on energy sufficiency of individual robots, with each robot having its dedicated charging station. Moreover, Notomista et al. use a simple linear discharge model for batteries, which may not be very representative of many actual battery discharge profiles.

In our previous work [136] we extended the results in [44] by presenting a CBF based framework that achieves the desired temporal coordination among robots, i.e. achieving controlled differences in arrival times to the charging station to ensure mutual exclusivity of its use, while providing guarantees for energy sufficiency of the robots, i.e. voltage never gets below a certain desired threshold, and convergence of the temporal coordination. The framework proposed in [136] has the added advantage of being flexible in the sense that it can be applied on top of a generic mission to ensure the satisfaction of energy sufficiency and coordination constraints. Such flexibility is an added advantage of using CBF constraints in a quadratic program setup as described by Ames et al. [47].

In this work, we extend our results [136] in the following directions:

- we use a more realistic battery exponential discharge model that closely follows experimental data;
- we enhance our coordination framework to allow for longer separations between arrival times, taking in account potential differences in properties among robots. We also present new results for the maximum amount of separation time between arrivals that could be achieved by a system;
- we present experimental results using a group of Cogniflys (lightweight, collision-resilient flying robots) showing the accuracy of the proposed discharge model, and the efficacy of the proposed framework.

This paper is organized as follows: in Section 5.2 we provide a review of the literature related to extended energy autonomy of multi-agent systems, as well as the control barrier functions, which is going to be the main tool in our work. In Section 5.3 we provide a background for our work by presenting some preliminaries, as well as laying out the problem statement and the robot model we use. In Section 5.4 we describe the energy sufficiency framework responsible for ensuring proper voltage levels in robots throughout mission execution. We present the coordination framework in Section 5.5, along with description of system's capacity and overall problem feasibility in section 5.6. We present the main results of the paper in Section 5.7 and discuss these results and conclude the paper in Section 5.8.

5.2 Related work

The energy related aspects for long-term autonomy of multi-robot systems have taken several forms in the literature with varying themes over the time. Early examples had the flavour of energy

awareness in the sense of manipulating the system to extend its execution time as much as possible. One example is the work on node scheduling in sensor networks [28] in which subsets of nodes are activated while keeping others deactivated to extend the lifetime of the network. [29] introduced an algorithm that generates mutually exclusive sets of sensors in a centralized manner, with each set covering the map sufficiently, then the operation of these sets is alternated in an energy efficient manner. Ye et al. [30] adopted node scheduling in a more decentralized way, as their algorithm wakes up each node with a certain probability to probe its neighbours, and become fully active if there are no active neighbours around.

With time, attention has shifted towards mobile robots and the theme has become making a mission execution dependent on the robots' energy content, with the goal of extending their life time. As an example, [35] tackled multi-robot coverage using Voronoi tessellations, introducing an energy augmented metric that results in agents with more energy having larger tessellations to cover. Optimal control was used by [37] to create a rendezvous mission between multiple robots, such that the location and time of gathering depends on initial voltages of the robots, in a way that minimizes energy consumption and ensures non-negative voltage levels for all robots.

Recently, more attention has been given to injecting energy into the system through charging stations. One way to do this is using mobile charging stations, as done by [55] where a persistent surveillance mission is carried out by a group of robots patrolling along predefined trajectories, while having other moving charging stations that rendezvous with the patrolling robots at certain points along their paths. [52] tackled a very similar problem, with mobile depots instead of mobile charging stations. These depots are able to drop batteries at certain locations along the trajectories of the surveying robots so that they can replenish their energy without the need to detour from their paths.

Charging stations can also be static, with the robots visiting them when they need to. One example [44] introduces a control barrier function framework to drive each robot from a coverage mission back to a dedicated charging station in a way that guarantees no voltage falls below a desired level. [137] presented a continuation to this theme by studying a similar problem, considering the charging process taking place in certain parts of the environment to be covered. [50] took another direction, as they tackled the problem of having multiple robots sharing multiple charging stations. The resource allocation problem was modeled as a non cooperative game and they provided an algorithm for calculating the pure strategy Nash equilibrium point, leading to a unique allocation of charging stations.

Another related question to having static charging stations is: how to place these charging stations? [53] studied this question and used the satisfiable modulo theory to solve two problem formulations: 1. knowing the threshold after which a robot heads back to charging station, what is the minimum

number of such stations and where should they be located 2. knowing the available number of charging stations, what is the power threshold and where should these stations be located. [54] took another path, where the environment was modeled as a graph of connected nodes, and the problem was to determine the set of nodes that creates a connected subgraph, embedded in the subgraph of nodes with highest demand, to ensure more accessibility to charging stations for nodes with higher demand.

Another relevant point to static charging stations is the method used for recharging, as this can affect the time spent by a robot inside a charging station. Different methods exist for replenshing robots' energy like contact charging [138, 139] and inductive charging [140, 141], noticing that both take relatively long time to recharge. A faster alternative can be battery swapping [142, 143] that is often used for flying robots.

In our work, we use control barrier functions (CBFs) [47] as the main tool for achieving energy sufficiency and coordination. Barrier functions were originally used in optimization problems [144], but [63] introduced the notion of a barrier certificate that can be used for quantifying the safety of a nonlinear system. These certificates have similar properties to Lyapunov functions, the existence of which ensures system's safety, or that the system state will not wander off to unsafe regions of state space. Also [65] introduced a converse theorem for barrier certificates, that looks conceptually similar to converse Lyapunov theorems: if a system is safe, then it has a valid barrier certificate. Barrier certificates lend themselves to be a more convenient way to quantify systems safety, instead of using other methods that rely more on estimating reachable sets [59, 60, 62] and verifying they do not intersect with unsafe sets, which is a computationally expensive process.

[67] introduced the notion of a control barrier function which can be seen as an extension to barrier certificates, with the aim of calculating the safe input that can ensure the safety of control affine nonlinear system. This extension is conceptually similar to control Lyapunov functions (CLFs) [145, 69] in that both help describing a control action that renders a control affine system safe in case of CBFs, or stable in case of CLFs.

One challenge that emerged after introducing CBFs is the idea that this "safe" control action may not be the desired control input for a robot's arbitrary mission, and this called for the need to "mix" the desired control action with the safe one. For this end, [70] introduced the notion of control Lyapunov barrier function (CLBF) to study the stabilization of nonlinear control affine systems with safety guarantees. The main idea behind CLBFs is to find or construct a bounded continuous function W that has a negative time derivative with respect to unforced system field outside of the unsafe region and is positive in the unsafe region. It can be then shown that if such a function exists, then the system will be stabilized without having the trajectories pass through the unsafe region.

[146] introduced another way to do this "mix" for control affine systems using quadratic programs (QPs). The main idea behind this approach is to construct a QP to minimize the difference between a desired control action, which is mission specific, and the actual control action acting on the system, with a set of linear constraints that are constructed from the time derivative of different CBFs in such a way that ensures positive invariance of the safe set.

5.3 Background

5.3.1 Control Barrier Functions (CBF)

A control barrier function (CBF) [47] is a tool that can be used for ensuring safety of control systems. In this context, safety means ensuring that the states of the system stay in a “safe set” and never wander off to “unsafe” regions.

The safe set is defined to be the superlevel set of a continuously differentiable function $h(x)$ such that [47]:

$$\begin{aligned} \mathcal{C} &= \{x \in \mathbb{R}^n : h(x) \geq 0\} \\ \partial\mathcal{C} &= \{x \in \mathbb{R}^n : h(x) = 0\} \\ Int(\mathcal{C}) &= \{x \in \mathbb{R}^n : h(x) > 0\} \end{aligned} \tag{5.1}$$

which means ensuring that $h(x) > 0$ implies the safe set $\mathcal{C}$ is positively invariant. For a control affine system of the form

$$\dot{x} = f(x) + g(x)u$$

where $u \in U \subseteq \mathbb{R}^m$, with U being the set of admissible control action values, is the input of the system, and f and g are Lipschitz continuous.

In this paper we use the zeroing control barrier functions (ZCBFs) defined in [73], owing to their robustness and asymptotic stability properties [72].

Definition 3. *[73] For a region $\mathcal{D} \subset \mathcal{C}$ a continuously differentiable function $h(x)$ is called a zeroing control barrier function (ZCBF) if there exists an extended class $\mathcal{K}$ function $\alpha(h(x))$ such that*

$$\sup_{u \in U} \left(L_f h(x) + L_g h(x) u + \alpha(h(x))\right) \geq 0 \quad \forall x \in \mathcal{D} \tag{5.2}$$

where $L_f h$ and $L_g h$ are the Lie derivatives of h in the directions of $f(x)$ and $g(x)$ and $\alpha(h(x))$ is an extended class $\mathcal{K}$ function [74] (a function that is strictly increasing, $\alpha(x) = 0$ only at $x = 0$,

and $\lim_{x\to\infty} \alpha(x) = \infty$). We can also define the set K_{zcbf} [73] for the ZCBF $h(x)$ as

$$K_{zcbf} = \{u \in U : L_f h(x) + L_g h(x)u + \alpha(h(x)) \geq 0\} \tag{5.3}$$

which is the set containing all the safe control inputs. Choosing a Lipschitz continuous controller u from the above set is sufficient to guarantee that the safe set $\mathcal{C}$ is forward invariant [73].

The inequality in (5.3) represents a basic requirement on the control action to ensure safety, but this control action is not necessarily that of an arbitrary mission. Quadratic programs (QPs) can be used in this regard to enforce (5.3) as a constraint that has to be satisfied by the system's control action in the following manner [44]:

$$\begin{aligned} u^* = \min_{u} \quad & ||u - u_{nom}||^2 \\ \text{s.t.} \quad & L_f h(x) + L_g h(x)u \geq -\alpha(h(x)) \end{aligned} \tag{5.4}$$

where u_{nom} is the mission related nominal control action. The purpose of u^* is to minimally deviate from u_{nom} while ensuring the satisfaction of the CBF constraint.

5.3.2 Nonsmooth control barrier functions

In presence of multiple and potentially conflicting constraints, the QP problem mentioned above could be infeasible. Thus, it would be beneficial to arbitrate between the constraints to produce a feasible set for the QP solver to handle. One way of achieving this is through control barrier function composition. We adopt the framework described by Glotfelter et al. [81], where boolean nonsmooth control barrier functions (BNCBF) are used to impose forward invariance of safe sets.

We are interested in situations where CBFs are composed using an AND operator, e.g $h = h_1 \wedge h_2 \wedge \dots$ that can be shown to be equivalent to $h = \min\{h_1, h_2, \dots\}$. Such discontinuous functions may invalidate conventional continuous solutions due to discontinuities in the flow field. Therefore, differential inclusions of the form

$$\dot{x}(t) \in F(x(t)), x(0) = x_0$$

where $x \in \Phi \subset \mathbb{R}^n$, to describe dynamics rather than ordinary differential equations, where $F : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ is a compact nonempty, convex and upper semi continuous set valued map, so that the existence of a Carathéodory solution is ensured [81].

Definition 4. *Let h be a function that is Lipschitz near x', then the generalized gradient $\partial h(x')$ is*

$$\partial h(x') = co\{\lim_{i\to\infty} \nabla h(x_i) : x_i \to x', x_i \notin S, \Omega_h\}$$

where co denotes a convex hull, S is an arbitrary set of Lebesgue measure zero in $\mathbb{R}^n$, and Ω_h is the set of points where h is non-differentiable.

Definition 5. *A function h is a valid NCBF if there exists $\alpha : \mathbb{R} \to \mathbb{R}$ that is locally Lipschitz extended class $\mathcal{K}$ function and a measurable locally bounded function $u : \Phi \subset \mathbb{R}^n \to \mathbb{R}^n$ such that*

$$\min\langle \partial h(x'), K\left[f + gu\right](x')\rangle > \alpha(h(x')), \forall x' \in \Phi$$

with $K\left[f + gu\right]$ being the Filippov's operator. The last inequality is a more generalized version of (5.2).

Definition 6. *[81, Definition 7] For $\epsilon > 0$, the almost active set for a candidate BNCBF given by $h = h_1 \wedge h_2$ is defined at each x' as*

$$I_\epsilon(x') = \{i : |h(x') - h_i(x')| < \epsilon\}$$

and the almost active gradient $\partial h_\epsilon : \Phi \subset \mathbb{R}^n \to 2^{\mathbb{R}^2}$ at a point $x' \in \Phi$ is

$$\partial h_\epsilon = \bigcup_{i \in I_\epsilon(x')} \partial h_i(x')$$

Lemma 7. *[81, Theorem 3] let h be a smoothly composed BNCBF, i.e. composed of continuously differentiable functions h_i. If there exists ϵ and locally Lipschitz extended class $\mathcal{K}$ function α s.t. the following QP*

$$\begin{aligned} u^*(x') \in \arg\min_u \quad & u^\top A(x')u + b^\top(x')u \\ \textit{s.t.} \quad & \langle \partial h_\epsilon(x'), f(x') + g(x')u\rangle \geq -\alpha(h(x')) \end{aligned} \tag{5.5}$$

has a solution for every x' and u^* is locally measurable and bounded, then h is a valid NCBF.

5.3.3 Robot model

We use a single integrator model to describe robot's kinematics, along with an exponential battery discharge model

$$\begin{aligned} \dot{x} &= u \\ V(t) &= V_0 + k_b\left(e^{-k_e(t-t_0)} - 1\right) \\ \Rightarrow \dot{V} &= k_e(V_0 - V - k_b) \end{aligned} \tag{5.6}$$

where $x \in \mathbb{R}^d$ is the position of the robot, V is the battery voltage, V_0 is the maximum battery voltage, k_e and k_b are battery related discharging coefficients and t_0 is the time at the beginning of discharge. For simplicity, we consider the dimension size $d = 2$, but the results are still applicable for $d = 3$ as well. We refer to $\mathbf{x} = \begin{bmatrix} x & V \end{bmatrix}^\top \in \mathcal{D} \subset \mathbb{R}^{d+1}$ as being the collective state of a robot. Figure 5.1 shows the actual voltage change of a Cognifly [122] while hovering inside a lab environment and shows the efficacy of an exponential discharge model in describing the actual battery discharge.

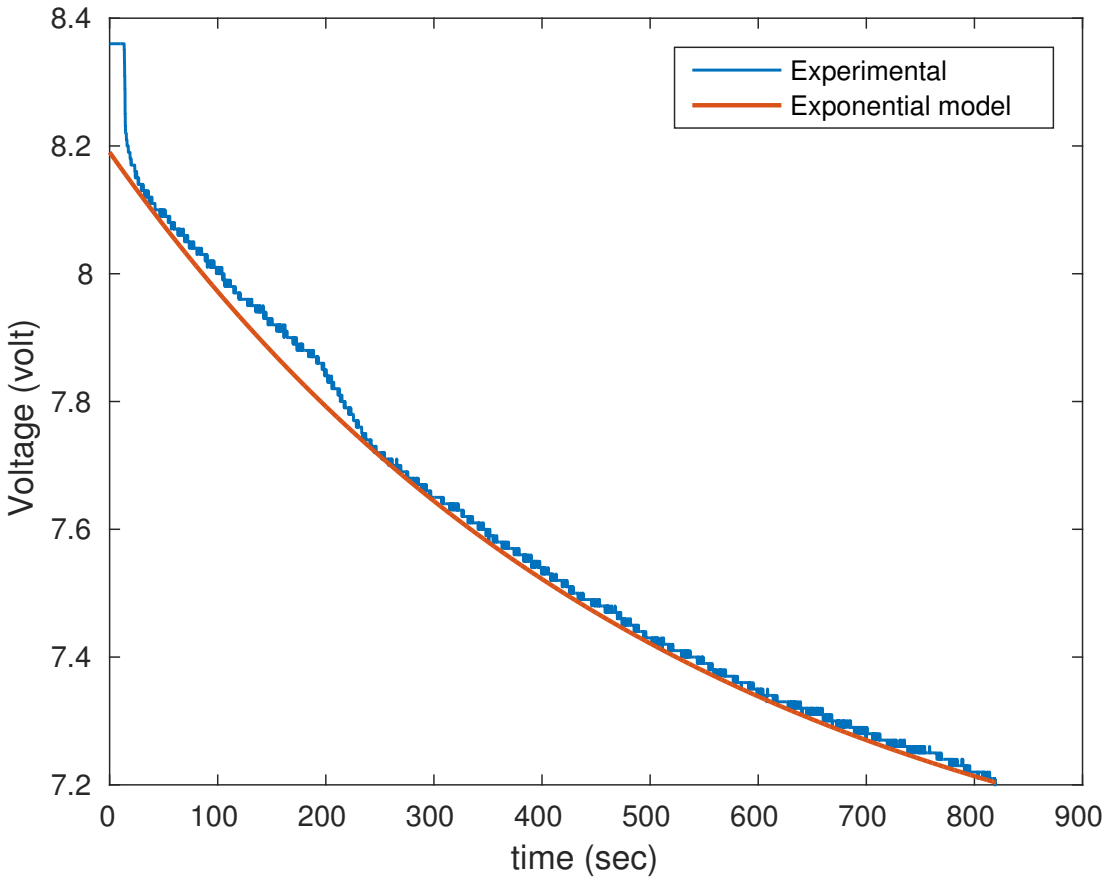


Figure 5.1 Comparison between actual voltage discharge of a hovering Cognifly drone, and the exponential discharge model with $V_0 = 8.19, k_b = 1.2411, k_e = 0.00193$.

For charging, we adopt a simple charging model that is agnostic to different charging modalities. We suppose that a robot gets within a distance δ from a charging station, then it automatically spends a specific amount of time t_{ch} till it finishes recharging and the voltage goes back to V_0 again. This charging station is located at position $x_c \in \mathcal{D} \subset \mathbb{R}^2$ that is known to all robots.

5.3.4 Problem statement

Given N robots, each modeled by the dynamics in (5.6) carrying out a generic mission, and supposing there exists one charging station, whose position is known for all robots, with one available charging slot, we need to ensure that:

- **Safety:** The battery voltage of any robot never goes below a certain desired lower bound V_{lb}.
- **Coordination:** No more than one robot will be at the charging station at any time. This requirement aims to respect the available charging slots (one per charging station), and to avoid queueing at the station. This can be achieved by making the robots arrive at the charging station at different times that are separated by at least δ_t. This way the robots alternate on the charging station periodically.
- **Feasibility:** The amount of time δ_t between arrivals should be commensurate to the number of robots N and the discharge properties of these robots. In other words, there is a maximum limit on δ_t that can be achieved in the system given N and the robots' discharge properties, and such limit should be respected.

5.3.5 Solution strategy

Figure 5.2 depicts a schematic overview of the strategy we adopt in this work. The core idea in this paper is similar to what we did in [136], that is to build two CBF-based layers: (i) an energy sufficiency layer, (ii) and a coordination layer. The energy sufficiency layer ensures the voltage of each robot stays higher than a certain minimum level $V_{min_i}, \quad i = 1, \dots N$. This layer is a "tracker" for the desired V_{m_i} value.

The coordination layer, on the other hand, is responsible for manipulating the value of V_{m_i} for each robot to ensure mutual exclusive use of the charging station. We do this by finding the relationship between V_{m_i} and the time at which each robot arrives at the charging station, and formulate a constraint on these arrival times to make sure they are properly separated, consecutive arrivals to CS are separated by a desired temporal separation δ_t. We also investigate the station capacity, namely, the relationship between robots' discharge properties and the allowable number of serviceable robots by one CS, as well as the allowable range for temporal separation δ_t.

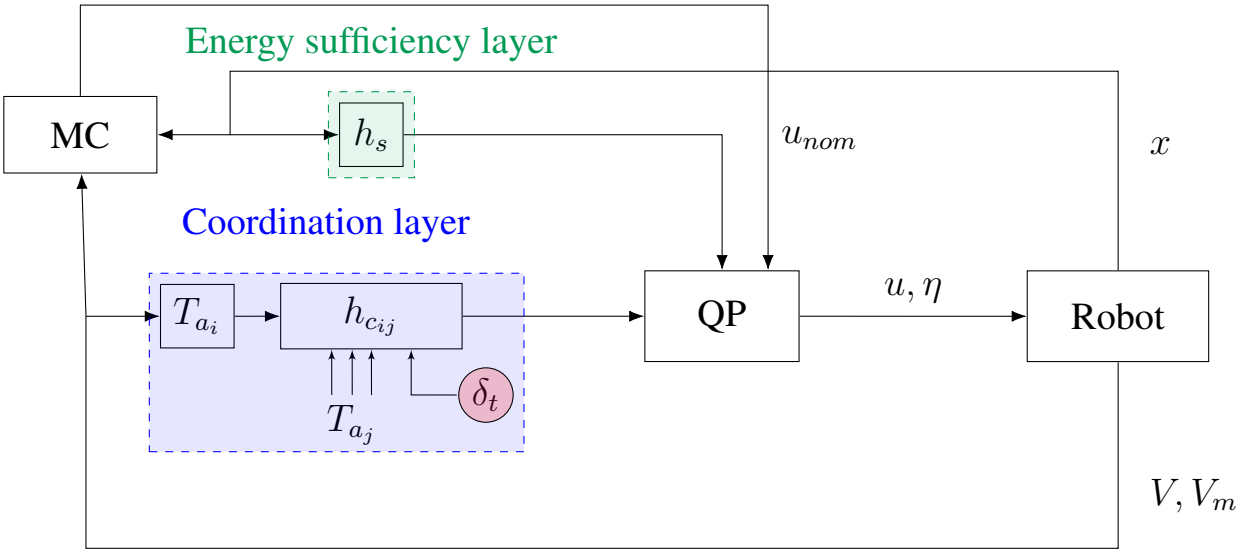


Figure 5.2 A schematic of the adopted strategy. Both ES and coordination layers consist of CBFs, and produce linear constraints that are used in the QP, which in turn manipulates the nominal control action produced by a mission controller.

5.4 Energy sufficiency framework

5.4.1 Energy sufficiency CBF

The goal of this CBF is to enforce proper constraints that will ensure that a robot's battery voltage does not get below a certain minimum level V_{min}. We propose the following candidate CBF:

$$h_s(\mathbf{x}) = V - V_m - \Gamma(\mathbf{x}) \tag{5.7}$$

such that

$$\Gamma(\mathbf{x}) = k_b \left(1 - e^{-\frac{k_e}{v_r}(D-\delta)}\right) \tag{5.8}$$

where V is the robot's voltage, V_m is the minimum voltage a robot should sustain, δ is the effective radius of the charging station, $D = ||x - x_c||$ is the distance from the charging station, v_r is the desired velocity by which the robot returns back to the charging station and $\Gamma(\mathbf{x})$ represents the amount of voltage needed to get back to the charging station. The linear constraint associated with the candidate CBF h_s is

$$\underbrace{k_e(V_0 - V - k_b)}_{\dot{V}} - \frac{k_b k_e}{v_r} e^{-\frac{k_e}{v_r}(D-\delta)} \frac{(x - x_c)^T}{D} u \geq -\alpha(h_s) \tag{5.9}$$

Note that according to Lemma 1 in [44], if $D > \delta$ we need $\Gamma(\mathbf{x}) = 0$ only at the boundary of the charging station $D = \delta$, which means $V = V_m$ at the boundary of the charging station by design. In other words, as long as the robot is outside of the charging station, the candidate CBF $h_s(\mathbf{x})$ should lead to moving the robot in such a way that $V \geq V_m$, and that $V = V_m$ only happens at the boundary of the charging station.

In the next theorem we prove that $h_s(\mathbf{x})$ is indeed a zeroing CBF.

Theorem 4. *For a robot with dynamics described in* (5.6)*, and having $u \in U =\in \mathbb{R}^d$, where U is the set of possible values for the control input u in* (5.6)*. Suppose that $u_p = \{u \in \mathbb{R}^d : \frac{(x-x_c)^T}{D} u = -v_r\}$, then $h_s(\mathbf{x})$ is a ZCBF if $u_p \in U$.*

Proof. To show that $h_s(\mathbf{x})$ is a CBF, it has to satisfy $h_s(\mathbf{x}(t_0)) \in \mathcal{C} \Rightarrow h_s(\mathbf{x}(t)) \in \mathcal{C}, \forall t > t_0$ for that it has to satisfy

$$\dot{h}_s \geq -\alpha(h_s) \Rightarrow L_f h_s + L_g h_s u \geq -\alpha(h_s) \tag{5.10}$$

without loss of generality we consider $t_0 = 0$ in (5.6) and differentiating (5.7) we get

$$\dot{h}_s = -k_b k_e \left(e^{-k_e t} + e^{-\frac{k_e}{v_r}(D-\delta)} \frac{\dot{D}}{v_r} \right) \tag{5.11}$$

suppose at time t_s and a distance D_s away from the charging station, $h_s(t_s) > 0$ and the robot starts moving such that $\dot{D} = -v_r$ (noticing that $\dot{D} = \frac{(x-x_c)}{D} u$ and that such input is achievable since $u_p \in U$), then $D(t) = D_s - v_r(t - t_s) = D_s - v_r \Delta t$, and

$$\dot{h}_s = k_b k_e \left(-e^{-k_e t_s} e^{-k_e \Delta t} + e^{-\frac{k_e}{v_r}(D_s-\delta)} e^{k_e \Delta t} \right) \tag{5.12}$$

in case $e^{-\frac{k_e}{v_r}(D_s-\delta)} \geq e^{-k_e t_s}$ then $\dot{h}_s \geq 0$ owing to the growing exponential term. This means if $h(t_s) > 0 \Rightarrow h(t) > 0, \forall t > t_s$. In case $e^{-\frac{k_e}{v_r}(D_s-\delta)} \leq e^{-k_e t_s}$, i.e. the control action u causing $\dot{D}$ as been activated early, then $\dot{h}_s$ starts negative at $\Delta t = 0$ till the positive exponential term catches up. The time at which $\dot{h}_s = 0$ in (5.12) is

$$\Delta t_c = \frac{1}{2} \left(\frac{D_s - \delta}{v_r} - t_s \right) \tag{5.13}$$

and note that t_s has to be less than the time to return back to the charging station for Δt_c to be valid (hence $e^{-\frac{k_e}{v_r}(D_s-\delta)} \geq e^{-k_e t_s}$). What we want to show is that although $\dot{h}_s$ starts negative, the value of h_s will stay positive till $\dot{h}_s$ becomes positive again. To show this we compare $h_s(t_s + \Delta t_c)$ with the value of h_s at a time t_s^* s.t. $t_s^* > \frac{D_s-\delta}{v_r}$ and $h_s(t_s^*) > 0$ and $D(t_s^*) = D_s$ (basically at a time when $\dot{D} = -v_r$ is activated at a time greater than $\frac{D_s-\delta}{v_r}$, and notice that we have established the forward

invariance for this case) so

$$\begin{aligned} &h_s(t_s + \Delta t_c) - h_s(t_s^*) = \\ &k_b \Big(e^{\frac{-k_e}{2}\left(\frac{D_s-\delta}{v_r}+t_s\right)} - e^{-k_e t_s^*} \\ &+e^{\frac{-k_e}{2}\left(\frac{D_s-\delta}{v_r}+t_s\right)} - e^{\frac{-k_e}{v_r}(D_s-\delta)} \Big) \end{aligned} \tag{5.14}$$

we notice that the difference between the first two terms and the last two terms is positive, since $t_s < \frac{D_s-\delta}{v_r}$ and $t_s^* > \frac{D_s-\delta}{v_r} > \frac{1}{2}\left(\frac{D_s-\delta}{v_r} + t_s\right)$ thus $h_s(t_s + \Delta t_c) - h_s(t_s^*) > 0$ which means that even if $\dot{h}_s(t_s) < 0$ (if $t_s < \frac{D_s-\delta}{v-r}$) and provided that $h_s(t_s) > 0$, then the value of h_s is still positive until $\dot{h}_s(t) > 0$ again.

In conclusion, we show that the control action u leading to $\frac{(x-x_c)^T}{D}u = -v_r$,i.e. that leads the robot to go back to the charging station with a constant speed v_r is indeed a solution that satisfies (5.10) and maintains forward invariance of the safe set $\mathcal{C}$, thus h_s is a ZCBF.

□

What we have shown in the last proof that when h_s is used as a constraint, returning the robot back to the charging station with a constant speed is sufficient, and this speed could be set to a value within realistic bounds of control action manageable by the robot.

It is also worth mentioning that h_s serves as a CBF that generates constraints that are intended to be used when $D > \delta$, since we do not care technically about what happens inside, as we assume a fixed waiting time in which recharging takes place.

In the next lemma we investigate the relation between the choice of $\alpha(h)$ in (5.9) and the voltage level of the robot when it arrives at the charging station. We use a linear $\alpha(h_s)$ function of the form

$$\alpha(h_s) = \beta_s h_s \tag{5.15}$$

where $\beta_s > 0$.

Lemma 8. *Given a robot with dynamics described in (5.6), and for the QP (5.4) using the ZCBF (5.7), the voltage difference $V - V_m$ is bounded by zero from below and by a quantity that decays exponentially with β_s from above.*

Proof. We start by noticing that in the QP (5.4), as long as the inequality is satisfied by u_{nom}, $u = u_{nom}$ will be the solution of the minimization problem. Otherwise, the resulting control input u satisfies

$$\dot{h}_s = -\beta_s h_s \Rightarrow h_s(t) = h_s(T)e^{-\beta_s(t-T)} \tag{5.16}$$

but from (5.7)

$$V - V_m - \Gamma(\mathbf{x}) = h_s(T)e^{-\beta_s(t-T)} \tag{5.17}$$

and on arrival at the boundary of the charging station at time $t = t_a$ we have $D = \delta$, thus

$$V(t_a) - V_m = h_s(T)e^{-\beta_s(t_a-T)}. \tag{5.18}$$

To show that $V(t_a) - V_m$ is bounded from below, it suffices to consider that if the robot starts in the safe set (i.e. $h_s(t_0) \geq 0$), then respecting the constraint in (5.4) for h_s ensures that $h_s(T) \geq 0$ as well (by the virtue of the fact that h_s is a ZCBF), so the right hand side in equation (5.18) is not less than zero.

$V(t_a) - V_m$ is bounded from above by equation (5.18), which shows that the bound on this difference decays exponentially with β_s. □

Remark 3. *It is worth noting that Lemma 8 shows that the ZCBF h_s has a tracking property in the sense that the voltage at the time of arrival at the charging station is close to V_m with a margin, the tightness of which can be manipulated.*

From equation (5.6) we can calculate the time of a full discharge as

$$T(V_0, V_m) = \frac{1}{k_e} \log \frac{k_b}{k_b + V_m - V_0} \tag{5.19}$$

Given the tracking ability of h_s we have discussed we can define the arrival time, which is the time remaining for each robot till it arrives back to the charging station, as

$$\begin{aligned} T_a(V_0, V) &= T(V_0, V_m) - T(V_0, V) \\ &= \tfrac{1}{k_e} \log \frac{k_b + V - V_0}{k_b + V_m - V_0} \end{aligned} \tag{5.20}$$

the above relation expresses the difference between the total time till V reaches V_m starting from V_0, and the time that has been already spent to get to V from V_0. This definition will be used later in the coordination framework.

5.5 Coordination framework

The purpose of the coordination framework is to ensure mutual exclusive use of the available charging station. The proposed strategy to achieve this is through manipulating V_m for each robot to change their arrival times.

Assumption 1. *The underlying communication graph between the robots is a complete graph, meaning that each robot can receive information from all other robots.*

It is worth mentioning that although assumption 1 may seem a bit constrictive, but this can be practically achieved through using multi-hop communication to spread each agent's information in a gossip like fashion [127].

The mechanism we propose to change the value of V_m is to assume that it changes according to the following dynamics

$$\dot{V}_m = \eta \tag{5.21}$$

where $\eta \in \Theta = \mathbb{R}$ with Θ being the set of all admissible values of η, and the nominal value of choice for η is zero (its default value is zero unless changed by other control laws to fulfill certain constraints, which means by default the value of V_m does not change). Also $V_{lb} > 0$ is a lower bound for V_m, which can be related to the physical characteristics of the robot.

Our main strategy is twofold: (i) introduce a constraint to ensure that the value of V_m stays above the lower bound V_{lb}, and (ii) introduce a new CBF, $h_{c_{ij}}$,to keep the difference in arrival times between any two robots i, j above a desired value δ_t through the manipulation of V_m for each robot.

5.5.1 Pairwise coordination CBF

This CBF aims at separating the arrival times of two robots with at least δ_t, and the core idea is very similar to the collision avoidance strategy proposed in [46], albeit it is to avoid collision of arrival times. For that we define a pairwise safe set $\mathcal{C}_{ij}$

$$\mathcal{C}_{ij} = \{(V_{m_i}, V_{m_j}) \in \mathbb{R}^2 : h_{c_{ij}} \geq 0, \forall j \in \mathcal{N}_i'\} \tag{5.22}$$

where $\mathcal{N}_i'$ is the set of neighbours of robot i that are outside of the charging station. We propose the following function $h_{c_{ij}}$ between two agents (i, j)

$$h_{c_{ij}} = \log \frac{|T_{a_i} - T_{a_j}|}{\delta_t}, \quad \forall j \in \mathcal{N}_i' \tag{5.23}$$

where T_{a_i} is the arrival time of robot i as described in (5.20), , and δ_t is the desired separation in arrival times between robots. The resulting constraint for the QP problem is

$$\frac{T_{a_i} - T_{a_j}}{|T_{a_i} - T_{a_j}|^2} \left(\dot{T}_{a_i} - \dot{T}_{a_j}\right) \geq -\alpha(h_{c_{ij}}), \quad \forall j \in \mathcal{N}_i' \tag{5.24}$$

where

$$\frac{dT_{a_i}}{dt} = -1 - \frac{\eta_i}{k_{e_i}(k_{b_i} + V_{m_i} - V_0)} \tag{5.25}$$

We can obtain a decentralized version of (5.24) by dropping out the term η_j from $\dot{T}_{a_j}$ in (5.24) , which means that agent i will deal with j as an external input in a sense, giving

$$-\frac{T_{a_i} - T_{a_j}}{|T_{a_i} - T_{a_j}|^2} \frac{1}{(k_{b_i} + V_{m_i} - V_0)} \eta_i \geq -\alpha(h_{c_{ij}}), \quad \forall j \in \mathcal{N}_i' \tag{5.26}$$

For the $\alpha(h_{c_{ij}})$ function in (5.26) we propose the form

$$\alpha(h_{c_{ij}}) = \beta_c h_{c_{ij}} \tag{5.27}$$

with $\beta_s > 0$. In the next lemma we want to show that $h_{c_{ij}}$ is a ZCBF

Lemma 9. *For a multi robot system with (5.6) as the dynamics of each robot, then for a pair of robots (i, j) satisfying $T_{a_i}(t_0) \neq T_{a_j}(t_0)$, $h_{c_{ij}}$ is a ZCBF for $\eta \in \Theta = \mathbb{R}$, with (5.26) rendering the set $\mathcal{C}_{ij}$ forward invariant.*

Proof. If $T_{a_i}(t_0) \neq T_{a_j}(t_0)$, and since $\eta \in \Theta = \mathbb{R}$ there exists a control action η that satisfies (5.26), thus $h_{c_{ij}}$ is a ZCBF. □

We note that if $(V_{m_i}(t_0), V_{m_j}(t_0)) \notin \mathcal{C}_{ij}$,then by virtue of $h_{c_{ij}}$ being a ZCBF then $(V_{m_i}(t), V_{m_j}(t))$ to converge to $\partial\mathcal{C}_{ij}$ asymptotically owing to the asymptotic properties of ZCBFs [72]. The speed of convergence and closeness to the boundary can be gauged by the choice of β_c in (5.27).This could be taken into account when choosing δ_t (so for example $\delta_t' = \delta_t + \epsilon$ could be chosen for finite time convergence to a desired δ_t).

Remark 4. *For $h_{c_{ij}}$, $\dot{T}_{a_i} = \dot{T}_{a_j}$ is a point of singularity for the gradient of $h_{c_{ij}}$. Hence this requires starting at different values of $\dot{T}_{a_i}$ for all robots, which can practically be done by choosing V_{m_i} depending on each robot's index, provided that they all have the same maximum voltage V_0.*

It is worth mentioning that the main motivation in the choice of (5.27) is the idea that each robot starts with an estimate of its V_{m_i} that does not necessarily lie in the pairwise safe set $\mathcal{C}_{ij}$, and then generates a control action η_i that satisfies (5.26) and satisfies the desired separation of arrival times in a finite time.

Additionally, the above result is applicable even if one robot i applies (5.26) with respect to another robot j, while robot j is not. This follows the same argument that since $\eta_i \in \Theta = \mathbb{R}$ then there is always a value of η_i that satisfies (5.26).

5.5.2 V_m lower bound CBF

Lower bounding E_{min} corresponds to the requirement of an acceptable voltage at the beginning of the charging process to avoid permanent damage to the battery. The proposed CBF for this matter is

$$h_L = k_p \log \frac{V_m}{V_{lb}} \tag{5.28}$$

where $V_{lb} > 0$ is the desired lower bound voltage and $k_p > 0$ is a scaling gain. The constraint for the QP (5.4) is:

$$k_p \frac{\eta}{V_m} \geq -\alpha(h_L) \tag{5.29}$$

and we choose

$$\alpha(h_L) = \beta_L h_L. \tag{5.30}$$

Lemma 10. *For a robot with dynamics (5.6) and V_m satisfying (5.21), h_L is a ZCBF.*

Proof. Since $\eta \in \Theta = \mathbb{R}$, so it is always the case that a value of η could be found that satisfies (5.2) for the constraint (5.29), so if $V_m(t_0) \geq V_{lb}$ then (5.29) ensures positive invariance of the safe set $\mathcal{C}_{V_m} = \{V_m \in \mathbb{R} : V_m \geq V_{lb}\}$. □

Remark 5. *As the value of V_m is changing due to input, it could be considered one of the robot's states and thus we can augment the state vector to be $\mathbf{x} = \begin{bmatrix} x & V & V_m \end{bmatrix}^\top$. Also the energy sufficiency constraint* (5.9) *needs to be modified to account for changing V_m, and hence becomes*

$$k_e(V_0 - V - k_b) - \eta - \frac{k_b k_e}{v_r} e^{-\frac{k_e}{v_r}(D-\delta)} \frac{(x - x_c)^T}{D} u \geq -\alpha(h_s) \tag{5.31}$$

but the result of lemma 8 still holds, since the idea of this lemma is that a set of control inputs (u, η) will be produced from the QP that satisfies (5.16)*, and hence the exponential decay of h_s leads to the bounded difference $V - V_m$ on the boundary of the charging region.*

Remark 6. *We note that the robot dynamics is Lipschitz outside of the charging station. Moreover, the CBFs for energy sufficiency and coordination defined thus far and their associated constraints are all continuous outside of the charging region. For this end each robot coordinates with neighbours that are outside of the charging station because a robot that has already arrived to the charging station does not affect the arrival times of subsequent robot.*

5.5.3 Coordination framework

In this section we construct for each robot a boolean nonsmooth CBF out of the pairwise coordination and lower bounding CBFs discussed earlier. The motivation behind this is that V_m, that we are using as a coordination variable, could be subject to contradictory constraints. For example, it may need to decrease to temporally coordinate with another robot, but it needs to increase in the same time to not go below V_{lb}. One solution to this is to arbitrate between different constraints and consider the most imminent at the moment. We can encode this as a boolean expression as following

$$h_{c_i} = \bigwedge_{j \in \mathcal{N}_i'} h_{c_{ij}} \wedge h_L$$

this effectively equivalent to

$$h_{c_i} = \min\{h_{c_{ij}}, h_L\}, \quad \forall j \in \mathcal{N}_i \tag{5.32}$$

with $\mathcal{N}_i$ being the neighbour list of robot i. For each robot the safe set is

$$\mathcal{C}_i = \bigcap_{j \in \mathcal{N}_i'} \mathcal{C}_{ij} \cap \mathcal{C}_L$$

with $\mathcal{C}_L = \{V_m \in \mathbb{R} : V_m \geq V_{lb}\}$. Since $h_{c_{ij}}$ are all continuously differentiable except at points where $T_{a_i} = T_{a_j}$ (which we start away from, and do not have by virtue of applying pairwise coordination CBFs), and h_L is continuously differentiable (expect at $V_m = 0$ which we start away from and do not get close to by virtue of V_m lower bound CBF), then h_{c_i} is smoothly composed, and the gradients of $h_{c_{ij}}$ and h_L are described in (5.26) and (5.29). We can follow the same line of reasoning in lemmas 9 and 10 by considering that since $\eta \in \mathbb{R}$, and since η is only the control action affecting h_{c_i}, then there exists a value of η that respects the constraint in (5.5). The QP combining both constraints for each robot is

$$\begin{aligned} \mathbf{u}^* = \min_{\mathbf{u}} \quad & \mathbf{u}^\top A \mathbf{u} + b^\top \mathbf{u} \\ \text{s.t.} \quad & L_f h_s + L_g h_s \mathbf{u} \geq -\beta_s h_s \\ & \langle \partial_\epsilon h_{c_i}, f + g\eta \rangle \geq -\beta_c h_{c_i} \end{aligned} \tag{5.33}$$

where for the combined system $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & \eta \end{bmatrix}^\top$, with (u_1, u_2) being the velocity control input in (5.6) and $L_g h_s \mathbf{u} = -\eta - \frac{k_b k_e}{v_r} e^{-\frac{k_e}{v_r}(D-\delta)} \frac{(x-x_c)^T}{D} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$. Note that A could be chosen as identity matrix, and $b = -2u_{nom}$ so that the QP would be solving a least square problem of the form $\min ||u - u_{nom}||^2$.

5.6 Limits on temporal separation

The purpose of the coordination framework described thus far is to impose temporal constraints on the difference between the expected arrival times of any two robots. However, the value of desired difference δ_t is not arbitrary and has to respect some limits.

To motivate this idea, suppose a robot can survive on one battery for a certain amount of time T, i.e. its voltage starts at V_0 and ends at V_{lb} over a period T, then theoretically speaking if we have two robots then the maximum δ_t we can ask for can not be greater than T (that is a robot starts with a fresh battery exactly at the time the other has depleted its own and about to recharge). Moreover, one would expect that when a new robot is added, the maximum value of δ_t will be even less.

The question now becomes: how to estimate this limit? And what is the relationship between this limit and the number of robots in the system, as well as the discharge and recharge characteristics of the system?

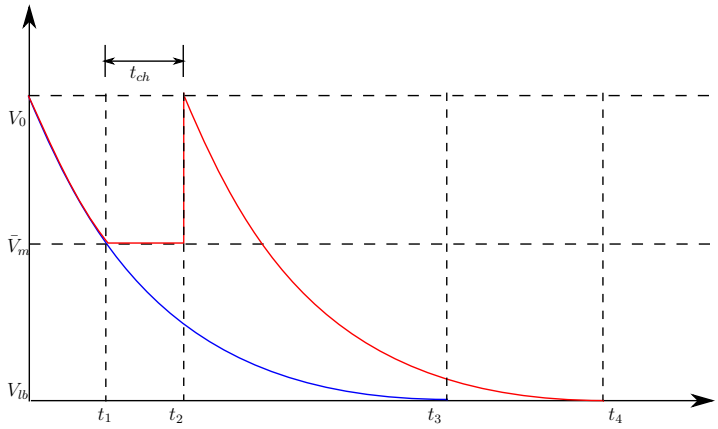


Figure 5.3 Schematic of the voltage for the first and last robots to recharge in one cycle, we call them $\mathcal{A}$ (in red) and $\mathcal{Z}$ (in blue) respectively. We call δ_s the terminal separation time, which is the separation time between first arrival of robot $\mathcal{Z}$ and the second arrival of robot $\mathcal{A}$.

Proposition 1. *For a group of N robots with same discharge characteristics, each staying for an amount of time T_{ch} inside the charging station to recharge, and if the value of V_{m_i} for each robot is $V_{m_i} = V_{lb}$ from the second discharge cycle onwards, then the maximum allowable temporal separation that can be achieved among robots is*

$$\delta_{t_{cr}} = \tfrac{1}{N}(T_a(V_0, V_{lb}) + T_{ch})$$

Proof. Consider a scenario where a group of N robots with each robot having a distinct value of V_{m_i}, and we want them to arrive at the charging station with a separation of arrival times equal to δ_t. We specifically consider the robot with the maximum value of $V_m = \bar{V}_m$, and the one with

smallest value of $V_m = V_{lb}$. We call the former robot $\mathcal{A}$ and the latter robot $\mathcal{Z}$. Since robot $\mathcal{A}$ has the highest value of V_m, it will be the first to visit the charging station, while robot $\mathcal{Z}$ will be the last, by virtue of having the least V_m value. This is depicted in Figure 5.3.

If V_m for robot $\mathcal{A}$ is $V_m = V_{lb}$ after the first visit to the charging station, we want the time difference between the first arrival of robot $\mathcal{Z}$ and the second arrival of robot $\mathcal{A}$ (we call it $\delta_s = T_4 - T_3$ in Figure 5.3), to be at least equal to δ_t. Supposing that each robot spends an amount of time T_{ch} inside the charging station to recharge, so what we want can be expressed as

$$T_{a_{\mathcal{A}}}(V_0, \bar{V}_m) + T_{ch} + T_{a_{\mathcal{A}}}(V_0, V_{lb}) - T_{a_{\mathcal{Z}}}(V_0, V_{lb}) \geq \delta_t \tag{5.34}$$

Notice that at this point we still do not know δ_t and $\bar{V}_m$. Now suppose we divide the time between the first arrival of robot $\mathcal{A}$ and the first arrival of robot $\mathcal{Z}$ into $N-1$ equal intervals (corresponding to uniformly timed arrivals of N agents from T_1 to T_3 in Figure 5.3), then we want this interval to be at least equal to δ_t

$$\frac{T_{a_{\mathcal{Z}}}(V_0, V_{lb}) - T_{a_{\mathcal{Z}}}(V_0, \bar{V}_m)}{N-1} \geq \delta_t \tag{5.35}$$

Supposing that all robots have the same discharge properties, then (5.34) becomes

$$\begin{aligned} &\tfrac{1}{k_e} \log \tfrac{k_b}{k_b + \bar{V}_m - V_0} + T_{ch} \geq \delta_t \\ &\Rightarrow k_b + \bar{V}_m - V_0 \geq k_b e^{-k_e(\delta_t - T_{ch})} \end{aligned} \tag{5.36}$$

and (5.35) becomes

$$\begin{aligned} &\tfrac{1}{k_e(N-1)} \log \tfrac{k_b + \bar{V}_m - V_0}{k_b + V_{lb} - V_0} \geq \delta_t \\ &\Rightarrow k_b + \bar{V}_m - V_0 \geq e^{(N-1)k_e\delta_t}(k_b + V_{lb} - V_0) \end{aligned} \tag{5.37}$$

taking the equality and equating the right hand side of the last two relations we get

$$\begin{aligned} \delta_{t_{cr}} &= \tfrac{1}{N}\left(\tfrac{1}{k_e} \log \tfrac{k_b}{k_b + V_{lb} - V_0} + T_{ch}\right) \\ &= \tfrac{1}{N}(T_a(V_0, V_{lb}) + T_{ch}) \end{aligned} \tag{5.38}$$

The last relation describes the maximum amount of temporal separation in arrival times given the number of robots N, the total discharge time of one cycle and the waiting time in the charging station. □

Thus for a normal operation δ_t has to satisfy

$$\delta_t \leq \tfrac{1}{N}(T_a(V_0, V_{lb}) + T_{ch}) \tag{5.39}$$

Remark 7. *Equation* (5.38) *could be rearranged to calculate the value of one of the variables with respect to others, so for example we can calculate the maximum allowable recharge time $T_{ch_{cr}}$ if we know the the number of available robots N, a desired value of the separation time δ_{t_d} and the total discharge time. We consider this relation to be expressive of the capacity of the system.*

Remark 8. *We notice that the first expression of* (5.38) *is the division of the total cycle period by the number of robots, which is an intuitive result for the feasible separation period. However, we notice that the maximum available temporal separation is directly proportional to the charging time T_{ch}, and this is due to the fact that when T_{ch} increases, the time of arrival of robot $\mathcal{A}$ for the second time has additional delays, which increases the difference $T_4 - T_3$ in Figure 5.3 leading to an additional margin for increasing δ_t.*

Based on the last remark, there should be a limit on T_{ch} beyond which the problem becomes infeasible (or else a robot would arrive to the charging station while another robot has not finished its T_{ch} waiting period). Thus $T_{ch} \leq \delta_{t_{cr}}$. In the critical case when they are equal, substituting in (5.38) we get

$$T_{ch_{cr}} = \frac{1}{(N-1)k_e} \log \frac{k_b}{k_b + V_{lb} - V_0} \tag{5.40}$$

which describes the maximum allowable charging time for a system of N robots with given discharging characteristics.

Remark 9. *Supposing that not all robots are having the same discharge properties, but they have the same maximum voltage V_0. We take robot $\mathcal{A}$ as the robot with the fastest discharge rate among all robots (thus it could be considered the neediest robot) and robot $\mathcal{Z}$ as the one with the slowest discharge rate. We can still use* (5.34) *and the reasoning behind it, but $T_{a_{\mathcal{A}}}(V_0, V_{lb}) - T_{a_{\mathcal{Z}}}(V_0, V_{lb}) < 0$. We define $T_p = T_{ch} + T_{a_{\mathcal{A}}}(V_0, V_{lb}) - T_{a_{\mathcal{Z}}}(V_0, V_{lb})$ and follow same steps in* (5.36) *and* (5.37) *and in this case the new $\delta_{t_{cr}}$ is*

$$\delta_{t_{cr}} = \frac{1}{N} \left(\frac{1}{k_e} \log \frac{k_b}{k_b + V_{lb} - V_0} + T_p \right) \tag{5.41}$$

but we note that since $T_{a_{\mathcal{A}}}(V_0, V_{lb}) - T_{a_{\mathcal{Z}}}(V_0, V_{lb}) < 0$, then $T_p < T_{ch}$, which means that the amount of maximum allowable temporal separation diminishes proportionately to the difference between the full discharge time of the slowest discharging and fastest discharging robots.

Our treatment thus far for $\delta_{t_{cr}}$ depends on the fact that the value of V_m changes to V_{lb} after the first charging cycle. However, the proposed coordination framework is not equipped for doing so. For that we propose to introduce a state jump inside the charging station for $V_m \to V_{lb}$.

Theorem 5. *Given a group of N robots with dynamics described in* (5.6)*, and applying both energy sufficiency and coordination constraints as in* (5.33)*. Suppose a state jump $V_m \to V_{lb}$ whenever*

$D \leq \delta$, and that a δ_t is chosen to satisfy (5.39)*, then the set*

$$\mathcal{C} = \bigcap_{i \in N} \bigcap_{j \in \mathcal{N}_i'} \mathcal{C}_{ij} \tag{5.42}$$

is forward invariant.

Proof. We want to show that if the system starts in a safe set, then switching $V_m \rightarrow V_{lb}$ when $D < \delta$ will still make the system in the safe set. Consider robot A and a consecutive robot B in the charging order, i.e. $\delta_t \leq |T_{a_A} - T_{a_B}| < |T_{a_A} - T_{a_i}|, \quad \forall i \in N \backslash \{A, B\}$. Suppose robot A precedes B in charging and that A starts charging at time t_c, and the switching $V_m \rightarrow V_{lb}$ takes place. Robot A will arrive back to the charging station at time $T_A = t_c + T_{ch} + T_a(V_0, V_{lb})$. Suppose robot B arrives at $t_c + \delta_t$, then it will arrive back to the charging station at time $T_B = t_c + \delta_t + T_{ch} + T_a(V_0, V_{lb})$. The difference in the arrival times $T_B - T_A$ in the new discharging cycle will be equal to $T_B - T_A = \delta_t$. Same argument can be applied iteratively between any two consecutive robots in the charging schedule.

Thus the switching of $V_m \rightarrow V_{lb}$ does not alter the forward invariance of $\mathcal{C}_{ij}$, which can be generalized to all pairwise sets. This means that if the system starts in $\mathcal{C}$, it remains there. □

Remark 10. *Assumption 1 could be relaxed to having a connected undirected graph, and use multi-hop based information dissemination to keep a shared data structure with all agents' arrival times information*[1]*. Such information update has a delay that depends on the level of connectivity of the communication network, i.e. agent i would know about agent j's information after some time $\tau \in (0, \bar{\tau}] \subset \mathbb{R}$. Consequently, the pairwise coordination constraint in* (5.24) *will need to take into account the effect of such delay to ensure proper pairwise coordination [148]. Moreover, the system capacity, as expressed by the limit on temporal separation, will have to be augmented to accommodate the effect of this delay. Such effect of delayed information dissemination is the subject of future work.*

5.7 Results

In this section, we highlight the ability of the proposed method to achieve successful coordination among the robots, as well as its applicability to different types of missions.

For that end, we consider the following tasks:

- Coverage task, in which a group of robots try to expand and cover a certain area of interest.

[1] A similar idea is utilized for example in maximum consensus for decentralized task allocation [147]

- Random exploration, in which robots are randomly wandering around in a certain area while avoiding collision among each other.
- Random exploration with obstacles, which is similar to the previous task, except that the area of interest contains a number of obstacles.

We run simulations for the three scenarios using ARGoS [117], a physics-based simulator designed to handle large numbers of robots, using models of the Khepera IV robot. Moreover, we run hardware experiments using a group of Cognifly quadcopters [149] doing a simple waypoint navigation task.

We use the Buzz programming language [127], which is a language developed for heterogeneous robot swarms, for programming the desired behaviours on robots both in simulation and hardware experiments.

It is worth mentioning that since the final control input resulting from the solution of (5.33) considers the robot as being a single integrator point as described in (5.6), it cannot be used directly on the Khepera IV robots in simulation, as they are better described by a unicycle kinematic model

$$\begin{aligned} \dot{x}_1 &= v\cos\theta \\ \dot{x}_2 &= v\sin\theta \\ \dot{\theta} &= \omega \end{aligned} \tag{5.43}$$

where $\theta \in \mathbb{R}$ is the yaw angle of the robot between its forward direction and the actual x axis of the environment and v is the forward velocity of the robot, and the pair $(v, \omega) \in \mathbb{R}^2$ is the actual input to the robot. We use the nonlinear mapping from single integrator to unicycle kinematics in [150]

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{5.44}$$

where (u_x, u_y) are the components of the control input u in (5.6), and l is the distance of an imaginary handle away from the center between the two wheels, as demonstrated in Figure 5.4, and serves as a tuning parameter for the robot response.

Remark 11. *The choice of the l variable in* (5.44) *can affect the power consumption, as highlighted in [151], which provides an estimate for the value of this parameter to balance between control effort and tracking. In this work, we presume the availability of a conservative estimate of the battery voltage change (e.g. figure 5.1) for a given robot carrying out a certain mission profile using a certain controller. However, for more characterization of the effect of l on power consumption, an energy augmented kinetic model of a differential drive robot, e.g. [152], could be used to find a*

relation between the control actions u_x and u_y and the current consumption, and hence the battery voltage change. We give further elaboration on this point in Appendix A.

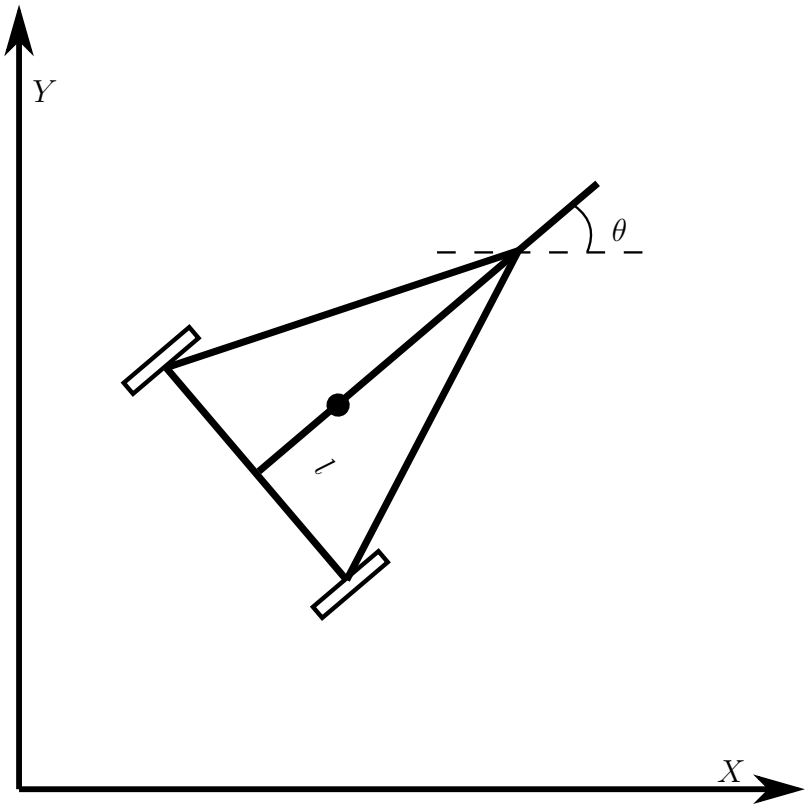
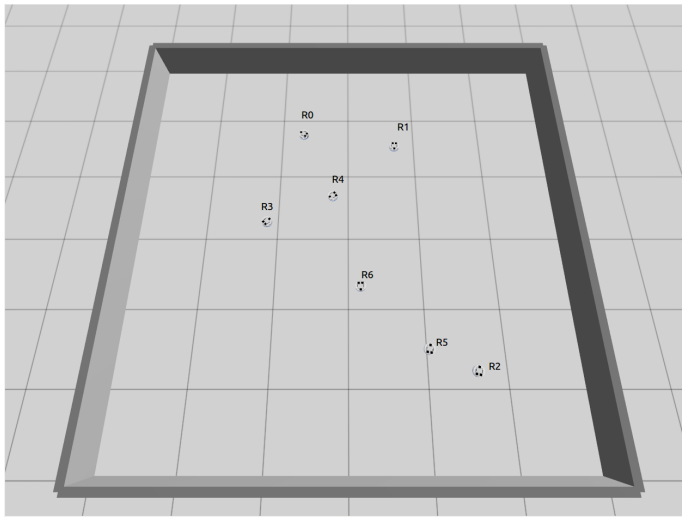


Figure 5.4 Schematic of a unicycle model, as well as the handle point, an l distance away from the center of two weels.

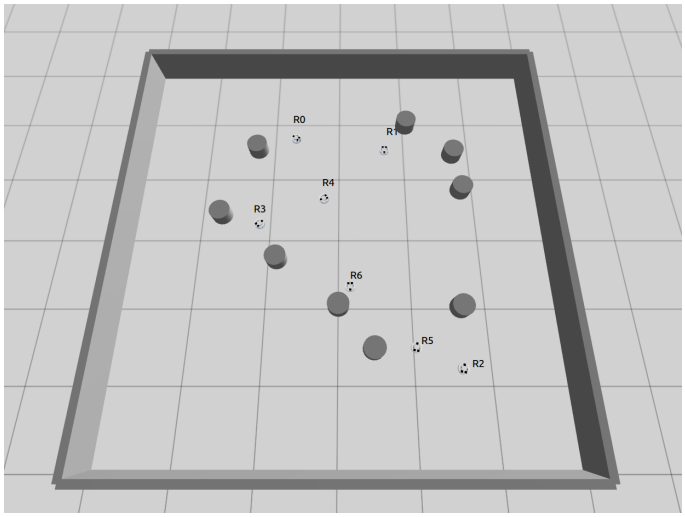
For the simulation cases, we carry out 30 different simulations with different initial positions and orientations for the robots that operate in an arena of 6m×6m that is depicted in Figure 5.5. Moreover, the robots stay for $T_{ch} = 25$ seconds inside the charging region, and their voltage changes in such a way that mimics a battery swapping process.

5.7.1 Coverage task

We applied the proposed energy sufficiency and coordination framework on top of a coverage mission, in a very similar way to what was done in [44]. In this task we have a convex polytope Q divided into a set of polytopes $\mathcal{W} = \{W_1, W_2, \ldots, W_n\}$ with one polytope assigned to each robot and representing its "dominance region", and the robots are required to move in such a way that



(a) Arena without obstacles



(b) Arena with obstacles

Figure 5.5 Snapshots of the arenas used in the simulations, without obstacles in 5.5a, and with obstacles in 5.5b.

minimizes the locational cost function [132]

$$\mathcal{H}(X, \mathcal{W}) = \sum_{i=1}^{n} \int_{W_i} f(||q - x_i||)\phi(q)dq \tag{5.45}$$

where $f(||q - x_i||)$ is a non-decreasing differentiable function $f : \mathbb{R}_+ \to \mathbb{R}_+$ representing the degradation of sensing quality with distance, $\phi(x) : Q \to R_+$ is a distribution density function that represents relative weights or importance over different points in Q. One way to solve this locational optimization problem is to use Voronoi partitions $\mathcal{V} = \{V_1, \ldots, V_n\}$, with a Voronoi

partition V_i defined as

$$V_i = \{x \in X | ||q - x_i|| < ||q - x_j||, \forall j \neq i\} \tag{5.46}$$

so if we let $W_i = V_i$, the solution to (5.45) consists of having each robot chase the centroid of its partition using the following control law

$$u_{cov_i} = k_{cov}(C_{V_i} - x_i) \tag{5.47}$$

where C_{V_i} is the centroid of the Voronoi partition for robot i.

To ensure that a robot will not randomly wander into the charging region before it should, We add a collision avoidance constraint with the following CBF

$$h_{cs} = ||x_i - x_{cs}|| - \Delta_{safe_{cs}} \tag{5.48}$$

where $\Delta_{safe_{cs}}$ should be at least equal to δ from (5.8). We can then calculate the nominal control action u_{nom} through solving the following QP, with the constraints corresponding to (5.48)

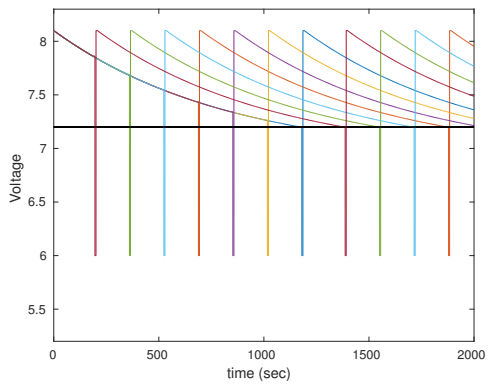
$$\begin{aligned} u_{nom} = \min_{u \in \mathbb{R}^2} \quad & ||u - u_{cov}||^2 \\ \text{s.t.} \quad & \frac{(x_i - x_{cs})}{||x_i - x_{cs}||} u \geq -p_{cs} h_{cs} \end{aligned} \tag{5.49}$$

for $p_{cs} > 0$, and the value of u_{nom} from (5.49) is the nominal control action used in (5.33). The values of the parameters used are presented in Table 5.1.

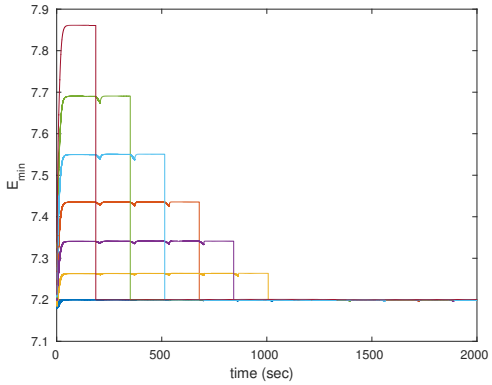
Table 5.1 Values of parameters used in coverage task simulations

Parameter	k_e	k_b	N	V_0	V_{lb}	$\delta_{t_{cr}}$
Value	0.0012	1.1911	7	8.1	7.2	171.3 sec

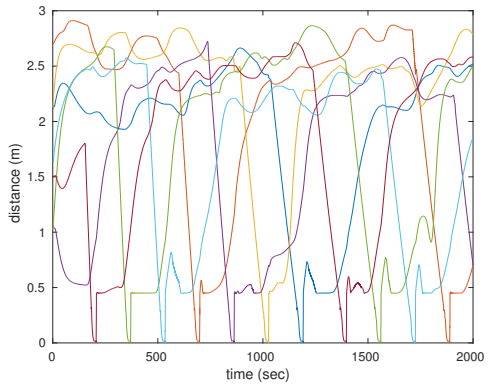
The value of $\delta_{t_{cr}}$ is calculated from (5.38) with $T_{ch} = 25$. For the coverage scenario the robots are tasked with covering the arena depicted in Figure 5.5a, following the nominal control action from (5.49), while achieving a desired temporal separation of $\delta_t = 165$ sec. Figure 5.6 shows a sample result of applying (5.33) on a coverage task. From Figure 5.6a, the desired separation in arrival times is achieved.



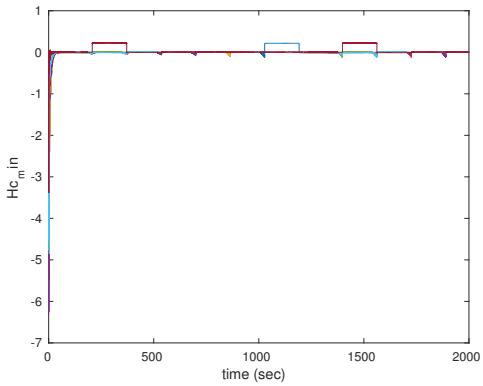
(a) Evolution of robots' voltages



(b) Evolution of robots' V_m values



(c) Distance from the center of the charging region



(d) Minimum value of h_c among all robots

Figure 5.6 Evolution of voltage V and V_m values for the coverage task simulation (5.6a and 5.6b). Figure 5.6c represents the distances of robots from the charging station during the mission. The black horizontal solid line in 5.6a represents V_{lb}. Figure 5.6d is the minimum value of the coodination CBF at all times.

5.7.2 Random exploration task

In this task each robot is required to roam randomly within a certain predefined area surrounded by walls, depicted in Figure 5.5a. Each robot uses a set of onboard ultrasonic sensors to avoid colliding with any of these walls, and thus becomes contained within the area of interest. Figure 5.7 shows a schematic of the obstacle avoidance method we are using. For each sensor registering a distance less than a desired safety range, we form a vector from the robot's origin towards the direction of this sensor in the local reference frame and with a magnitude equal to the registered reading (vectors v_i and v_j in Figure 5.7), we sum together all these vectors to get their resultant (vector V in Figure 5.7) and the escape direction will be the opposite of this resultant vector (vector e in Figure 5.7).

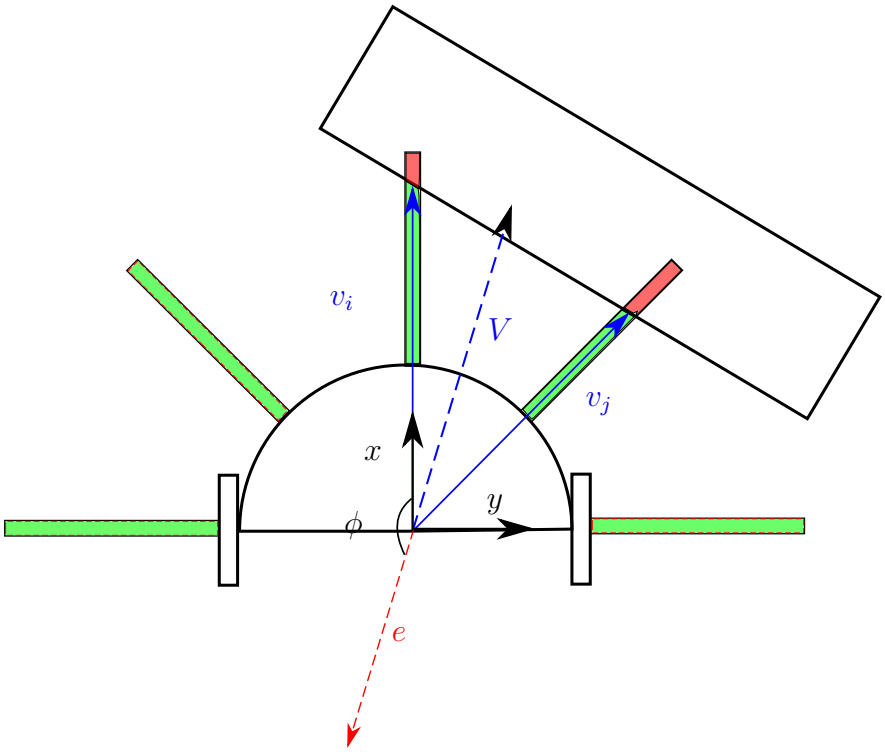


Figure 5.7 Schematic of the method used for avoiding obstacles. Vectors v_i and v_j are vectors in robot's local axes pointing from its origin towards each respective sensor, and the length of each vector is equal to each sensor's registered reading. The vector $V = v_i + v_k$ is the resultant of all the vectors of ultrasonic sensor registering readings and e is the escape vector.

In order to pursue the escape vector, we use a simple proportional controller to calculate the wheel speeds:

$$\begin{aligned} v_r &= v_b + k_{rot}\phi \\ v_l &= v_b - k_{rot}\phi \end{aligned} \tag{5.50}$$

where $k_{rot} > 0$, v_r and v_l are the right and left wheels' linear speeds, v_b is a desired base speed and ϕ is the angle between the escape vector and the local sagittal axis (front direction). We apply a similar collision avoidance layer as (5.49) to prevent random wandering into the charging station, and we depend on the ultrasonic sensors to avoid collision with walls. We use the following CBF for inter-robot collision avoidance

$$h_{col} = ||x_i - x_j|| - \delta_{col} \tag{5.51}$$

where $\delta_{col} > 0$ is a desired safe distance between robots. We calculate the robot's velocity relative to global axes u_{exp}

$$u_{exp} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & l \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{5.52}$$

where $v = \frac{v_r + v_l}{2}$ and $\omega = \frac{v_r - v_l}{b}$ with b being the width of the robot. The nominal control action then becomes

$$\begin{aligned} u_{nom} = \min_{u \in \mathbb{R}^2} \quad & ||u - u_{exp}||^2 \\ \text{s.t.} \quad & \frac{(x_i - x_j)}{||x_i - x_j||} u \geq -p_{col} h_{col}, \forall i \neq j \\ & \frac{(x_i - x_{cs})}{||x_i - x_{cs}||} u \geq -p_{cs} h_{cs} \end{aligned} \tag{5.53}$$

for $p_{col} > 0$ and $p_{cs} > 0$.Table 5.1 presents the values of parameters used in this simulation.

Table 5.2 Values of parameters used in coverage task simulations

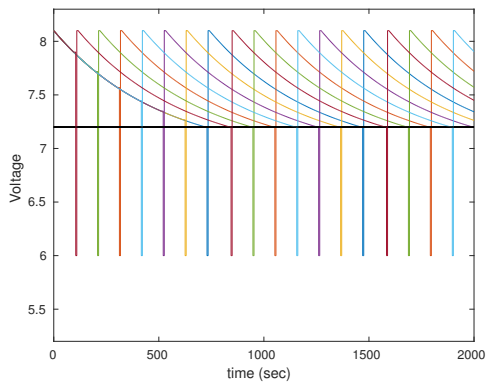
Parameter	k_e	k_b	N	V_0	V_{lb}	$\delta_{t_{cr}}$
Value	0.00195	1.1911	7	8.1	7.2	106.8 sec

Note that the value of $\delta_{t_{cr}}$ is calculated from (5.38) with $T_{ch} = 25$. In this scenario the robots are required to randomly explore (by random walking) the arena depicted in Figure 5.5a, while avoiding collisions and achieving a desired temporal separation of 105 seconds.

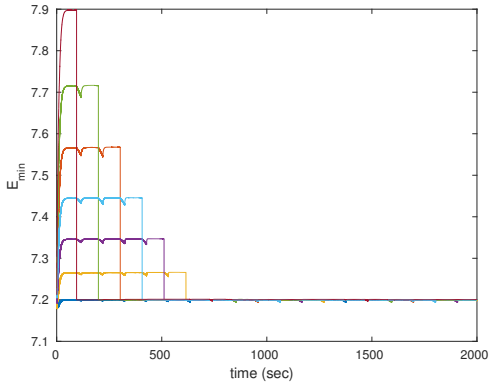
Figure 5.8 shows the results of a sample simulation. We can see from Figure 5.8a that the desired temporal separation in arrival times is achieved. Also similar to previous results, the value of $h_{c_{min}}$ converge to zero as shown in Figure 5.8d.

5.7.3 Random walk with obstacles

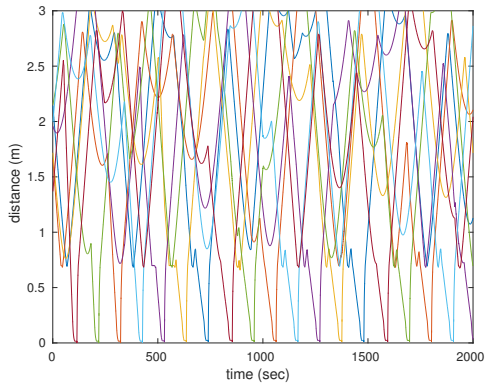
This task is essentially similar to the previous random exploration task discussed above for the same arena, except that the area of interest contains nine randomly distributed cylindrical obstacles, each



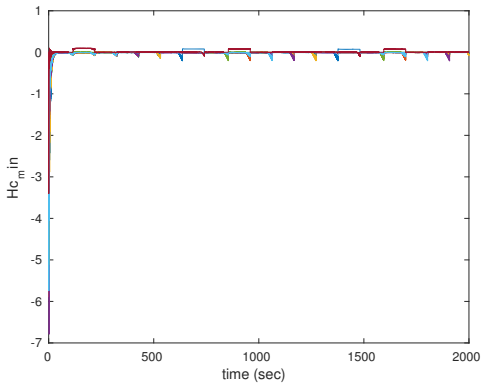
(a) Evolution of robots' voltages



(b) Evolution of robots' V_m values



(c) Distance from the center of the charging region



(d) Minimum value of h_c among all robots

Figure 5.8 Evolution of voltage V and V_m values for the random exploration task simulation (5.8a and 5.8b). The black horizontal solid line in 5.8a represent V_{lb}. Figure 5.8d is the minimum value of the coodination CBF at all times.

with a radius of 15cm (example in Figure 5.5b). We carried out simulations for 30 random initial positions, orientations, and obstacle locations. Although in many cases the framework manages to achieve charging/discharging cycles similar to what is depicted in Figures 5.6 and 5.8, but the presence of obstacles affects the framework negatively which is manifested in not achieving the desired separation in arrival times in many instances.

To illustrate this result, we calculate the differences in arrival times between each two consecutive arrivals for all 30 tests, and then calculate how far these differences are from the desired separation in each case to serve as a measure of the error. Figure 5.9 shows a box plot representing the median, maximum and minimum values of these errors. Ideally, the minimum value of this difference should be at least equal to zero (meaning that in the worst case the desired separation in arrival times is achieved), while in simulation some error could be expected due to the difference between

the point mass model we are using and the actual kinematics of the robot, as well as the fact that the collision avoidance between robots presents some unmodeled delay for the robot's way back to recharge. We point out that these errors are still two orders of magnitude less than the desired temporal separations in each case. The random exploration task with obstacles performs worse than the other two in the worst case due to the introduced disturbance from the environment (the obstacles) that haven't been taken care of in the energy sufficiency framework.

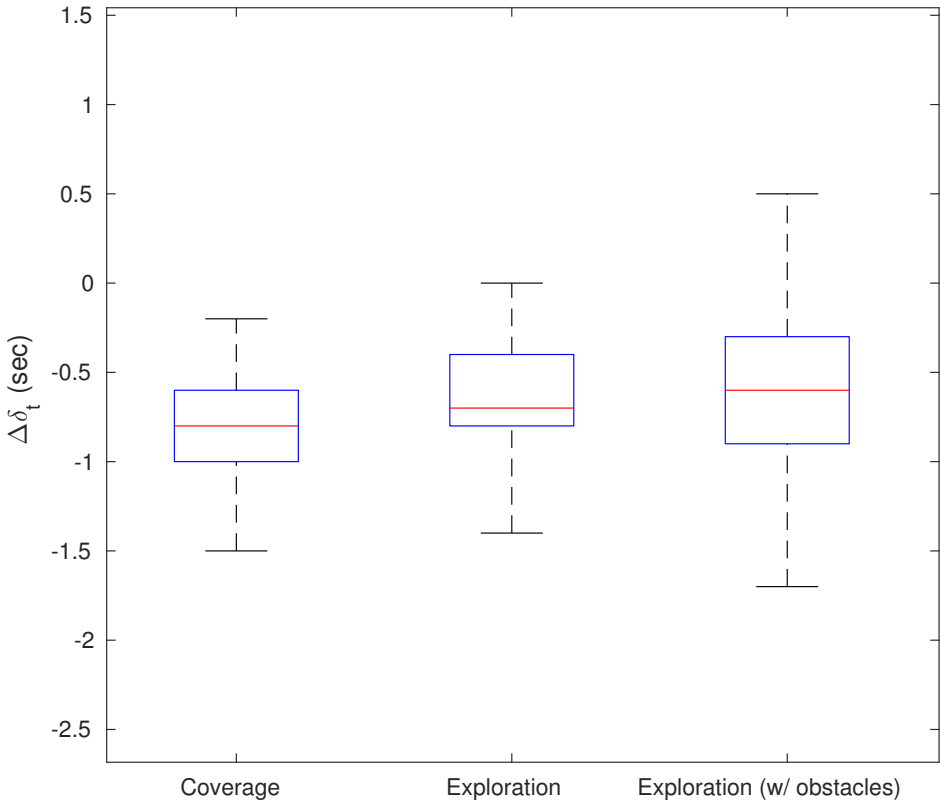


Figure 5.9 Box plots of differences in consecutive arrival times between robots over 30 experiments for the following tasks: (i) coverage, (ii) random exploration and (iii) random exploration with obstacles. A negative difference indicates that two arrival times are separated with a time less than δ_t. The vertical axis represents this error in temporal separation in seconds.

5.7.4 Physical robot experiments

For the hardware setup, we use a group of three Cognifly quadrotors [122] carrying out a simple waypoint navigation task. Such quadrotors can be modelled using a simple single integrator model

and the waypoint navigation is achieved via a proportional controller

$$u_{WN} = k_{WN}(x - x_t) \tag{5.54}$$

where $k_{WN} > 0$ is a proportional gain, and $x_t \in \mathbb{R}^d$ with $d = 2$ in our case.

The drones achieve indoor positional feedback through an Optitrack system, and each drone is running a Buzz [127] code at 10 Hz. Cogniflys are running on Li-po batteries with discharge characteristics similar to that depicted in Figure 5.1. We note that the parameters for the voltage discharge were chosen to bound the actual voltage change from below, i.e. to be conservative. More conservative parameter choice is possible, but it will lead to a more conservative system capacity estimation, i.e. the achievable temporal separation between robots' arrivals will be estimated to be less.

For recharging, the drone stays inside the charging region for $T_{ch} = 65$ seconds, and we apply a manual hot battery swapping process, meaning that the Cognifly is attached to an external power source while the battery is being swapped manually.

The parameters that model the actual system (the battery + mission discharge profile) are presented in Table 5.3. In this task, the drones are required to carry the on waypoint navigation task and

Table 5.3 Values of parameters used in coverage task simulations

Parameter	k_e	k_b	N	V_0	V_{lb}	$\delta_{t_{cr}}$
Value	0.00195	1.1911	7	8.1	7.2	262.3 sec

arrive to the station with a temporal separation of 220 seconds. Figure 5.10 depicts the voltage discharge/recharge cycles and the values of V_m. These voltage values are measured onboard and they are being directly used in the energy sufficiency framework.

5.8 Discussion and Conclusion

In this paper we present a control barrier function (CBF) based framework for extended multi-robot system energy autonomy. We extend the work by [44] considering the case of needing to share a charging station among several robots. We introduce a CBF based coordination layer that is mission-agnostic by nature to ensure mutually exclusive use of a charging station in a minimally invasive way. Moreover, we describe the limits on the temporal separation to be enforced by the coordination framework in such a way that ensures that the safety set for coordination is non-empty.

There are some considerations that may affect the performance of the proposed framework mostly due to some unmodelled aspects. One such example, as discussed in Section 5.7, is the existence

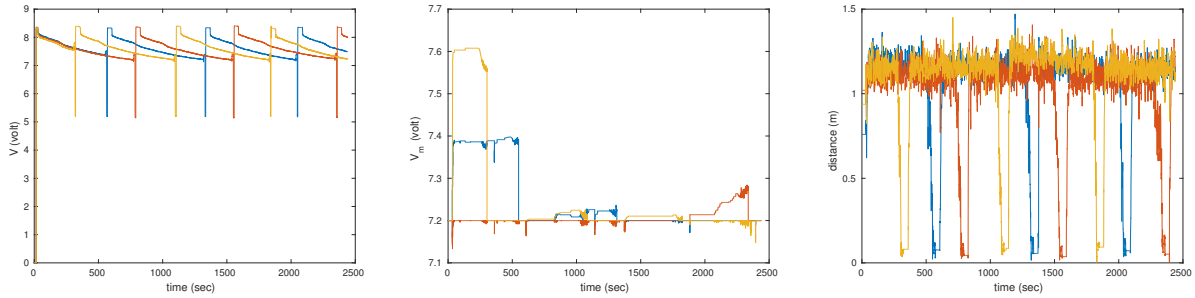
of obstacles in the environment and how the robots evade them, as well as how they evade collision with other robots. This can affect the overall performance because if a robot is hindered by obstacles or evasion of other robots on its way back to the charging station, it can arrive later than expected, which can perturb the differences in arrival times.

Another limitation of this work is the need for knowing a reasonable worst case estimate of battery voltage change depending on the mission under consideration, which makes it more suited for situations where the voltage decrease is uniform throughout the mission. If the power consumption is considerably less than this worst case value during mission execution, the proposed framework will react conservatively to the actual voltage change.

Moreover, the proposed framework uses a simple abstract robot model, namely a single integrator model. The output of the framework is then provided to the actual robot as an input through a proper interface, e.g. equation (5.44). However, this may add an additional layer of power consumption through the actual controllers onboard that pursue the output of our proposed framework, e.g. choosing small values of l in (5.44) may lead to higher power consumption. In Appendix A we present a model that shows the relation between a single integrator to unicycle transformation (5.44) and the voltage change, and we present a method for using our proposed framework on a full unicycle model. However, it would be interesting to use the full nonlinear model without any intermediate simplifying models (as single integrator model) to achieve better utilization.

In the future we think it would be interesting to extend the current work by considering multiple charging stations. Moreover, we need to extend the framework so that it can deal with the currently unmodelled obstacles/environment/terrain and provide energy sufficiency in scenarios where robots move in complex environments by complementing the proposed framework with planning capabilities.

Additionally, we would like to use more accurate robot-specific models, and customize the proposed framework accordingly, to have better ability predicting the voltage change for specific types of robots, thus leading to better utilization of the system.



(a) Evolution of robots' voltages (b) Evolution of robots' V_m values (c) Distance from center of charging region.

Figure 5.10 Evolution of voltage V and V_m values for three Cognifly quadrotors carrying out a waypoint navigation task.

CHAPTER 6 ARTICLE 3 : ENERGY SUFFICIENCY IN UNKNOWN ENVIRONMENTS VIA CONTROL BARRIER FUNCTIONS

Preface: A robot's ability to seek recharge in a complex environment is key for the long-term deployment of robots in realistic environments. In such scenarios, robots often use path planners to find valid paths in an environment represented by a map, either known *a-priori* or constructed by the robot. Therefore there is a need to ensure energy sufficiency over such paths at all times in a mission-agnostic fashion.
In this chapter, we propose a CBF-based framework that provides ES guarantees over paths produced by a general path planner during executing a general mission. The framework we propose is modular and compatible with any form of path planner that is capable of producing waypoints in the environment. Moreover, we provide a method for adapting the proposed framework, which uses a single integrator robot model, to a unicycle-type robot. We demonstrate through simulation results the efficacy of the proposed framework, and we present the results of conducing experiments using the proposed framework on a Scout Mini rover.

Full Citation: Fouad, Hassan, Vivek Shankar Varadharajan, and Giovanni Beltrame. "Energy Sufficiency in Unknown Environments via Control Barrier Functions." arXiv preprint arXiv:2306.15115 (2023). Manuscript submitted to the International Journal of Robotics Research (IJRR) on June 27, 2023.

DOI: `https://arxiv.org/pdf/2306.15115.pdf`

Abstract: Maintaining energy sufficiency of a battery-powered robot system is a essential for long-term missions. This capability should be flexible enough to deal with different types of environment and a wide range of missions, while constantly guaranteeing that the robot does not run out of energy. In this work we present a framework based on Control Barrier Functions (CBFs) that provides an energy sufficiency layer that can be applied on top of any path planner and provides guarantees on the robot's energy consumption during mission execution. In practice, we smooth the output of a generic path planner using double sigmoid functions and then use CBFs to ensure energy sufficiency along the smoothed path, for robots described by single integrator and unicycle kinematics. We present results using a physics-based robot simulator, as well as with real robots with a full localization and mapping stack to show the validity of our approach.

6.1 Introduction

Current advances in robotics and its applications play a key role in extending human abilities and allowing humans to handle arduous workloads and deal with dangerous and uncertain environments. For instance, search and rescue missions [2], construction [7], and mining [4] put a strain on the human body as well as being inherently dangerous. Moreover, tasks with a high degree of uncertainty like terrestrial [17] and extraterrestrial [3] exploration benefit immensely from using robots, especially with the current quest for planetary exploration and the need to discover locations to host humans [153, 154].

To this end, endowing robots with the ability to recharge during a mission is of vital importance to enable long term autonomy and successful execution of missions over extended periods of time. This gives rise to a crucial need for methods that guarantee that no robot runs out of energy mid-mission, i.e. energy sufficiency, while at the same time having the needed flexibility to adapt to various types of missions and environments. Many methods exist in literature to achieve this goal: using static charging stations [44, 155, 156, 157], using moving charging stations that do rendezvous with the robots during their mission [158, 159, 160] or deposit full batteries along robot's mission path [52]. The main shortcomings of these methods are that they either do not provide formal guarantees on performance, or they have limited ability to deal with scenarios involving unstructured and uncertain environments, e.g. in exploration missions where maps are not known beforehand.

One way to tackle the issue of unstructured environments in light of energy sufficiency is to perform path planning that incorporates energy cost as one of its metrics. As an examples of this energy-aware path planning, [161, 162, 163] formulate energy sufficiency as a combinatorial optimization problem with the environment modelled as a weighted graph encoding energy costs, travel times, and distances. One issue with these methods is the rapid increase in computational complexity for large environments. Other methods emerged to deal with this issue with heuristics like Genetic Algorithms [164], Tabu-Search [165] and Monte Carlo Tree Search [166]. However, one fundamental problem with these methods mentioned so far is their need to know the map beforehand, which may not be available for missions with unknown or dynamic environments such as exploration tasks.

Tackling the issue of unknown and unstructured environments calls for the use of exploration planners. Such planners use the collected sensor information over time and provide two types of trajectories: exploration paths that maximize environmental coverage, and homing paths from robot's current position to any desired point in the map that is being incrementally built as the robot keeps exploring. Several well-designed exploration planners exist in literature, many of which developed within the scope of the DARPA SubTerranian Challenge [167]: the Graph-Based exploration

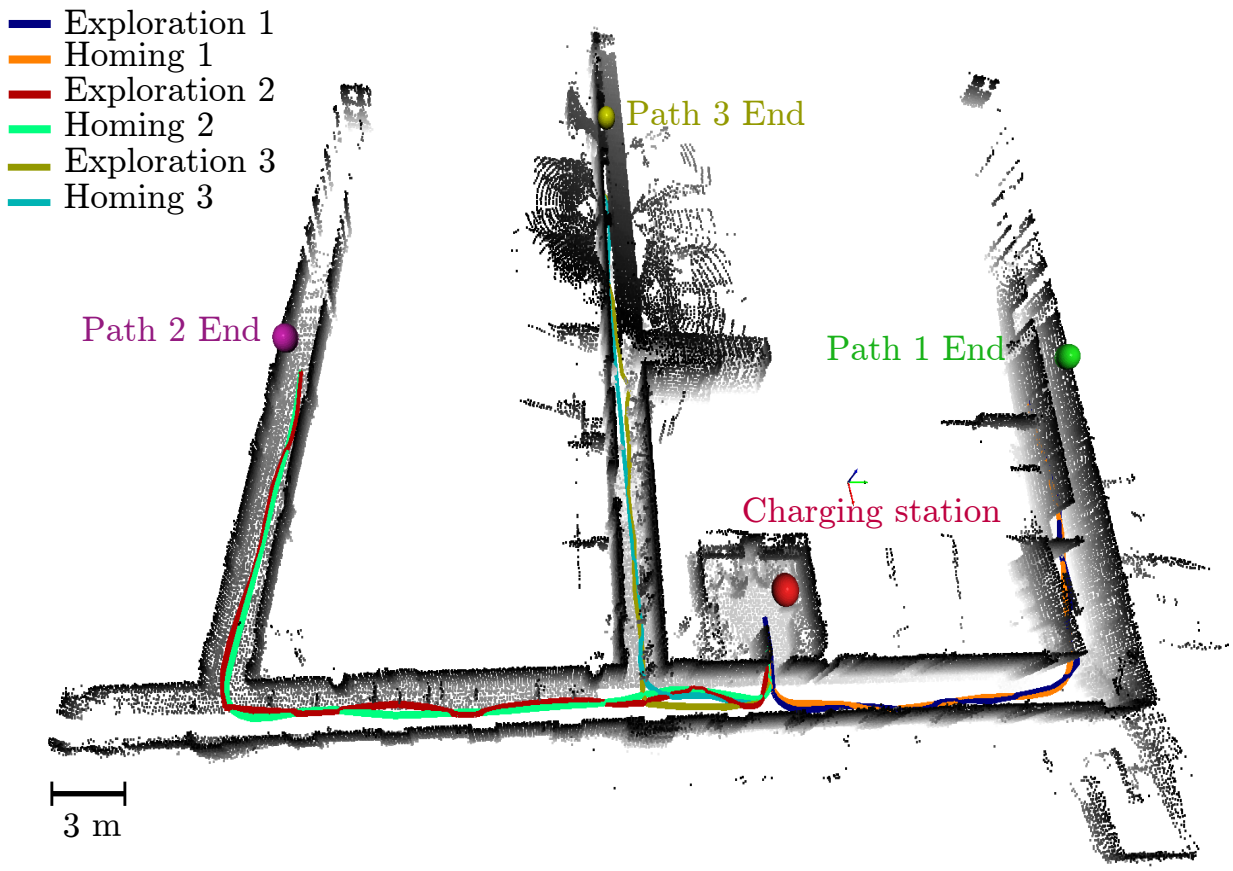


Figure 6.1 Maintaining energy sufficiency during the exploration of a corridor environment.

planner [168], the Next-Best-View planner [169], the motion primitives-based planner [170]), the Dual-Stage Viewpoint Planner [171], and the TARE planner [172].

In this work, we present a modular and mission-agnostic framework that uses a Control Barrier Function (CBF) [47] to guarantee energy sufficiency when applied alongside an arbitrary exploration planner. The approach builds upon our previous work [173, 157], which provides energy-sufficiency guarantees for robots in obstacle-free environments. We thus leverage the ability of an exploration planner to deal with unstructured and unknown environments and extend previous formulations to validate guarantees on energy sufficiency over paths generated by this planner, allowing for more realistic mission execution. The modular nature of our framework makes it suitable for a wide range of applications that employ a path planner, especially the exploration of unknown subterranean environments [168], as well a navigation in urban [174, 42, 175], and indoor environments [176].

In essence, the contribution of this paper is a CBF-based mission-agnostic modular framework that can be applied in conjunction with any path planner to ensure energy sufficiency of a robot

in unknown and unstructured environment. The framework applies to robots modelled as single integrator points or using unicycle kinematics. The framework is validated through physics-based simulation and on a physical AgileX Scout Mini rover, with a detailed description of our hardware setup and software stack (which is also available as open-source).

The paper is organized as follows: Section 6.2 reviews the literature around energy sufficiency, energy awareness in path planning, and some relevant topics to our frameworks like path smoothing and control barrier functions (CBFs); Section 6.3 presents some preliminaries, followed by the problem statement we are addressing; in Section 6.4 we lay out the main building blocks of our framework by addressing a case in which a robot, modelled as a single integrator point, is stationary with a non-changing path; in Section 6.5 we extend the results of the previous Section to the case of a moving robot with a varying path due to robot's motion and path updates; we then present a method for applying our proposed framework with robots described by unicycle kinematics in Section 6.6; Section 6.7 shows simulation and hardware results; then we conclude the paper and provide a discussion along with future work in Section 6.8.

6.2 Related work

Path planning methods for autonomous robots have been an active area of research for a long time [177, 178]. Different families of path planning methods can be found in literature that vary in their purpose (e.g., local planning vs. global planning), the way they encode the environment (grid maps, visibility graphs, voronoi diagrams...), the type of systems they plan for (e.g., holonomic, non holonomic, kinodynamic) and the way the path is created (sampling the space, graph searching, potential fields...).

Endowing path planning with energy awareness has been treated in literature in different forms that vary by purpose. For example, some works find energy efficient paths within an environment so as to increase a mission's life span as presented by [179] for four wheeled robots and by [180] for robots in hilly terrains. Other works focus on ensuring robot's ability to carry out missions within certain energy capacity and return to a charging station. For example, [165] use a graph with nodes representing tasks with energy costs and edges indicating spatial connectivity with distances, then use Tabu search to solve a Traveling Salesperson Problem (TSP) on this graph to minimize cost. [166] use a probabilistic roadmap method to generate a graph with routes to different goals and charging nodes, then they use Monte Carlo Tree Search (MCTS) to create a tour vising all goals, while having a utility function that diverts the robot from its tour to recharge when needed. [181] proposes something similar by creating an idealized version of the environment in the form of a MAKLINK graph, then use the Dijkstra's algorithm for finding paths to charging station. [164] consider the problem of UAV coverage of an area while needing to recharge, and uses a mix of

grid maps and genetic algorithms (GA) to produce trajectories that minimize mission time and cost while penalizing energy loss. In the electric vehicle literature, similar graph representations of environment are typically used, and an optimization problem is solved over the graph. [163] formulate the problem as a variation of the vehicle routing problem and use mixed integer programming to find the optimal paths. Similarly, [162] formulate an integer program that aims to find the best path with least cost to go from destination to goal while charging at a station. The problem is then solved in two stages: building a meta graph of best paths from destination to goal passing through stations, and the then using Dijkstra's algorithm to find the best path of this meta graph. *It is worth noting that these methods typically do not provide performance guarantees.*

The output of sampling based path planners is often in the form of waypoints. There is often a need to smooth the resulting piecewise linear paths to reduce the effect of sharp turns, which gives rise to a significant body of work pertaining to path smoothing [182]: using Bezier curves [183, 184], B-splines [185], among many others. [183] provide a framework for interpolating a set of waypoints with cubic Bezier segments in a way that maintains curvature limits and ensures no collision between obstacles and the interpolated path. [185] uses clamped B-splines to produce C^2 continuous paths, and they provide a scheme for point insertion in segments where there is collision with obstacles to iteratively rebuild the path till no collision takes place.

Another body of work attempts to merge path planning and smoothing: for example, [186] propose a method for finding shortest Bezier paths in a cluttered environment, where Bezier control points are searched for to minimize path length using Genetic Algorithms (GA). [187] provide a method for smoothing the output of variants of an A^* planner by considering the waypoints as Bezier curve control points and then introducing insertion points between every two of these control points, then use quadratic Bezier segments with the inserted points as control points to produce a smooth path. [188] describe a method for creating smooth paths in robot soccer, where the authors use a 4th-order Bezier curve with control points comprised of the robot's position and goal as ends, and the other robots' positions as rest of control points to produce a dynamically changing and smooth path.

We use Control Barrier Functions [47] as the base of our framework. Barrier functions have been used in optimization problems to penalize solutions in unwanted regions of the solution space [144]. This concept has been later exploited to certify the safety of nonlinear systems [63], in the sense that finding such functions guarantees a system's state does not to wander to unsafe regions of the state space. The notion of Control Barrier Function was introduced by [67] to express values of a system's control input that ensures safety for a control affine system, and [146] introduced the popular method of using quadratic programs to merge system tracking, encoded by a desired system input, and the safe control input dictated by CBF constraints. Other methods use Control Lyapunov Barrier Functions (CLBF) [70] to achieve tracking and safety simultaneously.

6.3 Background

6.3.1 Control barrier functions

A control barrier function is a tool that has gained much attention lately as a way of enforcing set forward invariance to achieve safety in control affine systems of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u$$

where $u \in U \subset \mathbb{R}^m$ is the input, U is the set of admissible control inputs, $\mathbf{x} \in \mathbb{R}^n$ is the state of the system, and f and g are both Lipschitz continuous. In this context, what is meant by safety is achieving set forward invariance of some safe set $\mathcal{C}$, meaning that if the states start in $\mathcal{C}$ at $t = t_0$, they stay within $\mathcal{C}$ for all $t > t_0$. This safe set $\mathcal{C}$ is defined as the superlevel set of a continuously differentiable function $h(x)$ in the following manner [47]:

$$\begin{aligned} \mathcal{C} &= \{x \in \mathbb{R}^n : h(x) \geq 0\} \\ \partial\mathcal{C} &= \{x \in \mathbb{R}^n : h(x) = 0\} \\ Int(\mathcal{C}) &= \{x \in \mathbb{R}^n : h(x) > 0\}. \end{aligned} \tag{6.1}$$

This condition can be achieved by finding a value of control input that satisfies $\dot{h} \geq -\alpha(h)$, with $\alpha(h)$ being an extended class $\mathcal{K}$ function [74].

Definition 7. *[47] For a subset $\mathcal{W} \subset \mathcal{C}$, a continuously differentiable function $h(x)$ is said to be a zeroing control barrier function (ZCBF) if there exists a function $\alpha(h)$ s.t.*

$$\sup_{u \in U} L_f h + L_g h u \geq -\alpha(h), \quad \forall x \in \mathcal{W} \tag{6.2}$$

where $L_f h$ and $L_g h$ are the Lie derivatives of $h(x)$ in direction of f and g respectively.

Supposing that we define the set of all safe inputs $U_s = \{u \in U : L_f h + L_g h u \geq -\alpha(h)\}$, then any Lipschitz continuous controller $u \in U_s$ guarantees that $\mathcal{C}$ is forward invariant [47]. Since the nominal control input $u_{nom} \in U$ for a mission may not belong to U_s, there should be a way to enforce safety over the nominal mission input. This could be done by the following quadratic program (QP) [47]

$$\begin{aligned} u^* = \min_{u} \quad & ||u - u_{nom}||^2 \\ \text{s.t.} \quad & L_f h(x) + L_g h u \geq -\alpha(h) \end{aligned} \tag{6.3}$$

noting that u^* tries to minimize the difference from u_{nom}, as long as safety constraints are not violated.

6.3.2 Problem definition

We adopt single integrator dynamics to describe the robot's position in 2D. Moreover, we consider the energy consumed by the robot as the integration of its consumed power, which in turn is a function of the robot's velocity

$$\begin{aligned} \dot{x} &= u \\ \dot{E} &= \mathcal{P}(u) \end{aligned} \tag{6.4}$$

with $x \in \mathbb{R}^2$ being the robot's position , $u \in \mathcal{U} \in \mathbb{R}^2$ is the robot's velocity control action, $E > 0$ is the energy consumed and $\mathcal{P}(u) > 0$ is the power consumed by the robot as a function of its input velocity. The power consumption follows the following parabolic relation

$$\mathcal{P}(u) = m_0 + m_1||u|| + m_2||u||^2 \tag{6.5}$$

for $m_0, m_1, m_2 > 0$. We consider a charging station at $x_c \in \mathbb{R}^2$ and that the robot starts a fast charge or a battery swap sequence as soon as it is at a distance δ away from x_c, i.e. $||x - x_c|| \leq \delta$.

Assume that there exists a path between a robot and a charging station described by a set of way-points $\mathcal{W} = \{w_1, w_2, \ldots, w_{n_w}\}$, with $w_i \in \mathbb{R}^2$ and the charging station at w_{n_w}, produced by a path planner every $\mathcal{T}$ seconds. Provided that such robot is carrying out a mission encoded by a desired control action u_d and a nominal energy budget E_{nom}, our objective is to ensure energy sufficiency for this robot, i.e. $E_{nom} - E(t) \geq 0 \quad \forall t > t_0$, while taking into account the path defined by $\mathcal{W}$ back to the charging station. Such scenario is relevant in cases where ground robots are doing missions in complex or unknown environments, or for flying robots in areas with no fly zones.

In this work we assume that the environment is static, i.e. obstacles don't change their positions during the mission.

Remark 12. *The power model in* (6.5) *is intended to be a generic power model that represents different sources of power consumption like payload power (the constant term), viscous friction (the linear term) and higher order drag forces like drag forces (the parabolic term).*

6.3.3 Strategy overview

Figure 6.2 depicts a schematic representation of the strategy we adopt in this work. We first introduce a path handling layer that consists of a path planner that has access to the environment, either through a known map or through a SLAM system that produces a map for the environment incrementally. The path planner produces a set of discrete waypoints from the robot's position to the CS, and then a path smoothing procedure is carried out on these discrete waypoints to produce a smooth representation of the same path as a function of a path parameter.

We then construct an ES layer that uses the parametric continuous representation of the path and manipulates the robot's position along the path in a manner that ensures that the robot's energy budget is not violated. We start by studying a simple case where both the robot and the path are initially static, then we investigate the effect of robot movement and discrete path changes from the path planner. Moreover, we highlight some stability properties of the proposed framework and use this knowledge to adapt our methods to unicycle type robots.

6.4 Energy sufficiency over a static path

In this section we discuss the foundational ideas of our approach. We start by considering a static scenario where the path does not change, and the robot is stationary and lies at one end of the path, while the charging station lies on the other end.

Briefly, we construct a continuous parametric representation of the piecewise linear path described by waypoints $\mathcal{W}$. We define a reference point along the path that depends on a path parameter value, then we modify the energy sufficiency framework in [157] to manipulate the location of the reference point in a manner proportional to available energy, and we make the robot follow this reference point. This way we can generalize the method in [157] to environments with obstacles.

6.4.1 Smooth path construction

To ensure that the CBFs we are using are Lipschitz continuous, we use a smooth parametric description of the piecewise linear path we receive from a path planner as a set of waypoints $\mathcal{W}$.

We define $p(s)$ to be a point on the path that corresponds to a parameter $s \in [0, 1]$, such that $p(0) = w_1$ and $p(1) = w_n$, i.e. $s = 0$ at the beginning of the path and $s = 1$ at its end. Such concept is common for describing parametric splines like Bezier curves. We seek an expression for $p(s)$ that closely follows a given piecewise linear path with waypoints $\mathcal{W}$.

For some point p lying on the path, we define the path parameter s as being the ratio of the path length from w_1 to p to the total path length. Figure 6.3 shows an illustrative example of five

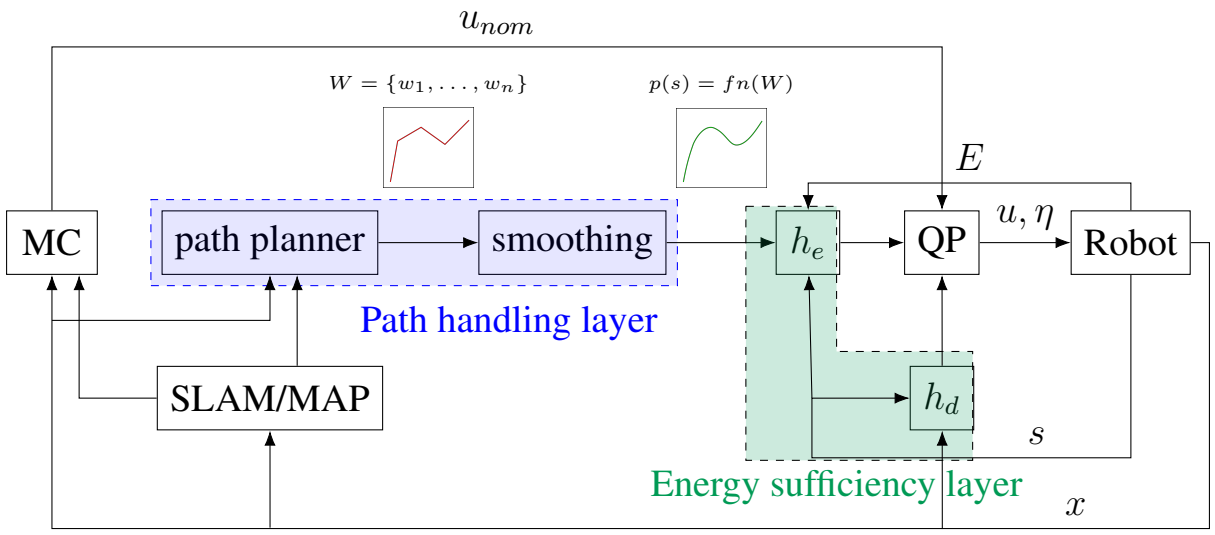


Figure 6.2 A schematic representation of the adopted strategy. A path planner produces discrete waypoints connecting the robot to the CS, provided it has knowledge of the environment's map. The smoothing module produces a parameteric representation of the path, that is in turn used by the CBFs in the ES layer to produce linear constraints for the QP to ensure that the robot's energy budget is not violated through manipulating its position along the path.

waypoints. We adopt a smooth representation for $p(s)$ using double sigmoid activation functions as follows

$$p(s) = \sum_{i=1}^{n_w - 1} \sigma_i(s) \bar{w}_i(s) \tag{6.6}$$

where $\bar{w}_i$ is expressed as

$$\bar{w}_i(s) = \frac{s_{i+1} - s}{s_{i+1} - s_i} w_i + \frac{s - s_i}{s_{i+1} - s_i} w_{i+1} \tag{6.7}$$

Here $s_i = \frac{L_i}{L}$ where $L_i = \sum_{k=1}^{i-1} ||w_{k+1} - w_k||$ and $L = \sum_{k=1}^{n_w - 1} ||w_{k+1} - w_k||$. We note that the relation between the path length $l(s)$ from path start (at w_1) to point $p(s)$ is $l(s) = Ls$ (by definition of s).

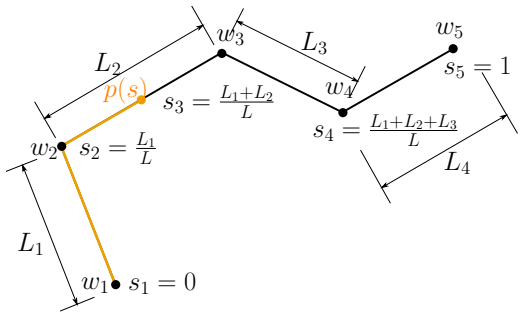


Figure 6.3 Am illustrative example of a path consisting of five waypoints. For a point $p(s)$ on the path s is defined to be the ratio of the length of the orange segment to the total path length. In this illustration $L = \sum_{k=1}^{n_w-1} L_k$

In (6.6), $\sigma_i(s)$ is a double sigmoid function defined as

$$\begin{aligned}
\sigma_i(s) &= \sigma_i^r(s)\sigma_i^f(s) \\
\sigma_i^r(s) &= \frac{1}{1+e^{-\beta(s-(s_i-\epsilon_1))}} \\
\sigma_k^f(s) &= \frac{1}{1+e^{\beta(s-(s_{i+1}+\epsilon_2))}} \\
\epsilon_1 &= \begin{cases} \epsilon, & i = 1 \\ 0, & \text{otherwise} \end{cases} \\
\epsilon_2 &= \begin{cases} \epsilon, & i = n_w - 1 \\ 0, & \text{otherwise} \end{cases}
\end{aligned} \tag{6.8}$$

where $\epsilon > 0$ and the superscripts r and f denote rising and falling edges. The introduction of ϵ_1 and ϵ_2 to the first and last segments in the previous relations is to emphasize that $\sigma_1(0) = 1$ and $\sigma_{n_w-1}(1) = 1$, thus ensuring that $p(0) = w_1$ and $p(1) = w_{n_w}$, otherwise $p(0) = p(1) \approx 0$ which is against the definition of $p(s)$. This idea is illustrated in Figure 6.4. We also note that in any transition region around $s = s_i$ there are two double sigmoid functions involving s_i, namely $\sigma_{i-1}(s)$ and $\sigma_i(s)$. Furthermore, the summation of these functions in the local neighbourhood of $s = s_i$ is equal to one, which follows directly from adding $\sigma_{i-1}^f(s)$ and $\sigma_i^r(s)$

$$\sigma_{i-1}^f + \sigma_i^r = \frac{2 + e^{\beta(s-s_i)} + e^{-\beta(s-s_i)}}{2 + e^{\beta(s-s_i)} + e^{-\beta(s-s_i)}} = 1 \tag{6.9}$$

This idea is highlighted in Figure 6.4. The derivative $\frac{\partial p}{\partial s}$ is

$$\frac{\partial p}{\partial s} = \sum_{i=1}^{n_w-1} \Big(\sigma_i(s) \left(\frac{w_{i+1} - w_i}{s_{i+1} - s_i} \right) + \beta \sigma_i(s) (\sigma_i^f(s) - \sigma_i^r(s)) \bar{w}_i(s) \Big) \tag{6.10}$$

We note that the larger the value of β in (6.8) is the more closely the smooth path described by

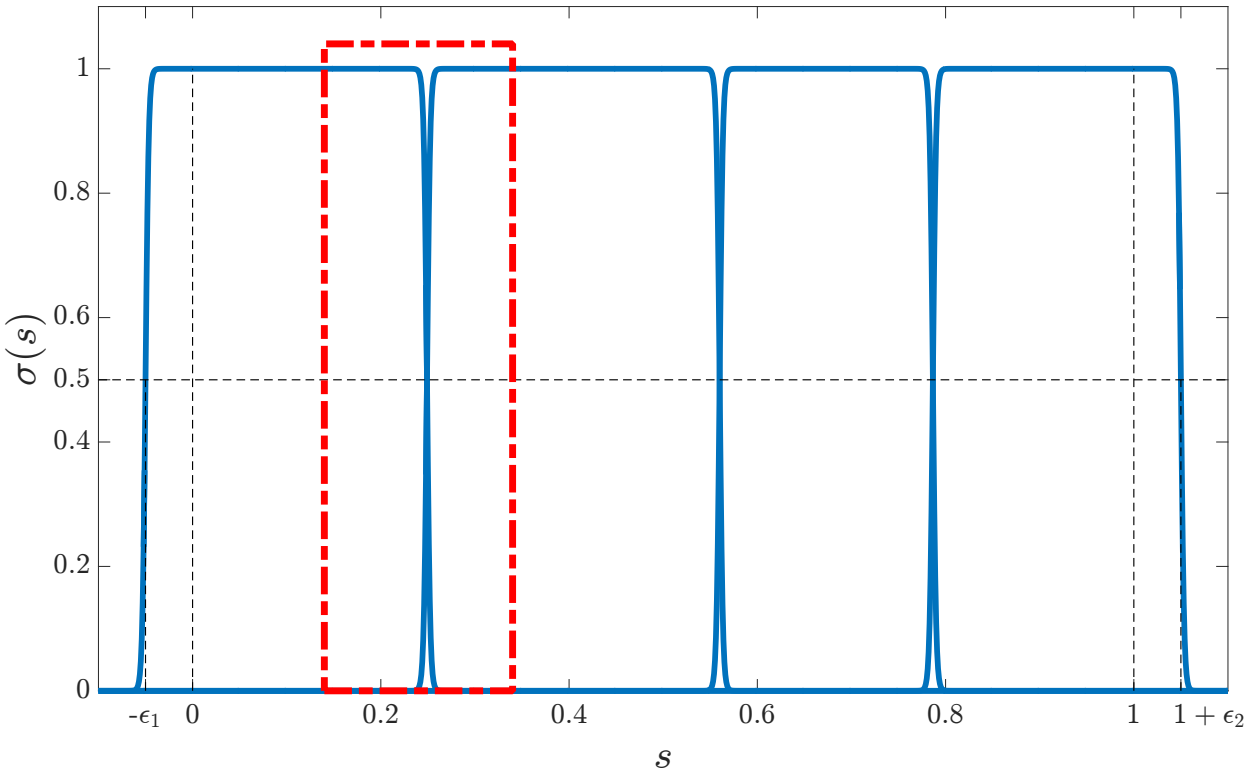


Figure 6.4 Example of double sigmoid functions $\sigma_k(s)$ for the set of five waypoints shown in Figure 6.3. The use of ϵ_1 and ϵ_2 the way described in (6.8) leads to $\sigma_1^r(-\epsilon_1) = 0.5$ and $\sigma_{n_w}^f(1+\epsilon_2) = 0.5$, thus ensuring that $\sigma_1(0) = 1$ and $\sigma_{n_w-1}(1) = 1$. The red rectangle highlights a transition region, and it can be shown that the sum of the two sigmoids involved in this transition is equal to one.

(6.6) follows the piecewise linear path between waypoints in $\mathcal{W}$. Figure 6.5 shows examples of paths at different values of β for the same path depicted in Figure 6.3.

Lemma 11. *For a path described by $p(s)$ in* (6.6)*, with the double sigmoid functions as described in* (6.8) *and provided that $\beta \gg 1$ then the following statement holds*

$$\sum_{i=1}^{n_w-1} \Sigma_i(s) \bar{w}_i(s) \approx 0$$

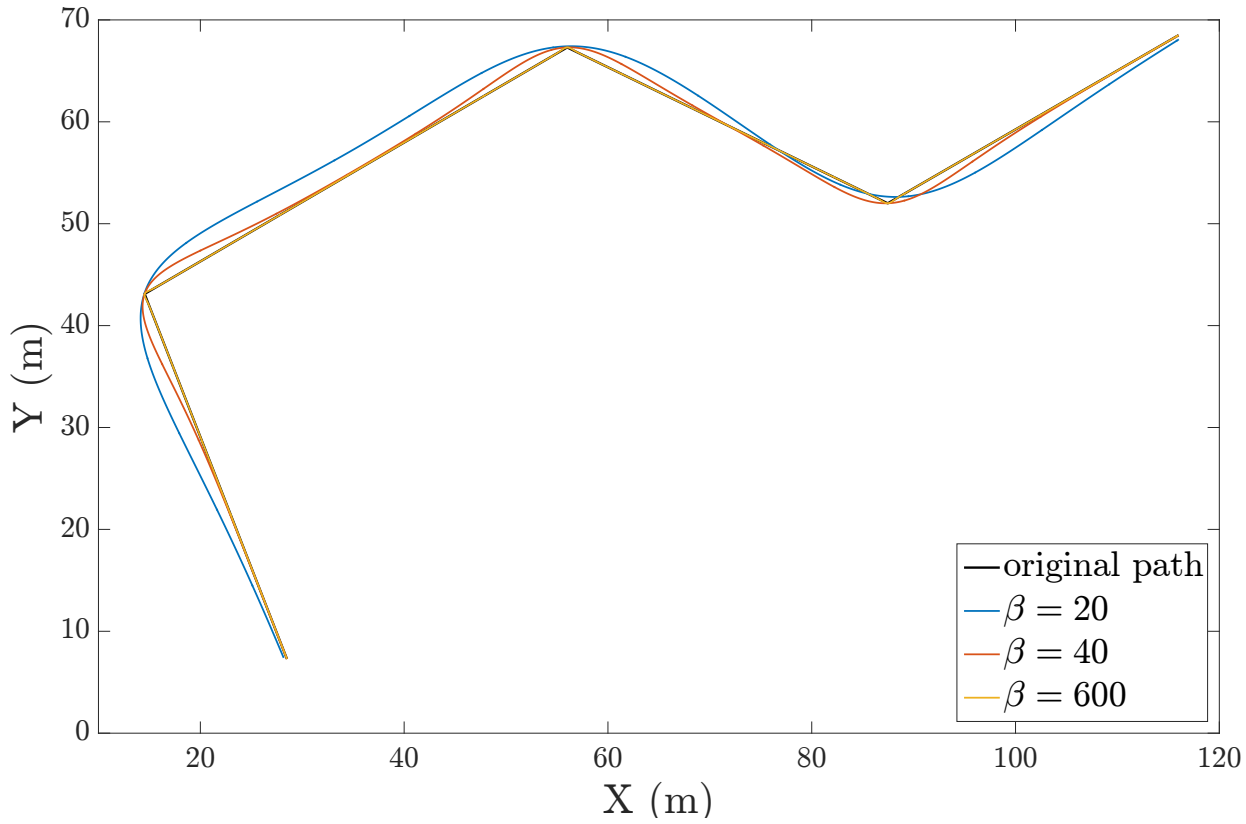


Figure 6.5 Demonstration of the effect of changing the value of β in (6.8) on how closely (6.6) follows the original piecewise linear path.

where $\Sigma_i(s) = \beta\sigma_i(s)(\sigma_i^f(s) - \sigma_i^r(s))$

Proof. At a waypoint w_i we consider the two functions $\Sigma_{i-1}(s)$ and $\Sigma_i(s)$ (both involve $s = s_i$ in their definition) and we note that the values of other Σ are equal to zero by definition. We want to evaluate $D = \Sigma_{i-1}(s)\bar{w}_{i-1} + \Sigma_i(s)\bar{w}_i$ but we do so within a band δ_s around s_i, i.e. at $s' = s + \delta_s$

$$\begin{aligned} D = {} & \beta\sigma_{i-1}(s_i + \delta_s)(\sigma_{i-1}^f(s') - \sigma_{i-1}^r(s'))\bar{w}_{i-1}(s') \\ & + \beta\sigma_i(s_i + \delta_s)(\sigma_{i-1}^f(s') - \sigma_i^r(s'))\bar{w}_i(s') \end{aligned} \tag{6.11}$$

then substituting $\beta \gg 1$ in the last equation we get the following

$$D = \frac{\beta\delta_s e^{-\beta\delta_s}}{(1 + e^{-\beta\delta_s})^2}\left(\frac{w_{i-1}}{s_i - s_{i-1}} + \frac{w_{i+1}}{s_{i+1} - s_i}\right) \tag{6.12}$$

If $\delta_s = 0$ in (6.12) then $D = 0$, and otherwise the quotient $\frac{\beta e^{\beta\delta_s}}{\left(1+e^{\beta\delta_s}\right)^2}$ can be made arbitrarily small by choosing large β. We also note that $\frac{\beta e^{\beta\delta_s}}{\left(1+e^{\beta\delta_s}\right)^2} = \frac{\beta e^{-\beta\delta_s}}{\left(1+e^{-\beta\delta_s}\right)^2}$ meaning the same result follows for $\delta_s > 0$ and $\delta_s < 0$. The statement of the lemma follows by applying the same summation for all values of i. □

6.4.2 Energy sufficiency

We consider the case in which the robot lies at the beginning of the smooth path (6.6) and moves along this path back to the station (at the other end of the path). We assume the path is static, i.e. not changing.

We define a reference point along the path as in (6.6)

$$\begin{aligned} x_r(s) = p(s) &= \sum_{i=1}^{n_w-1} \sigma_i(s) \left(\tfrac{s_{i+1}-s}{s_{i+1}-s_i} w_i + \tfrac{s-s_i}{s_{i+1}-s_i} w_{i+1}\right) \\ \frac{\partial x_r}{\partial s} &= \sum_{i=1}^{n_w-1} \sigma_i(s) \frac{w_{i+1}-w_i}{s_{i+1}-s_i} \end{aligned} \tag{6.13}$$

noting that the derivative expression follows from Lemma 11. We want to control the value of s in a way that makes the reference point approach the end of path in a manner commensurate to the robot's energy content. For this purpose, we introduce the following dynamics for s

$$\dot{s} = \eta \tag{6.14}$$

with $\eta \in \mathbb{R}$ and $s(0) = 0$. The outline of our strategy is as follows: we introduce constraints that manipulate the value of s in a way that makes the reference point x_r approach the end of path as the total energy content decreases, and use an additional constraint to make the robot follow x_r. The candidate CBF for energy sufficiency is

$$h_e = E_{nom} - E - \tfrac{\mathcal{P}(v_r)}{v_r}(L(1-s) - \delta) \tag{6.15}$$

where v_r is the desired velocity with which the robot moves along the path, δ is the distance of the boundary of charging region away from its center, noting that the center of the charging region is w_{n_w}. We note that the expression $L(1-s)$ expresses the length along the path from point $x_r(s)$ till its end. The constraint $\dot{h}_e \geq -\alpha(h_e)$ associated with this candidate CBF is

$$-\mathcal{P}(u) + \tfrac{\mathcal{P}(v_r)}{v_r} L\eta \geq -\gamma_e h_e \tag{6.16}$$

In (6.15) the value of s needs to be maintained above zero (otherwise the value of h_e can be still positive without having the reference point x_r moving back towards the end of the path). For this end we introduce a constraint that lower bounds s with the following candidate CBF

$$h_b = s \tag{6.17}$$

with the associated constraint

$$\eta \geq -\gamma_b h_b \tag{6.18}$$

We complement (6.15) and (6.17) with another candidate CBF that aims at making the robot follow $x_r(s)$ as it changes, and is defined as follows

$$h_d = \tfrac{1}{2}(d^2 - ||x - x_r(s)||^2) \tag{6.19}$$

with $0 < d < \delta$. The constraint associated with this candidate CBF is

$$-(x - x_r(s))^T(u - \dot{x}_r(s)) \geq -\gamma_d h_d \tag{6.20}$$

where $\dot{x}_r = \frac{\partial x_r}{\partial s}\eta$.

In the following lemmas we show that the proposed CBFs lead the robot back to the charging station with $E_{nom} - E \geq 0$. We note that we are not controlling u in (6.16) but rather give this task to (6.20), thus partially decoupling the reference point's movement from the robot's control action. *In other words, we deliberately make the system respond to changing energy levels by moving the reference point along the path without directly changing the robot's velocity.* This interplay between energy sufficiency and tracking constraints is highlighted in the next lemma.

Lemma 12. *For a robot with dynamics described in* (6.4) *and power consumption as in* (6.5)*, and has a maximum magnitude of control action u_{max}, the control barrier functions defined in* (6.15) *and* (6.19) *are zeroing control barrier functions (ZCBF) provided that*

$$v_r^* = \sqrt{\frac{m_0}{m_2}} \leq u_{max}$$

where $||u|| \leq u_{max}$. Moreover, provided that $L(s) > \delta$, then $E = E_{nom}$ only at $L(1 - s) = \delta$.

Proof. The idea of the proof is to show that there is always a value of η that satisfies (6.16) with its $\mathcal{P}(u)$ term, and there is always u to satisfy (6.20) at the same time. Since $\eta \in \mathbb{R}$ means there is always a value of η that satisfies (6.16), thus (6.15) is a ZCBF. However, the reference point x_r could be moving with a speed too fast for the robot to track depending on the value of $\mathcal{P}(u)$.

We consider the critical case of approaching the boundary of the safe set for both h_d and h_e, i.e. $h_e \approx 0$ and $h_d \approx 0$, in which case we can consider the equality condition of the constraints (6.16) and (6.20) (i.e. near the boundary of the safe set the safe actions should at least satisfy $\dot{h} = -\alpha h$ for both (6.16) and (6.20)). The aforementioned constraints become

$$\eta = \frac{\mathcal{P}(u)}{\mathcal{P}(v_r)} \frac{v_r}{L} \tag{6.21a}$$

$$u = \frac{\partial x_r}{\partial s} \eta = \frac{\partial x_r}{\partial s} \frac{\mathcal{P}(u)}{\mathcal{P}(v_r)} \frac{v_r}{L} \tag{6.21b}$$

noting that $x - x_r \neq 0$ when $h_d \approx 0$. We also note that

$$||u|| = \frac{\mathcal{P}(u)}{\mathcal{P}(v_r)} \frac{v_r}{L} \left\| \frac{\partial x_r}{\partial s} \right\|.$$

Assuming $\beta \gg 1$, the derivative $\frac{\partial x_r}{\partial s}$ is as described in (6.6). Moreover,

$$s_{i+1} - s_i = \frac{\sum_{k=1}^{i} \ell_k - \sum_{k=1}^{i-1} \ell_k}{L} = \frac{||w_{i+1} - w_i||}{L} \tag{6.22}$$

where $\ell_k = ||w_{k+1} - w_k||$ and consequently $\frac{\partial x_r}{\partial s}$ can be expressed as

$$\frac{\partial x_r}{\partial s} = L \sum_{i=1}^{n_w - 1} \sigma_i(s) \hat{e}_i \tag{6.23}$$

where $\hat{e}_i = \frac{w_{i+1} - w_i}{||w_{i+1} - w_i||}$ is a unit vector. To estimate $\left\| \frac{\partial x_r}{\partial s} \right\|$ it suffices to mention that in the range $s_i + \epsilon_m < s < s_{i+1} - \epsilon_m$ (for $i = 1, \ldots, n_w - 1$ and $\epsilon_m = \frac{2}{\beta}$) all the double sigmoid functions in (6.23) will be almost equal to zero except for one (by definition) and thus $\left\| \frac{\partial x_r}{\partial s} \right\| = L$. Moreover, if $s_i - \epsilon_m < s < s_i + \epsilon_m$, i.e. s is transitioning from one segment to the next, the sum of the two sigmoid functions locally around $s = s_i$ is equal to one as show in (6.9), meaning that (6.23) will be a convex sum of two unit vectors which will have at most a magnitude equal to one so $\left\| \frac{\partial x_r}{\partial s} \right\| \leq L$. Therefore $||u||$ becomes

$$||u|| = \frac{\mathcal{P}(u)}{\mathcal{P}(v_r)} v_r \tag{6.24}$$

which is a root finding problem for a polynomial of the second degree, since $\mathcal{P}(u)$ is a second order polynomial (6.5). Solving for the roots we get

$$\lambda_1 = v_r, \quad \lambda_2 = \frac{m_0}{m_2 v_r} \tag{6.25}$$

and these roots are equal when $v_r^* = \sqrt{\frac{m_0}{m_2}}$. Since we consider a case where the robot is stationary and the path is fixed, the robot starts from this stationary state and converges to $||u|| = \min(\lambda_1, \lambda_2)$ for a given value of v_r. This means that the maximum achievable return velocity is at $v_r = v_r^*$ where $\lambda_1 = \lambda_2$. If $v_r^* \leq u_{max}$ then there is always a control action u available to satisfy (6.20), rendering (6.19) a ZCBF. If $h_e = 0$ then from (6.15) $E = E_{nom}$ can only happen if $L(1-s) = \delta$, meaning the remaining length along the path is equal to delta, which only happens at the boundary of the charging region. □

Remark 13. *The previous proof assumes the presence of* a-priori *known model for power consumption. However, a mismatch between the power model $\mathcal{P}(u)$ in* (6.5) *and the actual power consumption $\bar{\mathcal{P}}(u)$ will lead to a different solution of* (6.24)*. We are interested in the case where $\bar{\mathcal{P}}(u) = \mathcal{P}(u) + \Delta_p$, with $\Delta_p \in \mathbb{R}$. The root finding problem in* (6.24) *becomes*

$$||u|| = \frac{\bar{\mathcal{P}}(u)}{\mathcal{P}(v_r)} v_r \tag{6.26}$$

and the roots will be

$$\bar{\lambda}_{1,2} = \frac{m_0 + m_2 v_r^2 \pm \mathcal{D}}{2 m_2 v_r} \tag{6.27}$$

where $\mathcal{D} = \sqrt{(m_0 - m_2 v_r^2)^2 - 4 m_2 v_r^2 \Delta_p}$. When $\Delta_p = 0$, $\bar{\lambda}_{1,2} = \lambda_{1,2}$ as described in (6.25)*. If $\Delta_p > 0$, then $\mathcal{D} < (m_0 - m_2 v_r^2)$ and as a result $\bar{\lambda}_1 > v_r$ and $\bar{\lambda}_2 < \frac{m_0}{m_2 v_r}$. In other words, the robot will converge to a faster speed in case the actual power consumption is more than expected, and the converse is true for $\Delta_p < 0$. When*

$$\Delta_p > \Delta_p^* = \left(\frac{m_0 - m_2 v_r^2}{2 v_r \sqrt{m_2}} \right)^2 \tag{6.28}$$

$\mathcal{D}$ becomes undefined and there will be no roots for (6.26)*, indicating a point of instability in velocity for power disturbances beyond Δ_p^*. This idea is illustrated in Figure 6.6.*

Lemma 13. *For a robot with dynamics* (6.4)*, the candidate CBF* (6.17) *is a ZCBF.*

Proof. Since $\eta \in \mathbb{R}$ then there exist a value of η that satisfies (6.18). We need to show that this constraint does not conflict with (6.16) when both constraints are on the boundary of their respective safe sets, i.e. $h_e = h_b = 0$. From (6.16)

$$\eta \geq \frac{\mathcal{P}(u)}{\bar{\mathcal{P}}(v_r)} \frac{v_r}{L} \tag{6.29}$$

while (6.18) becomes $\eta \geq 0$. Since the right hand side of (6.29) is always positive, it means there is always a value of η that satisfies both (6.29) and (6.18), thus (6.17) is a ZCBF. □

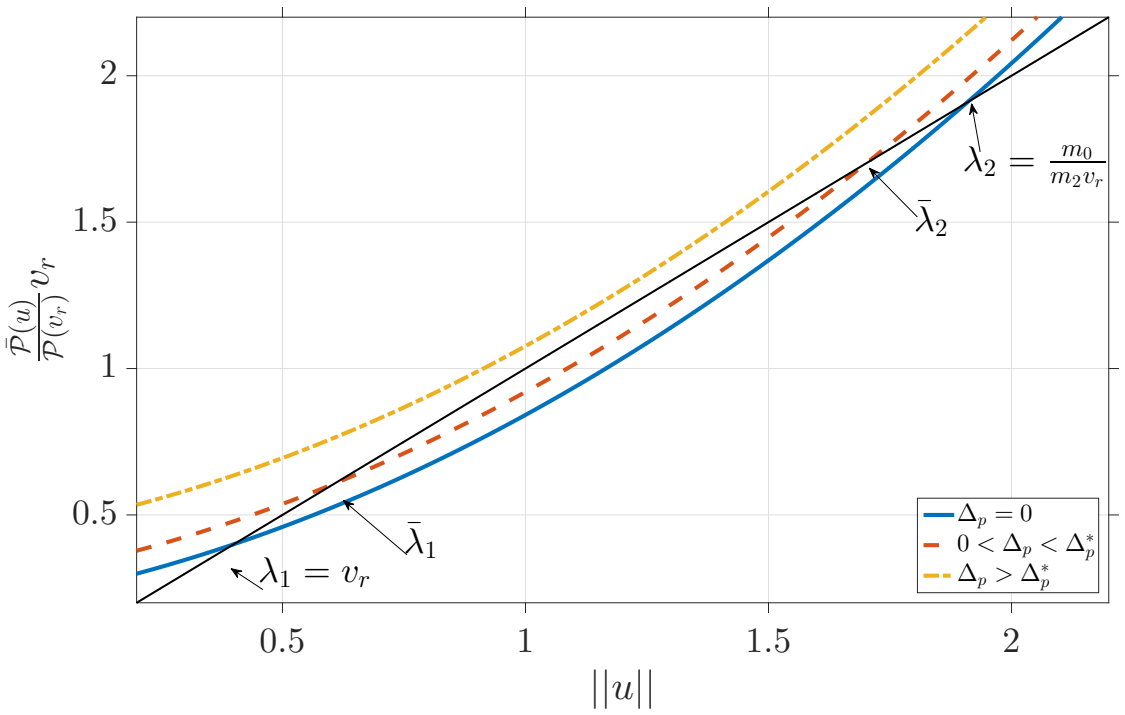


Figure 6.6 Graphical representation for the roots of (6.26) for different values of disturbance power Δ_p. The roots are intersections of the straight line $f_1(u) = ||u||$ in black and the parabolas $f_2(u) = \frac{\bar{\mathcal{P}}(u)}{\mathcal{P}(v_r)} v_r$ (representing RHS and LHS of (6.26) respectively).

Although from Lemma 12 we show that $E(t) = E_{nom}$ on the boundary of the charging region, this is a result that concerns the reference point's position x_r while the robot's actual position tracks x_r through enforcing the constraint (6.19). This situation implies the possibility of x_r reaching a point where $L(s) = \delta$ (boundary of charging region) while the robot's position is lagging behind. In other words, we need the instant where $E(t) = E_{nom}$ to happen inside of the charging region or at least on its boundary.

Proposition 2. *Consider a robot with dynamics* (6.4) *and applying the constraints pertaining to the CBFs* (6.15)*,* (6.17) *and* (6.19)*. We define a modified distance threshold* δ_m *as*

$$\delta_m \leq \delta - d \tag{6.30}$$

then using δ_m *in* (6.15) *ensures that* $E(t)$ *will be at most equal to* E_{nom}*.*

Proof. From Lemma 12 $E = E_{nom}$ only at $L(1-s) = \delta_m$, which is equal to the length of (x_c, x_r) segment in Figure 6.7, i.e. the remaining length along the path from x_r to x_c. This implies that $||x - x_c|| \leq \delta$ as demonstrated in Figure 6.7.

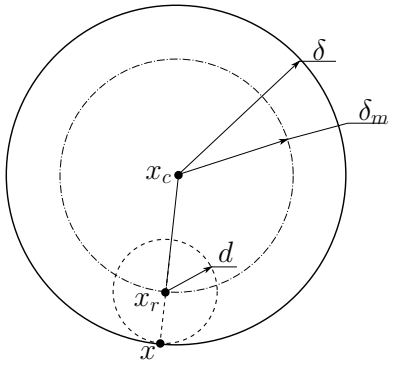


Figure 6.7 Demonstration of x_r pursuing δ_m as the boundary of the charging region in (6.15) while having a robot following the reference point x_r at a distance d away. Here x_r is the reference point position, x_c is the charging station center position, δ is the charging region's radius, and δ_m is a reduced radius to track as described in (6.30).

□

Theorem 6. *For a robot with dynamics described by* (6.4) *and maximum magnitude of control action* u_{max}*, applying the QP in* (6.3) *with constraints* (6.16) *(with* $\delta = \delta_m$ *from* (6.30)*),* (6.20) *and* (6.18)*, and with a static piecewise linear path with waypoints* $\mathcal{W}$*, then energy sufficiency is maintained , i.e.* $E < E_{nom}$ *if* $||x - x_c|| > \delta$.

Proof. If we substitute δ_m from (6.30) in (6.15) and (6.16), then from lemma 12 $E = E_{nom}$ iff $L(1 - s) = \delta_m$, and from proposition 2 this implies that $||x - x_c|| < \delta$ when $E = E_{nom}$, and since E is strictly increasing (due to $\mathcal{P}(u) > 0$ by definition), we conclude that $E \leq E_{nom}$ in $||x - x_c|| > \delta$. □

6.5 Energy sufficiency over a dynamic path

We extend the results from the previous section to consider the case in which the path is changing with time due to robot's movement and replanning actions.

6.5.1 Effect of robot's movement

Assuming that the path is fixed (i.e., there is no replanning), the main difference from the static case is that the first waypoint $w_1 \in \mathcal{W}$ is the robot's position, leading to a change in the total path length L as the robot moves. Additionally, the values of s_i at the different waypoints will change as a result. We therefore consider the following simple proportional control dynamics for w_1:

$$\dot{w}_1 = \xi = -k_w(w_1 - x) \tag{6.31}$$

where $k_w \gg 0$ and $\xi \in \mathbb{R}$. The change in total path length is:

$$\begin{aligned} \dot{L} &= \frac{d}{dt}\left(\sum_{i=2}^{n_w-1} ||w_{i+1} - w_i|| + ||w_2 - w_i||\right) \\ &= -\frac{(w_2 - w_1)}{||w_2 - w_1||}\xi, \end{aligned} \tag{6.32}$$

noting that all the waypoints other than w_1 are fixed. The change in s_i is

$$\dot{s}_i = \frac{d}{dt}\frac{L_i}{L} = \frac{d}{dt}\left(1 - \frac{\bar{L}_i}{L}\right) = \frac{\bar{L}_i}{L^2}\dot{L} \tag{6.33}$$

where $\bar{L}_i$ is the length along the path from waypoint w_i to the end of the path and is constant for $i = 2, \ldots, n_w - 1$. The derivative $\frac{dx_r}{dt}$ is:

$$\frac{dx_r}{dt} = \frac{\partial x_r}{\partial s}\eta + \frac{\partial x_r}{\partial t} \tag{6.34}$$

where $\frac{\partial x_r}{\partial t}$ follows from differentiating (6.6) with respect to time:

$$\frac{\partial x_r}{\partial t} = \sum_{i=1}^{n_w-1} \sigma_i(s)\,(w_{i+1} - w_i)\,\frac{\dot{L}}{L^2}\frac{\bar{L}_i(s_{i+1}-s)+\bar{L}_{i+1}(s-s_i)}{(s_{i+1}-s_i)^2}. \tag{6.35}$$

Consequently, the energy sufficiency constraint (6.16) becomes

$$-\mathcal{P}(u) + \frac{\bar{\mathcal{P}}(v_r)}{v_r}(L\eta - \dot{L}(1-s)) \geq -\gamma_e h_e \tag{6.36}$$

and the tracking constraint (6.20) now uses $\dot{x}_r$ as in (6.34).

The results from Theorem 6 rely on the fact that the path is static. To use the same result in the dynamic case we "freeze" the path when the robot needs to go back to recharge, i.e. we stop w_1 from tracking robot's position when it needs to go back to recharge:

Proposition 3. *Consider a robot with dynamics* (6.4) *applying the proposed energy sufficiency framework described by the CBFs* (6.15) *and* (6.17)*. Consider the following dynamics for* w_1

$$\dot{w}_1 = \xi = -k_w(w_1 - x)\,(1 - \zeta(s)) \tag{6.37}$$

where ζ *is an activation function defined as*

$$\zeta(s) = \begin{cases} 0 & s \leq \epsilon_a \\ 1 & \textit{otherwise} \end{cases} \tag{6.38}$$

with $0 < \epsilon_a \ll \bar{\epsilon}_a < 1$ *and* $||w_1 - x_r(\bar{\epsilon}_a)|| = d$*, then* (6.15) *and* (6.19) *are ZCBF.*

Proof. We start by noting that (6.37) achieves tracking of the robot's position in case $s = 0$, with an error inversely proportional to k_w, according to the candidate Lyapunov function

$$V = \tfrac{1}{2}(x - w_1)^T(x - w_1) \tag{6.39}$$

which has the derivative

$$\dot{V} = (x - w_1)^T u - k_w||x - w_1||^2$$

and this means that $\dot{V} \leq 0$ under high value of k_ω (which is feasible since ξ is an imaginary point with no physical characteristics). Moreover, since $\eta \in \mathbb{R}$ then there is a value of η capable of satisfying the following inequality

$$\eta \geq \frac{1}{L}\left(\tfrac{\mathcal{P}(u) - \gamma_e h_e}{\mathcal{P}(v_r)} v_r + \dot{L}(1 - s)\right). \tag{6.40}$$

We need to show that if $0 < s < \epsilon_a$ (when the reference point starts moving but the path freezing has not been activated yet, according to (6.38)), tracking and energy sufficiency constraints are not violated as well as that s increases so that $s > \epsilon_a$.

To prove the latter, we need to show that the right hand side of (6.40) is positive when $h_e \approx 0$, i.e. near the boundary of energy sufficiency safe set. The term $\frac{\mathcal{P}(u)}{\mathcal{P}(v_r)} v_r > 0$ by definition, so the sign of the right hand side of (6.40) depends on sign of $\dot{L}$. It can be shown that the sign of $\dot{L}(1-s)$ depends on the sign of $\frac{d}{dt}||w_1 - w_2||$, therefore even if $\frac{d}{dt}||w_1 - w_2|| < 0$, it will be so until $||w_1 - w_2|| \approx 0$, when the right hand side of (6.40) will be positive. Therefore, when $h_e \approx 0$, $\eta > 0$, meaning s will increase even when $0 < s < \epsilon_a$.

As a result, the reference point x_r moves along the path and h_d in (6.19) approaches zero (in the limit case $||w_1 - x_r|| = d$ at $s = \bar{\epsilon}_a$). The fact that $s(t)$ is continuous and $\epsilon_a \ll \bar{\epsilon}_a$ implies that $\dot{w}_1 = 0$ before $s(t) = \bar{\epsilon}_a$, i.e. the path freezes while $h_d > 0$, so the path freezing condition (6.38) does not violate the tracking CBF h_d, nor the energy sufficiency CBF h_e, and consequently the result of lemma 12 follows (since $h_e \approx 0$ and $\eta > 0$ implying $s > 0$ leading to $h_d \approx 0$).

□

The full QP problem with the constraints discussed so far can be expressed as

$$\begin{aligned} \mathbf{u}^* = \min_{\mathbf{u}} \quad & ||\mathbf{u} - \mathbf{u}_{nom}||^2 \\ \text{s.t.} \quad & \mathbf{A}\mathbf{u} \geq \mathbf{B} \end{aligned} \tag{6.41}$$

where

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \frac{\mathcal{P}(v_r)}{v_r} L & \mathbf{0}_{1\times 2} \\ 1 & \mathbf{0}_{1\times 2} \\ (x-x_r)^T \frac{\partial x_r}{\partial s} & -(x-x_r)^T \end{bmatrix} \\
\mathbf{B} &= \begin{bmatrix} -\gamma_e h_e + \mathcal{P}(u) + \dot{L}(1-s) \\ -\gamma_b h_b \\ -\gamma_d h_d \end{bmatrix} \\
\mathbf{u}_{nom} &= \begin{bmatrix} 0 & u_{nom} \end{bmatrix}
\end{aligned} \tag{6.42}$$

Theorem 7. *For a robot described by* (6.4) *with a set of ordered waypoints* $\mathcal{W} \in \mathbb{R}^{n_w \times 2}$*,* (6.41) *ensures energy sufficiency.*

Proof. Since (6.15) and (6.19) are valid ZCBFs from Proposition 3 and (6.17) is a valid ZCBF then from Lemma 13, then from Proposition 2 and Lemma 12 (augmented by Proposition 3) $E(t) = E_{nom}$ at $||x - x_c|| < \delta$ (inside the charging region), and since $E(t)$ is strictly increasing it means $E(t) < E_{nom}$ for $||x - x_c|| > \delta$, as shown by Theorem 6, i.e. energy sufficiency is maintained. □

6.5.2 Effect of path planning

During the course of a mission, the path planner keeps updating the waypoints back to the charging station every $\mathcal{T}$ seconds, meaning that there are discrete changes in the number of waypoints and their locations, which can lead to violating the energy sufficiency constraint. To account for these changes, we impose some conditions on the output of the path planner so as not to violate other constraints.

Definition 8. *Assuming there is a path* $\mathcal{W}^{(k-1)\mathcal{T}} = \{w_1^{(k-1)\mathcal{T}}, \ldots, w_{n_w}^{(k-1)\mathcal{T}}\}$ *at time* $(k-1)\mathcal{T}$ *between a robot at position* $x(k\mathcal{T}) = w_1^{(k-1)\mathcal{T}}$ *and the charging station, Sequential Path Construction (SPC) is the process of creating a new set of waypoints* $\mathcal{W}^{k\mathcal{T}} = \{w_1^{k\mathcal{T}}, \ldots, w_{n_w+1}^{k\mathcal{T}}\}$ *at time* $k\mathcal{T}$ *provided that* $\zeta(s) = 0$*, where* $\zeta(s)$ *is defined in* (6.38)*, such that*

$$\begin{aligned}
w_1^{k\mathcal{T}} &= x(k\mathcal{T}) \\
w_2^{k\mathcal{T}} &= \kappa w_1^{k\mathcal{T}} + (1-\kappa) w_3^{k\mathcal{T}} \\
w_{i+1}^{k\mathcal{T}} &= w_i^{(k-1)\mathcal{T}}, \quad i = 2, \ldots, n_w
\end{aligned} \tag{6.43}$$

where $0 \ll \kappa < 1$.

Lemma 14. *Sequential Path Construction is path length and path angle invariant, meaning the following two equations are satisfied*

$$\begin{aligned}\sum_{i=1}^{n_w-1} ||w_i^{(k-1)\mathcal{T}} - w_{i+1}^{(k-1)\mathcal{T}}|| &= \sum_{i=1}^{n_w} ||w_i^{k\mathcal{T}} - w_{i+1}^{k\mathcal{T}}|| \\ \sum_{i=2}^{n_w-1} |\psi_i^{(k-1)\mathcal{T}}| &= \sum_{i=2}^{n_w} |\psi_i^{k\mathcal{T}}|\end{aligned} \tag{6.44}$$

where

$$\psi_i = \cos^{-1} \frac{\Delta w_{i-1}^i . \Delta w_i^{i+1}}{||\Delta w_{i-1}^i|| . ||\Delta w_i^{i+1}||} \tag{6.45}$$

and $\Delta w_i^{i+1} = w_{i+1} - w_i$

Proof. The path length at time $k\mathcal{T}$ is

$$\begin{aligned} L^{k\mathcal{T}} &= \sum_{i=1}^{n_w} ||w_i^{k\mathcal{T}} - w_{i+1}^{k\mathcal{T}}|| \\ &= ||w_1^{k\mathcal{T}} - w_2^{k\mathcal{T}}|| + ||w_2^{k\mathcal{T}} - w_3^{k\mathcal{T}}|| + \sum_{i=3}^{n_w} ||w_i^{k\mathcal{T}} - w_{i+1}^{k\mathcal{T}}|| \\ &= ||w_1^{k\mathcal{T}} - w_3^{k\mathcal{T}}|| + \sum_{i=2}^{n_w} ||w_i^{(k-1)\mathcal{T}} - w_{i+1}^{(k-1)\mathcal{T}}|| \\ &= ||w_1^{(k-1)\mathcal{T}} - w_2^{(k-1)\mathcal{T}}|| + \sum_{i=2}^{n_w} ||w_i^{(k-1)\mathcal{T}} - w_{i+1}^{(k-1)\mathcal{T}}|| \\ &= L^{(k-1)\mathcal{T}}. \end{aligned} \tag{6.46}$$

By definition we have: $\psi_i^{(k-1)\mathcal{T}} = \psi_{i+1}^{k\mathcal{T}}$ for $i = 2, \ldots, n_w$. Since $w_3^{k\mathcal{T}} - w_2^{k\mathcal{T}} = (1-\kappa)(w_3^{k\mathcal{T}} - w_1^{k\mathcal{T}})$ and $w_2^{k\mathcal{T}} - w_1^{k\mathcal{T}} = \kappa(w_3^{k\mathcal{T}} - w_1^{k\mathcal{T}})$, then $\cos\psi_2^{k\mathcal{T}} = 1$ therefore $\psi_2^{k\mathcal{T}} = 0$. In conclusion $\sum_{i=2}^{n_w-1} |\psi_i^{(k-1)\mathcal{T}}| = \sum_{i=2}^{n_w} |\psi_i^{k\mathcal{T}}|$. □

Proposition 4. *Sequential Path Construction does not violate energy sufficiency, provided* $\mathbf{u}_{nom}$ *is Lipschitz.*

Proof. We need to show that changing the path with SPC does not affect the following two inequalities if they are satisfied for the original path:

$$h_e = E_{nom} - E - \tfrac{\mathcal{P}(v_r)}{v_r}(L(1-s) - \delta_m) \geq 0 \tag{6.47a}$$

$$-\mathcal{P}(u_{nom}) + \tfrac{\mathcal{P}(v_r)}{v_r}\left(L\eta_{nom} - \dot{L}(1-s)\right) \geq -\gamma_e h_e \tag{6.47b}$$

Since η_{nom} and u_{nom} are continuous, then $\mathcal{P}(u_{nom})$, E and s are all continuous with no jumps (i.e., discrete changes). Since path length is invariant under SPC by virtue of Lemma 14, then L in (6.47) does not change as well as $\dot{L}$, meaning that (6.47) is not violated under SPC. □

Based on the proposition above, we introduce the Algorithm 2 that admits new paths produced by the path planner as long as they do not violate (6.47), and otherwise switches to SPC.

Algorithm 2 Admission of new path at time $k\mathcal{T}$

Require: $\mathcal{W}^{(k-1)\mathcal{T}}$,$\mathcal{W}^{k\mathcal{T}}_{candidate}$, x,$\mathbf{u}_{nom}$
1: $L, h_e \leftarrow$ EVALUATE_PATH($\mathcal{W}^{k\mathcal{T}}_{candidate}, \mathbf{u}_{nom}$)
2: **if** $\zeta(s) == 0$ **then**
3: **if** (6.47a) == False OR (6.47b) == False **then**
4: $\mathcal{W}^{k\mathcal{T}} \leftarrow$ SPC_PATH($x(k\mathcal{T}), \mathcal{W}^{(k-1)\mathcal{T}}$)
5: **else**
6: $\mathcal{W}^{k\mathcal{T}} \leftarrow \mathcal{W}^{k\mathcal{T}}_{candidate}$
7: **end if**
8: **end if**
9: **return** $\mathcal{W}^{k\mathcal{T}}$

In Algorithm 2, EVALUATE_PATH($\mathcal{W}^{k\mathcal{T}}, \mathbf{u}_{nom}$) is a function that takes the candidate path points from the path planner and evaluates the path length, as well as the value of h_e, and SPC_PATH($x, \mathcal{W}$) is a function that updates a path using SPC, given a robot's position x and a current path $\mathcal{W}$.

Theorem 8. *For a robot described by* (6.4) *that applies the control strategy in* (6.41) *for an already existing set of waypoints $\mathcal{W}^{(k-1)\mathcal{T}}$ at time $(k-1)\mathcal{T}$, $k \in \mathbb{N}$. Suppose a path planner produces a candidate set of waypoints $\mathcal{W}^{k\mathcal{T}}$ at time $k\mathcal{T}$ that satisfies the conditions in* (6.47) *and provided that $\zeta(s) = 0$ from* (6.38)*, then Algorithm 2 ensures energy sufficiency is maintained.*

Proof. If the new set of augmented waypoints satisfies (6.47a), then switching from $\mathcal{W}^{(k-1)\mathcal{T}}$ to $\mathcal{W}^{k\mathcal{T}}$ does not violate the energy sufficiency constraint encoded by h_e. Moreover, if said switching satisfies (6.47b), then $h_e > 0$ is satisfied with η_{nom} at $s = 0$, meaning that w_1 tracks x as outlined in Proposition 3 and consequently x_r tracks the robot's position (since $s = 0$ and w_1 tracks robot's position), thus $h_d > 0$ is not violated as well, which means the sufficiency and tracking constraints are not violated by the path update.

When $s > 0$, i.e. $\zeta(s) \neq 0$, the path is frozen and energy sufficiency is maintained by virtue of Theorem 7. If either condition in (6.47) is violated, the path is updated using SPC which maintains energy sufficiency as discussed in Proposition 4. Therefore Algorithm 2 ensures energy sufficiency and tracking constraints are not violated.

□

6.6 Application to unicycle-type robots

The method described so far uses a single integrator model to describe robot dynamics. Although such model choice is widely used in robotics and has the advantage of versatility [189], applying it directly to more specific robot models needs proper adaptation, especially considering the effects of unmodelled modes of motion on power consumption. In this section we describe a method to apply the proposed framework on a non-holonomic wheeled robot, which has the added characteristic of being able to spin. More specifically we are interested in robots with the following unicycle kinematic model

$$\begin{aligned} \dot{x}_1 &= v\cos\theta \\ \dot{x}_2 &= v\sin\theta \\ \dot{\theta} &= \omega \end{aligned} \tag{6.48}$$

where $x = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T \in \mathbb{R}^2$ is robot's position, $\theta \in \mathbb{R}$ is its orientation, $v \in \mathbb{R}$ and $\omega \in \mathbb{R}$ are the linear and angular speeds respectively which act as inputs. The speed of a single integrator model u from (6.4) can be transformed to linear and angular speeds for a unicycle through the following relation [150]:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\ell} \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} u \tag{6.49}$$

where $\ell > 0$ is a distance from the robot's center to an imaginary handle point. We also choose to be able to move backward when the value of v becomes negative by doing the following

$$\begin{aligned} v' &= v \\ \omega' &= \omega \frac{v}{|v|} \end{aligned} \tag{6.50}$$

A robot described by (6.49) consumes additional power due to its angular speed ω in addition to what is consumed by its linear speed v, which calls for augmenting (6.5) with additional terms

$$\mathcal{P}_u(v,\omega) = m_{u_0} + m_{u_1}|v| + m_{u_2}|v|^2 + m'_{u_1}|\omega| + m'_{u_2}|\omega|^2 \tag{6.51}$$

We note that in this power model we assume no direct coupling effects between linear and angular speeds on power consumption.

Remark 14. *The power model in* (6.51) *is an extension of* (6.5)*, except that it adds the viscous damping and higher order drag effects due to the rotation speed input ω (linear and parabolic terms in ω respectively). Since the power model in* (6.51) *is different from that in* (6.5)*, we seek to*

establish a relation between the two. When a robot is moving in a straight line, then $\mathcal{P}_u(v,0) = \mathcal{P}(u)$. *From* (6.49) $||u|| = \sqrt{v^2 + (\omega\ell)^2}$ *so when* $\omega \neq 0$, v *decreases for the same* $||u||$, *meaning* $\mathcal{P}_u(v,0) \leq \mathcal{P}_u(v,\omega)$. *In this case we either have* $\mathcal{P}(u) > \mathcal{P}_u(v,\omega)$ *meaning that turning has no contribution to power consumption, or* $\mathcal{P}(u) \leq \mathcal{P}_u(v,\omega)$ *which means turning has significant contribution in power consumption. We are interested in the latter case and we consider that*

$$\mathcal{P}_u(v,\omega) \leq \mathcal{P}(u) + \Delta_\omega \tag{6.52}$$

where $\Delta_\omega \in \mathbb{R}$ *is the change in power due to rotation.*

Using a power model that only accounts for linear speed is akin to having the power consumption due to ω as a disturbance power Δ_ω, which may lead to instability as discussed in Remark 13. A solution to this issue is choosing a fairly slow return speed value v_r so as to increase the stability margin Δ_p^* in (6.28), however this may impose undesirable limitations on performance.

Since the path we are using is essentially a piecewise linear path with waypoints $w_i, i = 1, \ldots, n_w$, robot spinning will be mostly near these waypoints when it is changing its direction of motion. The idea behind our proposed adaptation is to add a certain amount of power $\tilde{\delta}$ to $\mathcal{P}(v_r)$ in (6.26) near the path's waypoints so that roots of (6.26) always exist. In other words, if we define $\tilde{\mathcal{P}}(v_r) = \mathcal{P}(v_r) + \tilde{\delta}$, we ensure that roots for the following equation always exist

$$||u|| = \frac{\bar{\mathcal{P}}(u)}{\tilde{\mathcal{P}}(v_r)} v_r = \frac{\mathcal{P}(u) + \Delta_\omega}{\mathcal{P}(v_r) + \tilde{\delta}} v_r \tag{6.53}$$

We note that choosing $\tilde{\delta} > \Delta_\omega$ has the effect of slowing down the robot (since $\bar{\mathcal{P}}(u) - \tilde{\mathcal{P}}(u) < 0$, we have a similar effect of having a negative disturbance power Δ_p in (6.26), which slows down the robot as discussed in Remark 13). Therefore, choosing a constant value for $\tilde{\delta}$ such that $\tilde{\delta} > \Delta_\omega$ is equivalent to choosing a lower value of v_r.

Instead of using a constant value for $\tilde{\delta}$, our approach is to use double sigmoid functions to make the value of $\tilde{\delta} > 0$ only near waypoints w_i and zero otherwise, meaning that $\tilde{\delta}$ is only activated near waypoints, which can be described as:

$$\tilde{\delta}(s) = \sum_{i=1}^{n_w - 1} P_i \tilde{\sigma}_i(s) \tag{6.54}$$

where $P_i > 0$ is a conservative estimate of power consumption due to rotation near waypoint w_i and $\tilde{\sigma}_i(s)$ is defined as

$$\begin{aligned}\tilde{\sigma}_i(s) &= \tilde{\sigma}_i^r \tilde{\sigma}_i^f \\ \tilde{\sigma}_i^r(s) &= \frac{1}{1+e^{-\tilde{\beta}(s-(s_i-\frac{1}{2}\phi))}} \\ \tilde{\sigma}_i^f(s) &= \frac{1}{1+e^{\tilde{\beta}(s-(s_i+\frac{1}{2}\phi))}} \\ \phi &= \frac{\tilde{d}}{L}\end{aligned} \tag{6.55}$$

where $\tilde{\beta} > 0$, and is $\tilde{d}$ a distance on the path from the start of slowing down till its end and L being the path length. The expression (6.54) aims to start activating $\tilde{\delta}(s)$ a distance $\frac{\tilde{d}}{2}$ before waypoint w_i along the path, and end this activation a distance $\frac{\tilde{d}}{2}$ after the waypoint along the path. Figure 6.8 illustrates this idea for the path example from Figure 6.5. We update the energy sufficiency candidate CBF in (6.15) to be

$$h_e = E_{nom} - E - \tfrac{\mathcal{P}(v_r)}{v_r}(L(1-s) - \delta_m) - \int_s^1 \tilde{\delta}(\tau)d\tau \tag{6.56}$$

and applying Leibniz rule to the last term the constraint associated with this candidate CBF is

$$-\mathcal{P}(u) - \Delta_\omega + \tfrac{\tilde{\mathcal{P}}(v_r)}{v_r}(L\eta - \dot{L}(1-s)) + \tilde{\delta}(s)\eta \geq -\gamma_e h_e \tag{6.57}$$

We note that the integrand in (6.56) can be carried out numerically. Before showing that (6.56) is a CBF, we need to choose appropriate values for P_i in (6.54) and $\tilde{d}$, both calling for an estimate for a bound on rotation speed of a robot near a waypoint.

Lemma 15. *For a unicycle type robot with kinematics described in* (6.48)*, applying the transformation* (6.49) *and* (6.50) *to follow a single integrator control input of a point moving with a speed v_r along the path, the rotation speed at a waypoint w_i is*

$$\omega_i \leq \frac{v_r}{\ell}\sin\psi_i \tag{6.58}$$

where ψ is defined in (6.45).

Proof. We start by considering $0 \leq \psi \leq \frac{\pi}{2}$. Without loss of generality, suppose there is a robot at origin with $\theta = 0$ (aligned with x-axis), and $u = v_r \begin{bmatrix}\cos\psi & \sin\psi\end{bmatrix}^T$. From (6.49) we have

$$\omega = \frac{v_r}{\ell}(\cos\theta\sin\psi - \sin\theta\cos\psi) = \frac{v_r}{\ell}\sin(\psi-\theta) \tag{6.59}$$

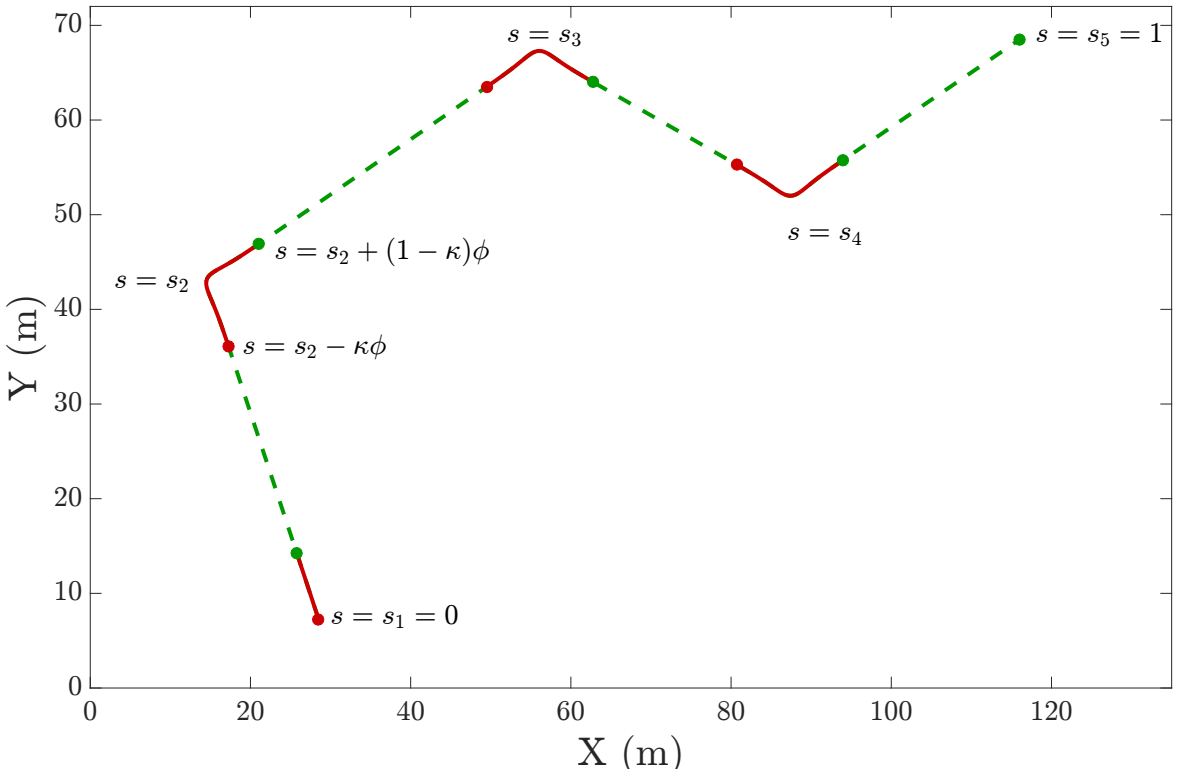


Figure 6.8 An example showing activation regions of $\tilde{\delta}(s)$. The path is similar to that illustrated in Figure 6.5. The red segments are segments where $\tilde{\delta}(s)$ is activated. The red dots indicate the points where $s = s_i - \frac{1}{2}\phi$ and the green ones are points where $s = s_i + \frac{1}{2}\phi$. In this example we choose $\tilde{d} = 15$m.

Consider the Lyapunov function $V = \alpha^2$ where $\alpha = \psi - \theta$, then $\dot{V} = -2\alpha\dot{\theta} = -2\frac{v_r}{\ell}\alpha \sin\alpha \leq 0$ and since $\dot{V} = 0$ only at $\alpha = 0$ then α converges to $\alpha = 0$ by virtue of Lasalle's invariance principle, meaning that α, and hence ω, monotonically decrease for $\psi \leq \frac{\pi}{2}$. We are interested in finding the maximum value of ω, so by differentiating (6.59)

$$\frac{d\omega}{d\theta} = -\frac{v_r}{\ell}\cos(\psi - \theta) = 0 \Rightarrow \theta^* = \psi - \frac{\pi}{2}. \tag{6.60}$$

Since we are considering $\psi \leq \frac{\pi}{2}$, it follows that $\theta^* \leq 0$. However, since we consider that the robot starts at $\theta = 0$ and that ω monotonically decreases, then we can take $\theta^* = 0$, meaning ω is maximum at $\theta = 0$. Thus

$$\omega \leq \omega^* = \frac{v_r}{\ell}\sin\psi \tag{6.61}$$

If the robot is at waypoint w_i pointed to the direction of vector $\Delta w_{i-1}^i = w_i - w_{i-1}$, there is always a rotation of axes from the global axes to new ones where $\theta = 0$ and translation of axes to place w_i at the origin, and we reach the same result in (6.61). The lemma follows by choosing $\psi = \psi_i$.

We note that the same result holds for $-\frac{\pi}{2} \leq \psi < 0$, but ω changes sign by virtue of (6.50). For $\frac{\pi}{2} < \psi < \pi$, from (6.49) $v < 0$ and therefore $\omega' = -\frac{v_r}{\ell}\sin(\psi - \theta) = \frac{v_r}{\ell}\sin((\psi - \pi) - \theta)$. Using the same procedure but for angle $(\psi - \pi) < 0$ we get the same result. □

We can use the upper bound estimate of rotation speed near a waypoint w_i to estimate an upper bound for the power consumed during a rotation Δ_ω

$$\Delta_\omega = \mathcal{P}_u(0, \tfrac{v_r}{\ell}\sin\psi) \tag{6.62}$$

In the following we estimate the activation distance $\tilde{d}$ near a waypoint w_i.

Proposition 5. *For a robot with model* (6.48) *and applying* (6.49) *and* (6.50) *to follow a single integrator control input for a point moving with speed v_r along the path, the distance needed till attenuation of angular speed, i.e. $\omega \leq \frac{v_r}{\ell}\epsilon_\omega$ with ϵ_ω being an arbitrarily small number, is*

$$d_a \leq \ell\frac{\pi}{2}\log\frac{\psi_i}{\epsilon_\omega} \tag{6.63}$$

Proof. We use the candidate Lyapunov function $V = \alpha^2 = (\psi_i - \theta)^2$, so $\dot{V} = 2\frac{v_r}{\ell}\alpha\sin\alpha$. One result of (6.50) is that $\alpha \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ as discussed in proof of Lemma 15. We can prove exponential stability for the candidate Lyapunov function if we can find $k_1, k_2, k_3 > 0$ such that [74]

$$\begin{aligned} k_1\alpha^2 \leq V \leq k_2\alpha_2 \\ \dot{V} \leq -k_3\alpha^2 \end{aligned} \tag{6.64}$$

Since $V = \alpha^2$, then $k_1 = k_2 = 1$. We can estimate k_3 by letting the parabola $f(\alpha) = k_3\alpha^2$ and $g(\alpha) = \frac{2v_r}{\ell}\alpha\sin\alpha$ intersect at $\alpha = \frac{\pi}{2}$, which gives $k_3 = \frac{4v_r}{\ell\pi}$. By virtue of V being exponentially stable on $\alpha \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, then

$$\alpha \leq \psi_i e^{-\frac{k_3}{2}t} \tag{6.65}$$

then at time $\tilde{t}$ the right hand side of the last inequality is equal to ϵ_ω

$$\psi_i e^{-\frac{k_3}{2}\tilde{t}} = \epsilon_\omega \Rightarrow \tilde{t} = \tfrac{2}{k_3}\log\tfrac{\psi_i}{\epsilon_\omega} \tag{6.66}$$

and we note that $\alpha < \epsilon_\omega$ at $t > \tilde{t}$. The attenuation distance then is

$$d_a \leq \tilde{d}_a = \tilde{t}v_r = \ell\frac{\pi}{2}\log\frac{\psi_i}{\epsilon_\omega} \tag{6.67}$$

We note that at $t = \tilde{t}$ the angular speed will be

$$\omega \leq \frac{v_r}{\ell} \sin \epsilon_\omega \approx \frac{v_r}{\ell} \epsilon_\omega \tag{6.68}$$

□

Theorem 9. *For a robot with unicycle kinematics, applying* (6.49) *and* (6.50)*, and provided that* $\tilde{\delta}$ *in* (6.54) *is formed such that*

$$\tilde{\delta} \geq \frac{L}{v_r} \left(\sum_{i=1}^{n_w - 1} \mathcal{P}_u(0, \tfrac{v_r}{\ell} \sin \psi_i) \tilde{\sigma}(s) + \Delta_\epsilon \right) \tag{6.69}$$

and $\tilde{d} = 2 \max\{\tilde{d}_a, d\}$*, where* d *is the tracking distance from* (6.19) *and* Δ_ϵ *is robot's power consumption when* $\omega = \epsilon_\omega$*, and if* $v_r^* = \sqrt{\frac{m_0}{m_2}} \leq u_{max}$*, with* $u_{max} > 0$ *being maximum robot speed, then* h_e *in* (6.56) *is a ZCBF. Moreover, if* (6.57) *is applied in* (6.41) *instead of* (6.16)*, energy sufficiency is guaranteed.*

Proof. We start by considering when the robot moves on a straight line and away from waypoints, i.e. $s_{i-1} + \frac{\phi}{2} < s < s_i - \frac{\phi}{2}$, in which case the proof is similar to proof of lemma 12 since $\mathcal{P}(u) = \mathcal{P}_u(v, 0)$ as pointed out in remark 14.

We note that when $h_d \approx 0$, the robot is following the reference point x_r along a straight line and it starts rotating after x_r passes waypoint w_i, i.e. $s \geq s_i$, and since $||x - x_r|| \approx d$ it means that the robot will start rotation a distance d away from w_i, but since $\tilde{d} = 2 \max\{\tilde{d}_a, d\}$, $\tilde{\delta}$ will be activated at a distance greater than or equal d before w_i along the path, i.e. before the robot starts spinning. Also when $s_i < s \leq s_i + \frac{\phi}{2}$ and due to choice of $\tilde{d}$, $s = s_i + \frac{\phi}{2}$ happens at least a distance $\tilde{d}_a$ after w_i along the path and at this point $\omega < \frac{v_r}{\ell} \epsilon_\omega$, i.e. $\tilde{\delta}$ will be deactivated after ω has been attenuated.

When $\tilde{\delta}$ is activated, i.e. $s_i - \frac{\phi}{2} < s < s_i + \frac{\phi}{2}$, and considering the critical case where $h_e \approx 0$ and $h_d \approx 0$, similar to what we did in the proof of Lemma 12, we consider the equality of (6.20) and (6.57), so from (6.57) (and noting that $\dot{L} = 0$ when the robot is moving along the path by virtue of proposition 3)

$$\eta = \frac{\mathcal{P}(u) + \Delta_\omega}{\mathcal{P}(v_r) + \tilde{\delta} \frac{v_r}{L}} \frac{v_r}{L} \tag{6.70}$$

and doing the same steps to obtain (6.24)

$$||u|| = \frac{\mathcal{P}(u) + \Delta_\omega}{\mathcal{P}(v_r) + \tilde{\delta} \frac{v_r}{L}} v_r \tag{6.71}$$

which is a similar root finding problem to (6.24). Since $\tilde{\delta} \geq \mathcal{P}_u(0, \frac{v_r}{\ell} \sin \psi_i) \frac{L}{v_r}$, then $\tilde{\delta} \frac{v_r}{L} \geq \Delta_\omega$, which ensures roots for (6.71) exist and that $||u||$ will converge to a slower speed than v_r as dis-

cussed in Remark 13 (since having $\tilde{\delta}\frac{v_r}{L} - \Delta_\omega < 0$ has a similar effect as having $\Delta_p < 0$ in Remark 13). Moreover, when $\omega < \frac{v_r}{\ell}\epsilon_\omega$, (6.71) will become

$$||u|| = \frac{\mathcal{P}(u) + \Delta_\epsilon}{\mathcal{P}(v_r) + \tilde{\delta}\frac{v_r}{L}} v_r \tag{6.72}$$

which is guaranteed to have roots since $\tilde{\delta}\frac{v_r}{L} \geq \Delta_\epsilon$. Thus provided that $\sqrt{\frac{m_0}{m_2}} \leq u_{\max}$ there is always a value of u that satisfies (6.57). Since (6.19) and (6.56) are ZCBFs and if δ in (6.56) is equal to δ_m from (6.30) then the robot's energy satisfies $E(t) = E_{nom}$ only when $||x - x_c|| < \delta$ as discussed in the proof of Theorem 7, thus ensuring energy sufficiency. □

We can apply the same quadratic program in (6.41), but with replacing the definition of energy sufficiency CBF, and for that the A and B matrices in (6.41) will be

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \frac{\mathcal{P}(v_r)}{v_r}L + \tilde{\delta} & \mathbf{0}_{1\times 2} \\ 1 & \mathbf{0}_{1\times 2} \\ (x - x_r)^T \frac{\partial x_r}{\partial s} & -(x - x_r)^T \end{bmatrix} \\
\mathbf{B} &= \begin{bmatrix} -\gamma_e h_e + \mathcal{P}(u) + \Delta_\omega + \dot{L}(1-s) \\ -\gamma_b h_b \\ -\gamma_d h_d \end{bmatrix} \\
\mathbf{u}_{nom} &= \begin{bmatrix} 0 & u_{nom} \end{bmatrix}
\end{aligned} \tag{6.73}$$

We can follow the same steps as in Theorem 7 to show that energy sufficiency is maintained solving this QP problem over a fixed path, with the same path freezing idea as in Proposition 3.

The treatment thus far concerns a unicycle robot moving around, with a fixed path back to charging station. A path planner could be used to update the path in the same manner discussed in Section 6.5. We can use a similar sequence as in Algorithm 1, but we need to show that the SPC method is a valid backup for the proposed unicycle adaptation.

Proposition 6. *Sequential Path Construction does not violate energy sufficiency for a robot described by* (6.48) *when applying* (6.41) *with transformation* (6.49)*,* (6.50)*, and with A and B matrices described in* (6.73)*.*

Proof. Similar to proposition 4, provided that there exists a value of $u = u_{nom}$ and $\eta = \eta_{nom}$ satisfying following inequalities

$$h_e = E_{nom} - E - \tfrac{\mathcal{P}(v_r)}{v_r}(L(1-s) - \delta) - \int_s^1 \tilde{\delta}(\tau)d\tau \tag{6.74a}$$

$$-\mathcal{P}(u) - \Delta_\omega + \frac{\bar{\mathcal{P}}(v_r)}{v_r}(L\eta_{nom} - \dot{L}(1-s)) + \tilde{\delta}(s)\eta_{nom} \geq -\gamma_e h_e \tag{6.74b}$$

we need to show that (6.74) is not violated at a path update. Similar to proof of Proposition 4, provided that nominal control inputs are continuous and satisfying (6.74), then there are no jumps (i.e. instantaneous changes) for $E, \mathcal{P}(u), \Delta_\omega$ and v_r. Moreover since SPC is path length invariant, L does not change. Also since SPC is path angle invariant from Lemma 14, then the increase in power due to the addition of the new waypoint is equal to zero (because ψ_2 for the new path is equal to zero) and no change occurs for the power consumption along the path, therefore $\tilde{\delta}$ does not jump as well, meaning (6.74) is not violated under SPC. □

We can apply Algorithm 1 and the same logic in Theorem 8 to show that energy sufficiency is maintained under discrete path updates.

Remark 15. *The adaptation we are using for the method based on single integrator dynamics (in Section 6.5) to unicycle dynamics is versatile and can go beyond accounting for excess power consumption near waypoints. This is due to the fact that the estimated excess power, e.g.* (6.54)*, is modelled as a summation of double sigmoid functions, activated along different segments of the path. Moreover, this excess in the estimated power consumption is incorporated in the energy sufficiency CBF* (6.56) *through numerical integration, making it easier to account for different types of "resistance" along the path. For example, effects like surface inclinations, variability in friction and increased processing power, among many others, could be modelled in a similar way to* (6.54) *through identifying ranges of the path parameter s corresponding to different segments on the path, each associated with a double sigmoid function multiplied by the estimated power consumption related to the effect being modelled.*

Remark 16. *The adaptation proposed so far can be applied to different types of robot models. The core idea behind this adaptation is to apply the single integrator model input to the actual robot, that could be modelled more accurately with another model, i.e. map the single integrator control action to the control action pertaining to a specific robot. This mapping can be achieved through a transformation as in* (6.49)*, or through a cascaded control that tracks the single integrator velocity as an input. We then study the effect of this mapping on the additional disturbance power, e.g. the δ_ω term in* (6.53) *that models the effect the angular speed ω on power consumption, and consequently we choose a "correction" function, e.g. the $\tilde{\delta}$ function in* (6.53)*, such that system stability is maintained, i.e. the root finding problem* (6.53) *has a solution.*

6.7 Results

6.7.1 Simulation Setup

We present the simulation results that highlight the ability of our proposed framework to ensure energy sufficiency during an exploration mission. The considered experimental scenario allows the robots to perform the exploration mission while ensuring robot's energy consumption is within the dedicated energy budget E_{nom}.

We evaluate the approach using a physics-based simulator [117]. We use a simulated KheperaIV robot equipped with a 2D lidar with a field of view of 210 degrees and a 4m range as the primary perception sensor. The architecture of the autonomy software used in simulations is shown in Figure 6.9A. The autonomy software allows the robot to explore and map the environment. Each robot is equipped with a volumetric mapping system [190, Voxblox] using Truncated Signed Distance Fields to map the environment. A graph-based exploration planner [GBPlanner, 168] uses the mapping system to plan both the exploration and homing trajectories. We carry out a path shortening procedure as described in [183, Algorithm 1] to eliminate redundant and unnecessary points from the original path planner output, making the final path straighter and shorter. Our proposed framework is implemented as a Buzz [118] script that periodically queries the exploration planner for a path and applies the required control commands to the robot.

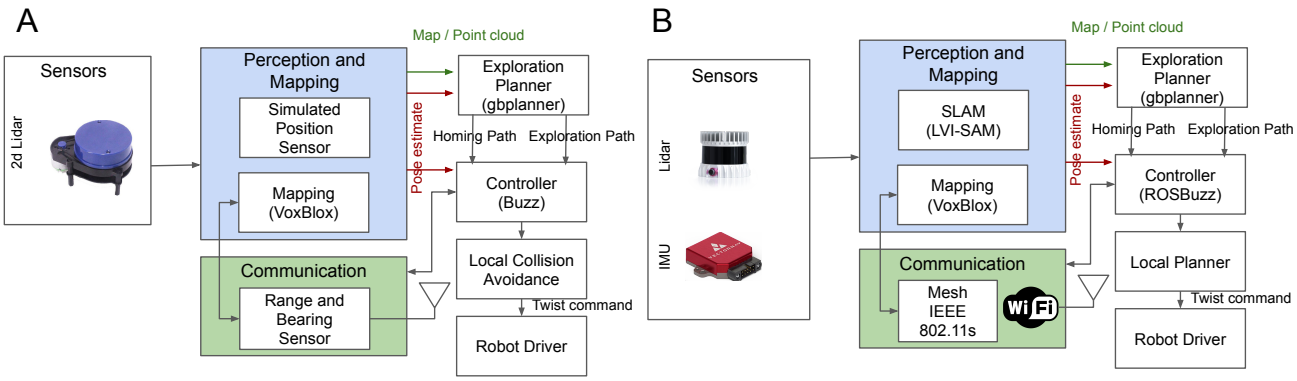


Figure 6.9 Software Architecture used during the simulation study (A) and on the experimental hardware (B).

We use the maze map benchmarks from [1] as a blueprint for obstacles in the environment, and each map is scaled so that it fits a square area of 30×30 meters. In each simulation one robot maps the unexplored portions of the map to maximize its volumetric gain [168]. We run four groups of experiments for three different maps from the benchmarking dataset [1], and for each case we run 50 simulations with randomized configurations to obtain a statistically valid dataset.

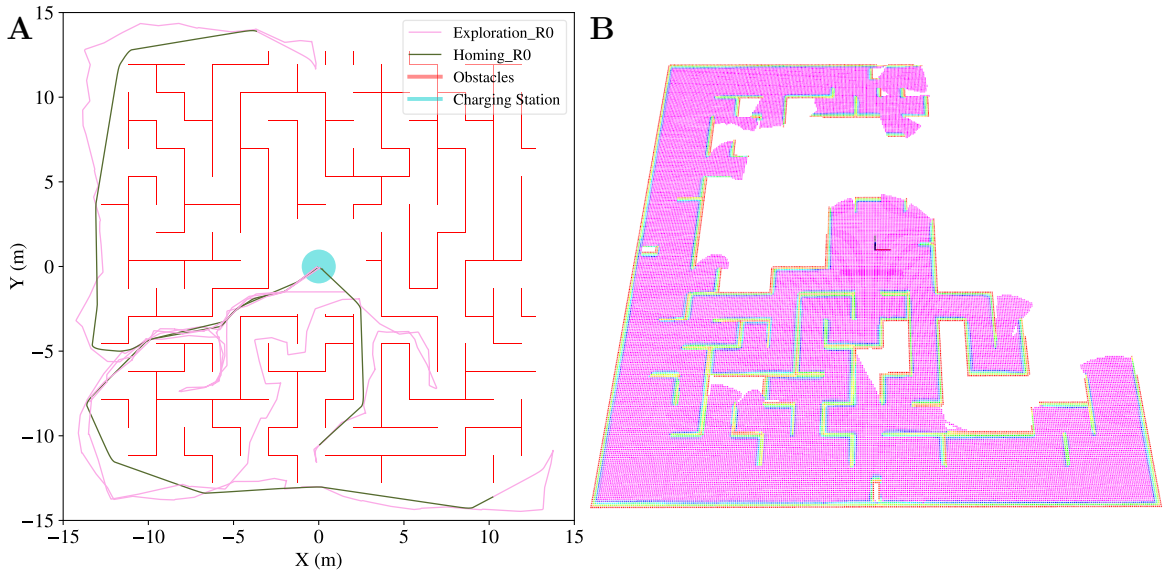


Figure 6.10 A sample result for the trajectories generated by the robot (A) and the map constructed in the same simulation run (B) for an exploration task in a maze environment (maze-4 from [1]), while using our proposed approach to maintain energy sufficiency.

We use a polynomial power model to describe power consumption of the robots in simulation. We derive this model by collecting power consumption readings from a physical AgileX Scout Mini [124] robot at different values of linear and angular speed, then we fit a surface through these readings to obtain out power model. Figure 6.11 shows the fitted power model, along with the actual collected power readings from the robot. We interface the single integrator output $\mathbf{u}^*$ of (6.41) to the unicycle model of the robots using the transformation (6.49) and the modification (6.50). We use the robot's linear and angular speeds to estimate the robot's power according to the following polynomial

$$\begin{aligned}\mathcal{P}_u(v,\omega) = 27.8126||v||^2 - 107.7343|\omega|^2 \\ + 31.4578||v|| + 179.9095|\omega| + 1.234\end{aligned} \tag{6.75}$$

and the power model is depicted in Figure 6.11. We also add to this model an additional power of $\mathcal{P}_{payload} = 20$W to account for payload power consumption.

6.7.2 Simulation Results

Figure 6.10 shows an example of the robots' trajectory during exploration and returning to the charging station for one simulation run in one of the maze environments (maze-4). Figure 6.10 also shows the map built by the robot during this simulation run. As observed in the trajectory plot, any

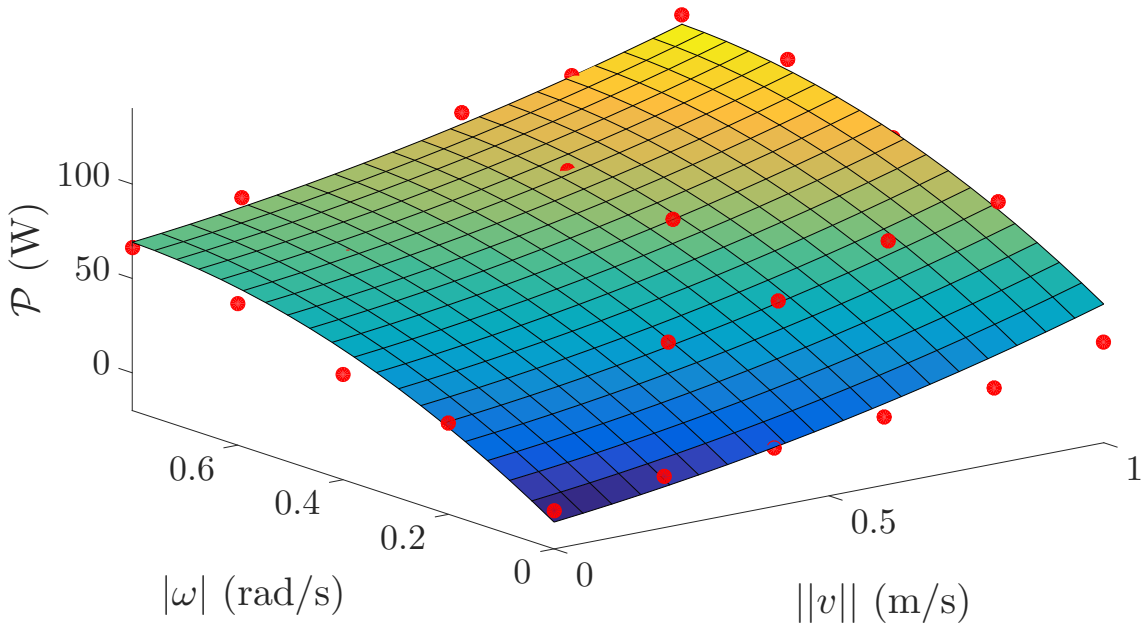


Figure 6.11 Surface plot of the power model used in simulation. The red dots are the actual measured power values at different values of linear and angular speeds (v and ω) and is fitted by a 3 dimensional surface to minimize the mean least square error between the model and the real data points.

given robot's exploration trajectory is always accompanied by a homing trajectory to the charging station satisfying the energy constraints.

For all simulation runs we measure the estimated Total Area Covered (TAC) and the Energy On Arrival (EOA), which is the amount of energy consumed by the robot by the time it arrives back to the station, and we use these values as metrics for performance. The TAC serves as a measure of the mission execution quality, and the EOA is a measure of the extent the available energy budget has been used. We run all test cases at two desired values of return speed: a slow speed of $v_r = 0.1$m/s and a faster speed of $v_r = 0.5$m/s. We highlight the efficacy of our approach by comparing the aforementioned metrics to the results of a baseline method in which a robot returns back on the path when the available energy reaches a certain fixed threshold percentage of the total nominal energy (as it is a standard procedure with commercial robots). For the baseline we use only the tracking CBF (6.19) in a QP problem similar to (6.41) and we change the path parameter s according to the following relation

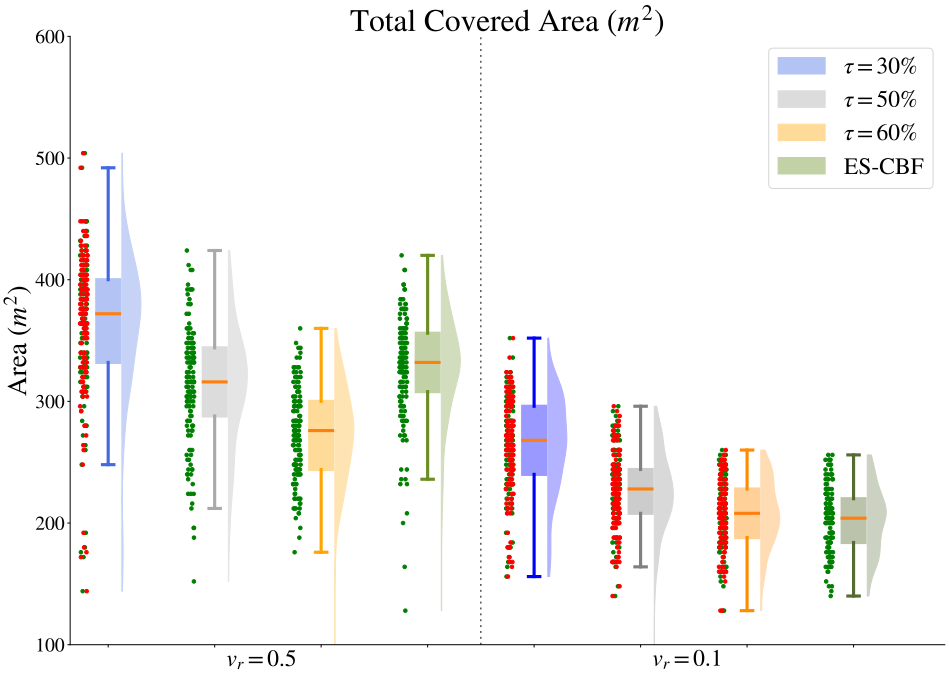
$$\dot{s} = \begin{cases} \frac{v_r}{L}, & \text{if} \quad \frac{E}{E_{nom}} > \tau \\ 0, & \text{otherwise} \end{cases} \tag{6.76}$$

where $0 < \tau < 1$ is a threshold return energy ratio, and L is the total path length at the point when the robot starts moving back towards the charging station. We show the results of our comparison for the aggregated values of TAC for different simulation scenarios in Figure 6.12a. To highlight the relation between TAC and EOA we use red dots in Figure 6.12a to indicate TAC values corresponding to simulation runs during which the energy budget is violated, i.e. EOA is less than zero at least once, indicating the robot's failure to recharge before its energy budget is fully consumed. Figure 6.12b and 6.12c show histograms of EOA values distribution for our method as well as baseline at different values of τ for $v_r = 0.5$m/s and $v_r = 0.1$m/s respectively. In all simulation runs the total energy budget is set to be 12kJ.

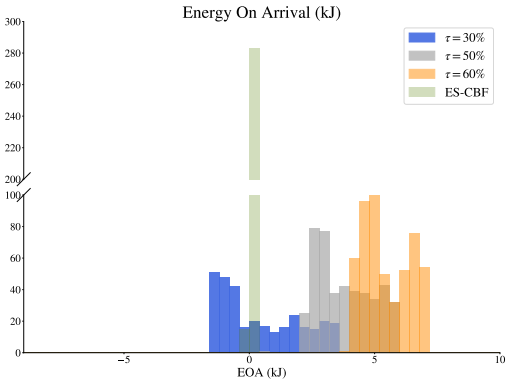
We note that in Figure 6.12a for $v_r = 0.1$m/s the area covered consistently increases with decreasing return energy threshold percentages, i.e. when a robot starts returning to recharge at $\tau = 0.3$ it typically covers more area than when it needs to return at $\tau = 0.5$ as it uses more of its energy to carry out its mission. Although for this case the area covered using our proposed method is less than baseline (box plot median value of 268m^2 for $\tau = 0.3$ and 204m^2 for ES-CBF, meaning a 24% reduction in TAC in the worst case), baseline results have significantly more red dots than ES-CBF, indicating significantly more violations of energy budget than ES-CBF, so although TAC is more for baseline the energy budget is violated for most test runs.

For $v_r = 0.5$m/s, Figure 6.12a shows an overall increase in TAC for both baseline and ES-CBF compared to the case where $v_r = 0.1$m/s. Moreover, we notice an increase in TAC in case of ES-CBF over baseline with $\tau = 0.5$ and $\tau = 0.6$ (5% and 20% increase in TAC respectively), while there is a decrease of 10% in TAC between baseline with $\tau = 0.3$ and ES-CBF. For baseline cases with $\tau = 0.5$ and $\tau = 0.6$ there are no red dots at v_r=0.5m/s in Figure 6.12a, but there are numerous violations of energy sufficiency for baseline with $\tau = 0.3$. Overall, choosing a threshold value to return to the charging station depends on the map and task at hand, and does not provide guarantees for either optimal mission success or return or respecting the energy budget. On the contrary, our method guarantees that the energy budget is fully exploited, without affecting mission performance.

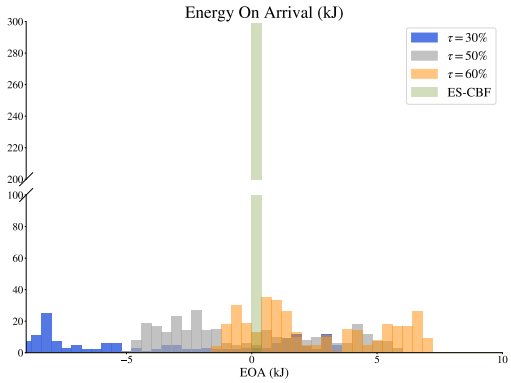
It is also worth noting from Figure 6.12b and 6.12c that the distribution of EOA values is very tight around zero, meaning that robots applying ES-CBF framework arrive to the station without violating the energy budget allocated and without wasting energy, i.e. robots do not arrive too late or too early, which means full utilization of the energy allocated. On the other hand for the baseline method we can see in Figure 6.12c for $v_r = 0.1$ that EOA values are more widely dispersed around zero, with a significant portion of the values being positive or negative, indicating robots arriving to station either too early or too late, which is a direct result of not considering needed energy to return back to station (e.g. a robot could reach the return threshold τ when it is relatively close to



(a) Estimated total area covered



(b) Energy on arrival for simulations with $v_r = 0.5$m/s.



(c) Energy on arrival for simulations with $v_r = 0.1$m/s.

Figure 6.12 Comparison between baseline method for three different threshold percentages τ and our CBF-based approach for energy sufficiency, denoted *ES-CBF*. Simulation data for total area covered and energy values upon arrival to charging station is collected for three test environments and two different desired return speeds ($v_r = 0.5$m/s and $v_r = 0.1$m/s), each run for 50 instances with different random seeds. The red dots in Figure 6.12a indicate area values corresponding to simulation instances where the energy budget is violated at least once, while green dots indicate no violation of energy budget. Histograms 6.12b and 6.12c show distribution of energy on arrival (EOA) values for $v_r = 0.5$m/s and $v_r = 0.1$m/s respectively.

the station so it will eventually arrive back with a significant amount of energy, and it may reach τ when it is far away so that the energy budget is fully depleted on the way back). We also note that for $v_r = 0.5$m/s the values of EOA are mostly positive for baseline with $\tau = 0.5$ and $\tau = 0.6$ indicating significant non-utilized energy when the robot returns back to recharge. Therefore for these two baseline cases the robots utilize less energy for exploration and this explains the advantage that ES-CBF has in TAC over these two baseline cases at $v_r = 0.5$m/s.

6.7.3 Hardware setup

We study the performance of the energy-sufficiency approach using an AgileX Scout Mini rover equipped with a mission payload to perform exploration and mapping missions, shown in Figure 6.13. The robot has an Ouster OS0-64 lidar as the primary perception sensor and a high-performance Inertial Measurement Unit (IMU) from VectorNav. A mesh communication router implements IEEE802.11s to communicate with the base station.

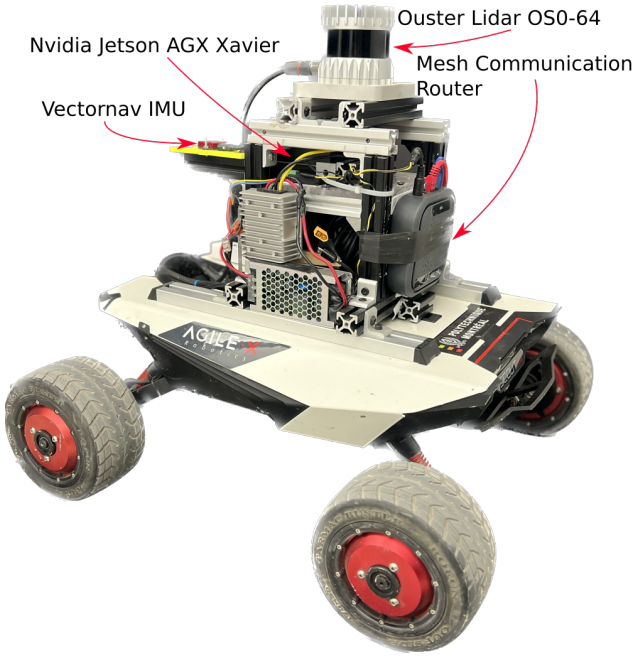


Figure 6.13 Experimental setup we use to perform the exploration mission while maintaining energy sufficiency. It consists of an AgileX Scout Mini rover with a mission payload mounted on top as demonstrated above.

Figure 6.9B shows the software architecture deployed on the Nvidia Jetson AGX Xavier of the rover. We implement a full stack Simultaneous Localization And Mapping (SLAM) system, mesh communication system, and a local planner for collision avoidance. Unlike the simulation, the rover performs a full-stack 3D localization and mapping using a variant of LVI_SAM [191] with a

front-end generating pose graphs and a back-end performing map optimization. The mapping [190, Voxblox] and planning (Gbplanner modules and controller were the same for both simulation and hardware. We apply the path-shortening procedure in [183, Algorithm 1] on the output of the path planner, as we do in the simulation setup.

We estimate the robot's power consumption using voltage and electric current values. Upon arrival to charging region the robot executes a simple docking manoeuvre to enter the charging region and carries out a simulated battery swap operation to replenish the robot's energy. We point out that such setup does not affect the validity of the experiment and could be justified by the fact that the energy consumed by the robot is consistent, meaning that the power needed to move the robot at a certain speed does not depend on the battery, but rather depends on the robot's mechanical properties and the environment which are both static.

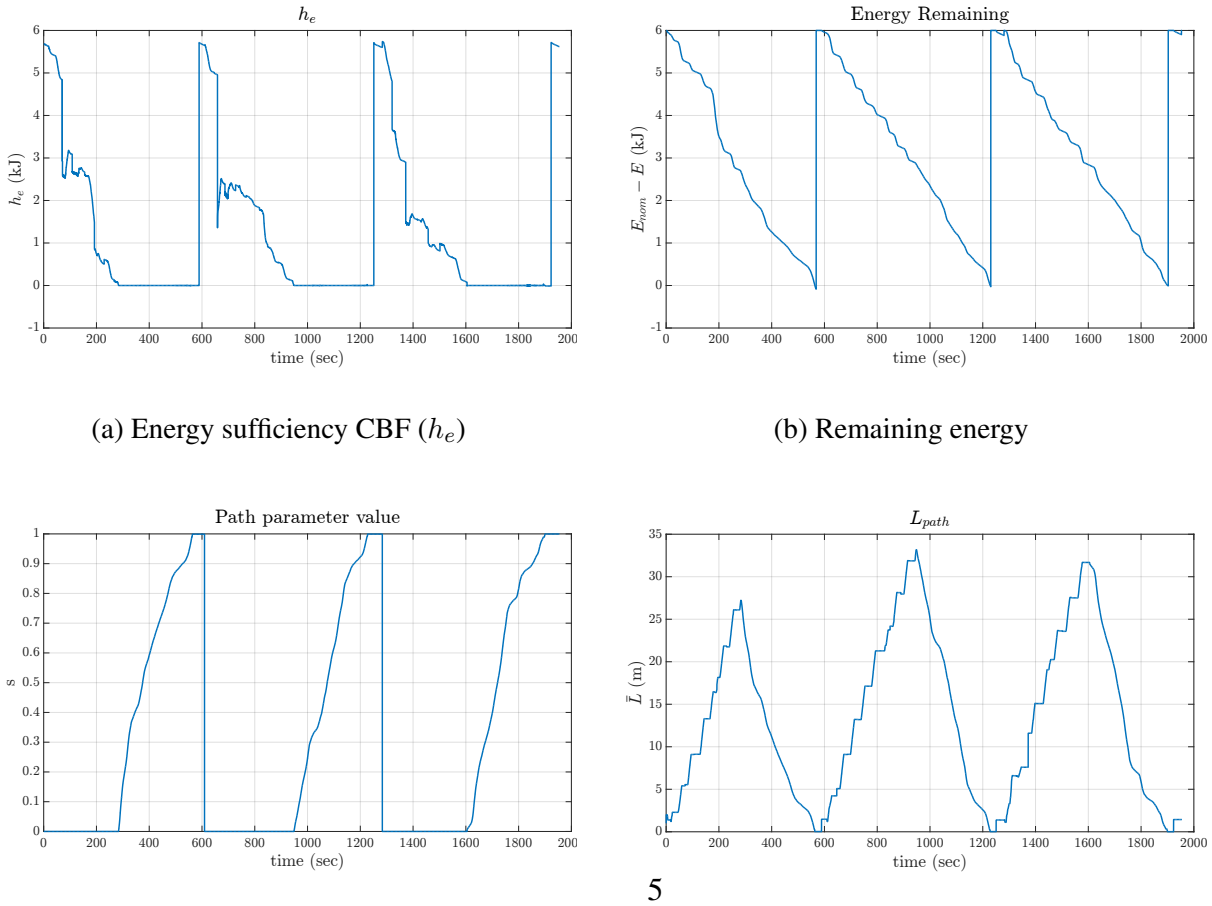
6.7.4 Hardware results

We apply the proposed method on our experimental setup and we show the results in Figure 6.14, as well as the point cloud map for the experimental run in Figure 6.1. In this experiment the robot is tasked with exploring and mapping a set of corridors and hallways while returning back to a charging spot. The map generated by the robot and the trajectories taken by the robot during an experimental run are shown in Figure 6.1.

For this experiment the allocated energy budget is 7kJ and the desired return speed was set to $v_r = 0.2$m/s. We note from Figure 6.14b that the robot consumes the energy budget fully by the time it arrives back to recharge, which shows the ability of our proposed approach to maintain energy sufficiency in cluttered environments. From Figure 6.14c the path parameter value is equal to zero as long as the energy sufficiency constraint (6.16) is not violated, then when $h_e \approx 0$ (in Figure 6.14a, indicating energy sufficiency being close to the boundary of its safe set) it starts to increase and drive the robot back towards the station along the path. Figure 6.1 shows examples of paths taken by the robot while exploring its environment.

6.8 Conclusions

In this work we present a CBF based method that provides guarantees on energy sufficiency of a ground robot in an unknown and unstructured environment. Our approach is to augment a sampling based path planner [like GBplanner, 168] by a CBF layer, extending our work [157] to endow a robot with the ability to move along a path in an energy aware manner such that the total energy consumed does not exceed a predefined threshold. We described a continuous representation for piecewise continuous paths produced by a path planner. We define a reference point that slides



(a) Energy sufficiency CBF (h_e)

(b) Remaining energy

(c) Path parameter

(d) Path length

Figure 6.14 Results from the robot experiment. The robot returns back to station with the expected battery level to recharge, achieving energy sufficiency.

along this continuous path depending on robot's energy. We show the relationship between the constraints for controlling both the reference point and robot's position and show conditions for these constraints to complement each other. We demonstrate how these ideas are valid for dynamic cases in which the path planner updates the path frequently and the robot is carrying out a mission. Finally we demonstrate a method for adapting our framework, based on a single integrator model, to a unicycle model. We highlight through simulation and experimental results the ability of our method to deal with unknown and unstructured environments while maintaining energy sufficiency.

Our proposed framework has the advantage of flexibility and adaptability to different types of robot models and environments. Such framework can be useful in many application where long term autonomy is needed, e.g. underground and cave exploration, robot reinforcement learning, self driving cars in urban environments, and many others.

As a future work we plan to extend our framework to be able to handle coordination between multiple robots to share a charging station in the same spirit as [157], while being able to deal with unstructured and complex environments. Another direction could be using online estimation and learning techniques to handle power models that are variable by nature and need constant adaptation, such as wind fields, snowy conditions, etc.

CHAPTER 7 GENERAL DISCUSSION

In this chapter, we examine the impact of the results presented so far on the field. In Section 7.1 we review the general properties of the main body of our work presented in previous chapters. Then in Section 7.2 we describe briefly some of the technical issues we faced during developing the results in this thesis, as well as some of the ideas that we tried for tackling these issues.

7.1 General discussion: features and advantages

7.1.1 Energy sufficiency for multi-robot systems

Energy sufficiency framework

The first key contribution of this thesis is the development of an energy sufficiency (ES) framework for multi-robot systems sharing a charghing station (CS, **FW1**), developed and presented in Chapters 4 and 5. We can summarize the core idea as: CBF constraints for ES are formulated in terms of the voltage needed by a robot to return to a CS, and coordination CBF constraints between robots are formulated in terms of differences in arrival times of each two robots to the CS, and these constraints are used in a quadratic problem (QP) that produces a control action that drives the robot. The constraints are constructed in a manner that ensures the feasibility of the QP. Adopting this approach, gives **FW1** the following advantages:

- Performance guarantees: Using CBFs gives theoretical guarantees on ES and mutually exclusive use of CS for robots in a multi-robot system.
- Mission-agnostic and decentralized: **FW1** can be applied on a wide range of multi-robot missions, as we demonstrated through simulations of various types of missions (coverage and random walks). Moreover, it has no need for central agents to coordinate among robots.

Capacity of a charging station

Throughout the development of **FW1**, we focused our attention on the scenario where multiple robots share one charging station due to its relevance to the field of MRS and its potential for being used as a building block for more general scenarios with multiple stations. This directly calls for the need to be able to size such a system, namely, to understand how many robots can be serviced using one charging station. From our observation in the literature, this question has not been clearly addressed [39]. We manage throughout Chapters 4 and 5 to find expressions

describing CS capacity. We find that the two most important aspects affecting this capacity are the robot discharge rate (mostly dependent on the robot, its mission, and the environment, but we use a worst-case estimate for it) and the desired difference in arrival times. We also show the effect of having heterogeneity in robots' discharging characteristics and how this can affect station capacity.

7.1.2 Energy sufficiency in unknown environments

One of the main limitations of **FW1** is that robots are capable of traversing only straight lines in a non-cluttered environment. Extending the ES guarantees to scenarios where robots operate in cluttered and unknown environments greatly expands the venues where such framework could be applied. Moreover, demonstrating the ability to extend **FW1** to different and more sophisticated (and common) robot models gives more practical value to our framework. We discussed the construction of the modified ES framework (**FW2**) in Chapter 6, and we focus on achieving ES for individual robots and leave the extension of the coordination capability for future work.

Path smoothing using sigmoid functions

The **FW1** developed in Chapters 4 and 5 uses the distance between a robot and the CS directly in the formulation of ES barrier functions. Extending this framework to more generic environments calls for some way of injecting information about the environment into the CBF formulation. We do this through using a path planner, that indirectly injects the CBF formulation with information about viable paths in the environment. The issue is that most common planners give piecewise linear paths in the form of waypoints, but we need a continuous representation of the path with a parameterized expression to make it easy to integrate path information into the CBF formulation. To this end, we develop a sigmoid-based path smoothing method, and demonstrate its ability to produce parameterized smooth paths that are arbitrarily close to the original piecewise linear path.

We note that the requirement of the path smoothing method being arbitrarily close to the original path is motivated by the idea that paths produced by the path planner are safe by default, and if we stick to these paths with the added advantage of being continuous and parametric, it makes the development of ES CBFs easier, while knowing that the paths are not hitting obstacles.

We also note that the sigmoid-based path-smoothing method we use has a slight similarity with the definition of B-splines. The basis functions for creating a B-spline are recursively defined as

$$B_{i,k}(s) = \frac{s - s_i}{s_{i+1} - s_i} B_{i,k-1}(s) + \frac{s_{i+1} - s}{s_{i+1} - s_i} B_{i+1,k-1}(s) \tag{7.1}$$

where $s_i \in \mathbb{R}$ are knot values. The need for using recursion to construct these basis functions makes the use of B-spline less convenient for our application. Our path-smoothing method essentially replaces these basis functions with sigmoid functions, so the outcome is not a polynomial as in the case of Bezier curves or B-splines. Moreover, we use the length along the path between each of the waypoints and the beginning of the path as the knot value.

Another point that is worth mentioning is that we attempted another method for path smoothing based on Bezier curves. The idea is to create cubic interpolation splines between every pair of consecutive waypoints, then merge these segments following the method in [192] to produce one Bezier curve that has one parameter and resembles the piecewise linear path described by the waypoints. The main issue with this technique is that it does not scale well to paths with many waypoints. For this method to perform well, the order of the final merging Bezier curve has to be increased in a manner proportional to the complexity of the piecewise linear path, but the merging works relatively well for Bezier curves of orders 20 and less, which limits the complexity of paths that could be smoothed.

Energy sufficiency in unknown environments

In **FW2** we leverage the parameterized nature of the smoothed path to create a reference point along the path, the position of which depends on the energy content of the robot. The CBF formulation for ES is then modified to manipulate this reference point in a manner that ensures it arrives at the station by the time the robot's energy budget is up.

We manage to show one key result regarding the stability of **FW2** in relation to the power model and the desired return speed along the path, and we show that there is a cap on the return speed with which a robot can follow the reference point and that the margin of stability decreases with higher desired return speeds.

FW2 has several key advantages:

- Modularity and mission-agnosticism: **FW2** can work alongside any path planner that produces a list of waypoints from the robot's position to a target position. In Chapter 6 we utilized **FW2** alongside an exploration path planner that incrementally builds a map for the environment while the robot carries out frontier exploration. Using CBFs makes **FW2** valid regardless of the mission being carried out, in the same manner as in **FW1**.
- Flexibility of robot model extension: **FW2** is constructed using a single integrator robot model, but we demonstrate the possibility to extend the same framework to more complex robot models (e.g. unicycle robot model as presented in Chapter 6).

Adaptability to different robot models

One key idea we demonstrate in Chapter 6 is the ability to adapt **FW2** to robots described by a unicycle model. The core idea behind this adaptation is to study the effects of the added states (from the new model) on power consumption, then modify the framework to ensure the stability of the framework. For instance, using a unicycle model implies the robot's ability to spin near waypoints, which adds more power consumption that can destabilize **FW2**, and to fix this issue we adjust the framework to slow the robot down near waypoints to ensure framework stability.

We believe that the adaptation we demonstrated in Chapter 6 could be slightly modified for other types of non-holonomic robot models like robots with Ackerman steering. Moreover, in our discussion of Chapter 6 we note that the unicycle model adaptation treated therein could also be extended to take into consideration the variability of power consumption along the path due to different factors like ground friction, surface roughness, and slope changes.

Using voltage vs. energy

Throughout the development of Chapters 4 and 5 we used voltage as the main quantity to be tracked and maintained, while in Chapter 6 we opted for using energy. One main property of the energy metric used in formulating the energy sufficiency CBF is that it has to be non-increasing. If the non-increasing property is not fulfilled, then one viable strategy for a robot to maintain ES constraints is to solely slow down or stop in place, but we want the robot to maintain ES constraints by moving closer to the CS.

In Chapters 4 and 5 we are concerned with using drones in an indoor test environment, so the power consumption is almost constant since the majority of power is used for hovering the drone, and as a result the voltage monotonically decreases and can be used as the main metric in formulating ES CBFs. However, in Chapter 6 we are interested in using a ground robot that can arbitrarily stop or move during its mission, which means that the power consumption is not fixed anymore. With that in mind, using energy is a better fit for ES CBF formulations, and since there is always nonzero payload power consumption we know that the energy will be monotonically decreasing, which makes it more suitable in ES constraints formulation. Moreover, the use of energy facilitates taking into account other forms of energy loss or gain due to friction, changing robot heights on sloped grounds, etc.

7.2 Issues and ideas to explore

In this section we provide a brief discussion of some of the relevant technical issues that we came across during the development of the results in this thesis, as well as some insights and suggestions for tackling these issues.

7.2.1 Communication requirements for multi-agent coordination

In the development of **FW1** for coordination among multiple robots to access a common charging station, presented in Chapters 4 and 5, one key assumption we use is having a complete communication graph among robots is a complete graph, and we leverage this assumption to provide guarantees on the ability of multiple robots to coordinate access to a common charging station in a decentralized manner. This may appear as a strong assumption with high demand on communication bandwidth. In this section, we present the argument that having a complete graph is necessary to provide guarantees on coordination performance.

The degree of situational awareness needed by robots in a multi-robot system varies depending on the task they carry out. To further elaborate on this idea we give an example of two different types of missions with different information demands and communication graph density needs.

The first example is a consensus task carried out by a group of robots. Consensus is the ability of a group of robots to reach a common value, only using information available locally to each robot from its neighbours [193]. The common value agreed upon may differ in its nature from one task to another, which enables a multi-robot system to carry out a wide range of tasks like rendezvous [194], distributed state estimation [195], flocking [196] and formation control [197]. In this example, it is sufficient to use relative state differences to reach a consensus among robots.

The other example is a task allocation mission. In this type of mission, a group of robots reaches an agreement in which some tasks or resources are allocated to robots. Several methods exist in the literature to deal with this problem like consensus-based methods [198, 199], market-based methods and bidding [200, 201], or a blend of both methods [202]. In these methods, one common theme is the need for complete communication among robots in a multi-robot system, and that relative state information is not enough. This usually takes the form of needing to broadcast and receive processed information from all other robots.

One useful idea that is relevant to this discussion is situational awareness[202], i.e. the knowledge of the current state of the system in a manner relevant to the task at hand. We can argue that one fundamental difference between the two examples discussed so far is the idea that the situational awareness required for consensus tasks is of intrinsic nature (local), while that required for task allocation tasks is of an extrinsic nature (global). What we mean by intrinsic situational awareness

is that robots exchange relative information about their states or measurements. On the other hand, extrinsic situational awareness means that robots exchange information about their states relative to external factors. For example, a robot carrying out a rendezvous task need to only exchange information about their relative positions w.r.t. their neighbors (i.e. robots exchange information pertaining to relative states with neighbors). However, in a task allocation scenario using auction-based methods, robots exchange their bids for different tasks based on metrics signifying how capable they are of carrying out these tasks (i.e. robots exchange information of their states relative to external factors that are not part of the multi-robot system).

It is worth noting that we can notice the idea of extrinsic and intrinsic situational awareness being present in other fields besides robotics. Some examples of the need for complete communication graphs to provide extrinsic situational awareness include ledgers in blockchains, and decentralized estimation of Gaussian processes [203].

We present an example problem that highlights the need for a complete graph to achieve successful coordination.

Space coordination in 1-D (SpC) Consider a set of n agents, each described by its state $x_i \in \mathbb{R}, \quad i = 1, \ldots, n$; and each agent changes its state according to the following dynamics

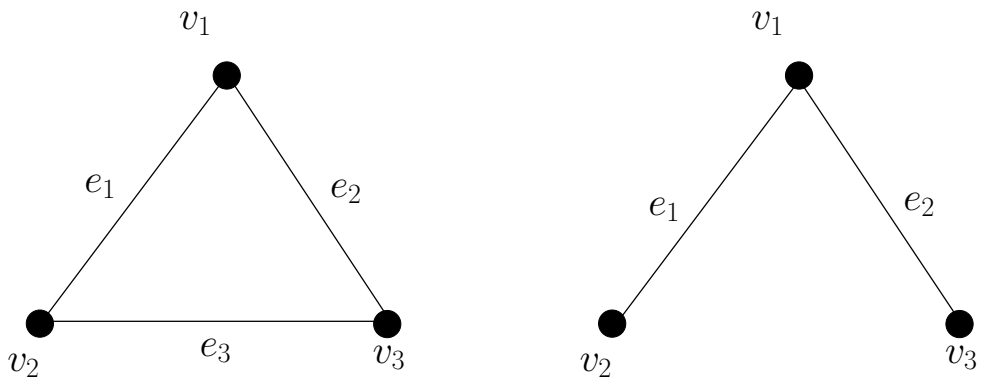
$$\dot{x}_i = u_i \tag{7.2}$$

where $u_i \in \mathbb{R}$ is the control action for robot i. It is required to find control actions u_i for all agents such that starting from any set of initial states, the robots will converge to a configuration where the difference between any two consecutive agents' states is to be at least equal to Δ, i.e. $|x_i - x_j| \geq \Delta, \quad \forall i, j \in \{1, \ldots, n\}, i \neq j$.

The main reason we are interested in the SpC problem is that it can be used to model the separation in arrival times of robots to a charging station. Briefly, if a group of robots can agree on what time instances each robot should arrive (the state becomes the arrival time which is a 1-D quantity), they can do it in a way that makes them avoid arriving at the station while other robots are there.

We consider a scenario with three agents connected in two different ways: a complete graph and a minimally rigid graph, and we follow a simplified version of the treatment in [204]. We consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of edges and $|\mathcal{V}| = n$, $\mathcal{E} = m$, where $|.|$ is the cardinality. The two types of graphs we examine are depicted in Figure 7.1. We define a local potential function for each edge to be

$$\phi_{ij} = \tfrac{1}{2}(|x_i - x_j| - \Delta)^2 \tag{7.3}$$



(a) Complete graph (b) Minimally rigid graph

Figure 7.1 Two example graphs for a set of four agents: a complete graph in 7.1a and a minimally rigid graph in 7.1b. We note that this graph models the communications only among agents and not the actual manifestation of the graph in 2-D, as we are only considering only agents that have 1-D states.

where Δ is the desired inter-agent separation. Each agent follows a gradient descent control law as follows

$$\dot{x}_i = -\sum_{j \in \mathcal{N}_i} \frac{\partial \phi_{ij}}{\partial x_i} \tag{7.4}$$

where $\mathcal{N}_\rangle$ are the neighbours of agent i. A measure of the collective deviation of all edge lengths from Δ can be captured by the following function

$$V = \sum_{(\mathcal{V}_i, \mathcal{V}_j) \in \mathcal{E}} \phi_{ij} \tag{7.5}$$

The idea is to let all agents change their states in a manner that minimizes V through gradient descent [204]. In this treatment, we assume without loss of generality that one of the robots has to have a fixed known state value, and we pick it to be agent #1 and to have a state value equal to zero.

Complete graph

In light of the previous assumption we redefine ϕ_{ij}

$$\begin{aligned}
\phi_{12} &= \frac{1}{2}\left(|x_2| - \Delta\right)^2 \Rightarrow \frac{\partial \phi_{12}}{\partial x_2} = (|x_2| - \Delta)\frac{x_2}{|x_2|} \\
\phi_{13} &= \frac{1}{2}\left(|x_2| - \Delta\right)^2 \Rightarrow \frac{\partial \phi_{13}}{\partial x_3} = (|x_3| - \Delta)\frac{x_3}{|x_3|} \\
\phi_{23} &= \frac{1}{2}\left(|x_2 - x_3| - \Delta\right)^2 \Rightarrow \frac{\partial \phi_{23}}{\partial x_2} = (|x_2 - x_3| - \Delta)\frac{x_2 - x_3}{|x_2 - x_3|} = -\frac{\partial \phi_{23}}{\partial x_3}
\end{aligned} \tag{7.6}$$

For the graph depicted in Figure 7.1a we show the convergence of (7.4), so we calculate $\dot{V}$

$$\dot{V} = \frac{\partial V}{\partial x}\dot{x} = -\begin{bmatrix}(\frac{\partial \phi_{12}}{\partial x_2} + \frac{\partial \phi_{23}}{\partial x_2}) & (\frac{\partial \phi_{13}}{\partial x_3} + \frac{\partial \phi_{23}}{\partial x_3})\end{bmatrix}^T \begin{bmatrix}\frac{\partial \phi_{12}}{\partial x_2} + \frac{\partial \phi_{23}}{\partial x_2} \\ \frac{\partial \phi_{13}}{\partial x_3} + \frac{\partial \phi_{23}}{\partial x_3}\end{bmatrix} = -\left|\left|\frac{\partial V}{\partial x}\right|\right|^2 \tag{7.7}$$

which means the system is stable in the sense of Lyapunov. According to LaSalle invariance principle [74], the system will converge to the largest invariant set in the set of states satisfying $\dot{V} = 0$, thus we need to study states at which $\frac{\partial V}{\partial x} = 0$, and doing so gives the following set of equations

$$\begin{aligned}(|x_2| - \Delta)\frac{x_2}{|x_2|} + (|x_2 - x_3| - \Delta)\frac{x_2 - x_3}{|x_2 - x_3|} = 0 \\ (|x_3| - \Delta)\frac{x_3}{|x_3|} + (|x_2 - x_3| - \Delta)\frac{x_3 - x_2}{|x_3 - x_2|} = 0\end{aligned} \tag{7.8}$$

which are two equations in two unknowns (x_2 and x_3). We need to do further simplifying assumptions to be able to solve them. Without loss of generality, we consider the following two scenarios

Case 1: $x_1 \leq x_2 \leq x_3$**:** in this case $|x_2 - x_3| = (x_3 - x_2)$, $|x_2| = x_2$, $|x_3| = x_3$, and (7.8) becomes

$$\begin{bmatrix}2 & -1 \\ -1 & 2\end{bmatrix}\begin{bmatrix}x_2 \\ x_3\end{bmatrix} = \begin{bmatrix}0 \\ 2\Delta\end{bmatrix} \tag{7.9}$$

which has a unique solution at $\begin{bmatrix}x_2 & x_3\end{bmatrix}^T = \begin{bmatrix}\frac{2}{3}\Delta & \frac{4}{3}\Delta\end{bmatrix}^T$

Case 2: $x_2 \leq x_1 \leq x_3$**:** in this case $|x_2 - x_3| = (x_3 - x_2)$, $|x_2| = -x_2$, $|x_3| = x_3$, and (7.8) becomes

$$\begin{bmatrix}2 & -1 \\ -1 & 2\end{bmatrix}\begin{bmatrix}x_2 \\ x_3\end{bmatrix} = \begin{bmatrix}-2\Delta \\ 2\Delta\end{bmatrix} \tag{7.10}$$

which has a unique solution at $\begin{bmatrix}x_2 & x_3\end{bmatrix}^T = \begin{bmatrix}-\frac{2}{3}\Delta & \frac{2}{3}\Delta\end{bmatrix}^T$. From the results of the previous two cases, we notice that the spacing between any two states in equilibrium will be separated by the same value, regardless of the initial values of these states. We note that the assumption we made by fixing the state of one of the robots is key in solving (7.8) and without this assumption, the system is not anchored and it will have no unique solution [205].

As a quick remark, in the previous results we notice that the final separation is $\frac{2}{3}\Delta < \Delta$, so to achieve the requirement for solving SpC, we can substitute Δ in (7.3) by $\frac{3}{2}\Delta$.

Minimally rigid graph

Here we omit one of the edges and we choose it to be the edge between nodes v_2 and v_3, as shown in Figure 7.1b. Therefore (7.8) becomes

$$\begin{aligned}(|x_2| - \Delta)\frac{x_2}{|x_2|} = 0 \\ (|x_3| - \Delta)\frac{x_3}{|x_3|} = 0\end{aligned} \tag{7.11}$$

similar to what we did earlier, we study the following two cases without loss of generality

Case 1: $x_1 \leq x_2 \leq x_3$**:** In this case we have a unique solution which is $\begin{bmatrix} x_2 & x_3 \end{bmatrix}^T = \begin{bmatrix} \Delta & \Delta \end{bmatrix}^T$

Case 2: $x_2 \leq x_1 \leq x_3$**:** In this case we have a unique solution which is $\begin{bmatrix} x_2 & x_3 \end{bmatrix}^T = \begin{bmatrix} -\Delta & \Delta \end{bmatrix}^T$

so we can notice from the solution of case 1 that robots # 2 and 3 could have the same state value, and still solve (7.11), meaning that taking out one edge of the complete graph could lead to states of different agents coinciding while carrying out the gradient decent control law (7.4) depending on initial conditions of the states. This does not happen when the graph is complete, regardless of the initial values of the states. The same result also holds for larger graphs with more nodes and edges. We note that the treatment adopted so far to find the equilibrium point resembles finding the equilibrium point of a system of masses and springs, for which a solution exists if the system is anchored, i.e. connected to a wall.

From the previous discussion, we observe that for a group of robots to be able to acquire the necessary situational awareness to coordinate their behaviors in a global manner, there is a need for a full communication graph among them. This graph completeness may not need to be direct, as it could be indirect as well (intermittent communication, gossip-based communication, etc), where the agents aggregate information from other agents with time. This could potentially lead to performance deterioration due to sampling speed, thus there is a need to study the boundaries of such delay to maintain acceptable performance when it comes to coordination among robots with time.

We conjecture that we can use some tools from sampled control theory and attempt to relate minimal graph structures (e.g. spanning trees in a network) with the maximum delay allowable for a network for coordination (e.g. temporal coordination) to work. Another possibility could also be finding ways with which agents can locally change their initial conditions, so that when (7.4) is applied, no multiple states claim the same value simultaneously.

7.2.2 Multiple charging stations

One important aspect of the wide-scale deployment of multiple robots in any environment is to cater to the energy needs of robots in a scalable manner by using multiple charging stations. Supposing that each charging station has one available charging slot at a time, the requirement of mutually exclusive use of charging stations has to be extended to multiple charging stations.

To this end, **FW1** should be able to handle spatial, as well as temporal, coordination. In other words, the robots should decide where, as well as when, to go to recharge to satisfy the mutually exclusive station use constraint. Moreover, this should be carried out in a decentralized fashion without the need for a central agent for decision-making.

In this section, we describe some ideas we attempted to achieve this goal and give some insights into tackling this problem. some work exists in the literature that deals with coordinating the use of multiple resources in an environment. One common way to deal with this issue is through the use of discrete optimization techniques [49] where the desired behavior is modeled as integer constraint in a mixed integer program, or through the use of market-based methods [48] where each robot picks a CS based on some preset rules, but provides no guarantees.

One approach we tried to extend **FW1** to achieve spatiotemporal coordination was to pursue the spatial and temporal aspects of coordination separately, with the hope of putting things together at a later stage. This calls for the need to extend the ES framework in **FW1** to help robots decide which station to go to, noting that the coordination framework in **FW1** already handles the temporal aspect of coordination. In the next section, we describe one idea we attempted that we call ES vector separation method.

ES vector separation

The goal of this method is to come up with a set of constraints that if satisfied, they seamlessly lead to robots choosing different stations, thus achieving spatial coordination.

Suppose we have two robots, each is described by single integrator dynamics $\dot{x} = u$, where $x \in \mathbb{R}^2$ is the robot's position, and $u \in \mathbb{R}^2$ is the control action. Moreover, consider we have two stations, and the energy sufficiency CBF for robot i towards station j is described as

$$h_{ij} = V - V_{min} - \frac{k_e}{v_r}(||x_i - x_{c_j}||) + \delta_{ij} \tag{7.12}$$

where $V > 0$ is the robot's voltage, $V_{min} > 0$ is the desired minimum voltage threshold, $k_e > 0$ is the battery discharge rate, v_r is the desired return speed of a robot towards the station, and x_{c_j} is the position of the charging station j. The variables $\delta_{ij} \in \mathbb{R}$ are design variables that could

be manipulated to affect the decision of a robot i of which station it picks. With this setup, each robot has a vector of different h_{ij} values towards each station (i.e. in this example we can say $h_i = \begin{bmatrix} h_{i1} & h_{i2} \end{bmatrix}^T$). Here we assume that robots have a constant linear discharge rate for their batteries.

Each robot picks the station with the higher h_{ij} value to go to (picks station corresponding to the dominant component in the h_i vector), this means that for these two robots to pick different stations, the goal is to make the dominant components for h_1 and h_2 different. Provided that the two vectors h_1 and h_2 start at the first quadrant (i.e. both robots can pick either station), if we can enforce a certain angle between these two vectors, then this goal can be achieved. We claim that there is a minimum value of inter-vector angle separation, we call it $\alpha_s \in [-\pi, \pi]$, that if satisfied, we can ensure that the dominant component in each vector will be different. The value of α_s depends on the size of the dimension we operate in, i.e. how many robots and stations we have. For the toy example of having two stations and two robots, the value of $\alpha_s = \frac{\pi}{4}$. This idea is demonstrated in Figure 7.2.

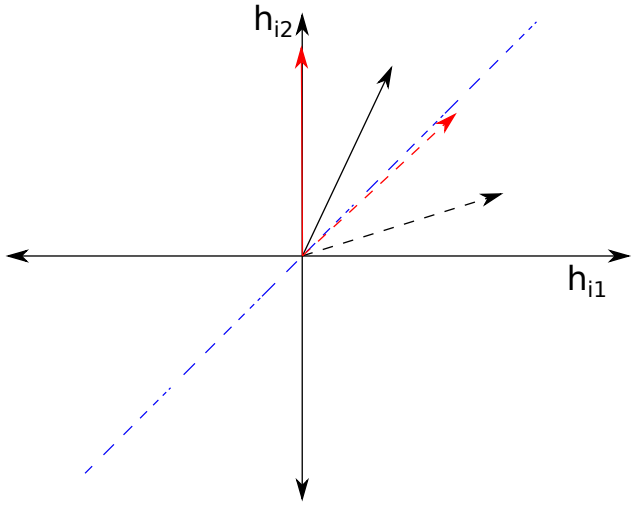


Figure 7.2 A vector parallel to the blue dotted-dashed line has both components equal, so no component dominates the other. The solid vectors are vectors for robot $i = 1$, and the dashed vectors are for robot $i = 2$. We compare angles of vectors of same color.

From Figure 7.2 we notice that if a vector lies on the blue dotted-dashed lines, it will have both components equal, thus no component dominating the other (i.e. the robot will be indecisive as of which station to pick). Therefore if a vector lies on either side of this blue line, it will have one dominating component over the other. Thus what we seek is to keep the two vectors pertaining to two robots on different sides of this blue line (for example like the black solid and dashed vectors in Figure 7.2).

If a vector starts aligned with the vertical axis (i.e. a robot starts knowing that it only chooses station 2), then the other vector has to be separated by an angle of at least $\frac{\pi}{4}$ to ensure it falls on the other side of the blue line (a vector aligned with the blue line has no dominant component), which justifies our choice of $\alpha_c = \frac{\pi}{4}$.

Therefore, a constraint could be constructed in the form of a CBF to ensure the angle between two vectors is greater than α_s

$$h_s = \cos^{-1} \frac{h_i.h_j}{||h_i||.||h_j||} - (\alpha_c + \epsilon) \tag{7.13}$$

where $\epsilon > 0$ is a small number. The main problem we faced with this method is the difficulty of generalizing the inter-vector separation conditions in higher dimensions. Moreover, the integration with the temporal coordination was not so straightforward.

Provided that we find a solution for the proper separation condition in higher dimensions, this method (separation of decision-related vectors in higher dimensional spaces) can serve as a new method to tackle task/resource allocation problems, while providing guarantees on solution convergence.

We conclude by giving some suggestions of ways to extend **FW1** to handle multiple CS:

- Figure out proper conditions for separating vectors in higher dimensional spaces in a manner similar to what we described earlier for 2d space, then understand how this can be integrated with the temporal coordination framework in **FW1**.
- Incorporate some rule-based methodology, like robots picking the nearest station, and letting robots assigned to this station coordinate temporally (e.g. Voronoi diagram with stations as seeds, and robots falling in same tessellation do temporal coordination).
- Use a blend of market-based methods and control barrier functions, by defining some form of price to pick a station. This price increases with less availability (i.e. more robots picking a particular station) and then use CBFs to manipulate these prices to ensure robots pick the right stations and do proper temporal coordination. Such an approach could incorporate methods from mechanism design in game theory, as well as CBFs.

7.2.3 Effect of CBF on mission execution

One central advantage of using CBFs with QP formulation during the development of the results in this thesis is achieving minimal intervention with the original mission's control action. Indeed, the QP cost function we adopt minimizes the difference between the final control action and that of the mission. However, there are some aspects related to the CBF constraint formulation that can affect

the quality of mission execution. In this section, we discuss briefly and qualitatively the relation between the CBF formulation and the mission quality. The main metric we adopt to judge mission execution quality is how long (or how far) a system can execute its mission control action before the safety-inducing control action is activated by the QP.

The value of the $\alpha(.)$ function in (2.7) can control how gracefully or aggressively a CBF maintains safety [47], which means that the choice of this function can affect how far the state of the system will be from the boundary of the safe set when the safety inducing control action will be activated by the QP. As a result, depending on the choice of this function, the system could abandon the mission earlier or later in a manner that can affect how long the mission control action is adopted, and thus the mission execution quality.

Another relevant aspect is the construction of the CBF itself. One clear example is the results we have shown in Chapter 6, especially Figure 6.12 which shows that choosing a higher return speed during the construction of the energy sufficiency CBF enances the overall quality of mission execution, as evident in the increase of the total explored area. This motivates the question of how to construct a CBF with the biggest safe set possible. Some methods that came out recently deal with this issue in various ways like using Hamilton Jacoobi reachability methods [90], supervised learning methods [97, 106], and reinforcement learning [99].

One interesting idea to investigate is to adopt a similar method of manipulating the $\alpha(.)$ functions in [84] using penalty functions, but instead of only doing this manipulation to ensure QP feasibility, we can link the $\alpha(.)$ functions to a mission performance related metric, and essentially manipulate them to give the system as much leeway to carry its mission as possible, and at the same time ensure the feasibility of the QP with different CBF constraints and input limits.

CHAPTER 8 CONCLUSION

8.1 Summary of Works

In this thesis, we present ES frameworks based on CBFs which provide guarantees that robots will be sufficiently energized during their missions. We developed two main frameworks:

- ES framework for multi-robot systems (**FW1**): developed in Chapters 4 and 5, and provides ES guarantees for robots in a multi-robot system carrying out a general mission, while coordinating the use of a common CS. We demonstrated that this framework has several favourable properties like flexibility and energy agnosticism.
- ES framework for robots in unknown environments (**FW2**): developed in Chapter 6 and provides guarantees on energy sufficiency for a robot carrying out a generic mission in a complex environment by adapting ES constraints to safe paths produced by path planners, and throughout our treatment we highlighted its various favourable properties like energy agnosticism, modularity and ability to operate with any path planner, and adaptability of the framework to different robot models.

Through the development of **FW1** we address the research objectives **O1** and **O2**. We first introduce **FW1** in Chapter 4 using a simpler battery discharge model and providing rudimentary results on CS capacity, and we extend the treatment in Chapter 5 with a more accurate battery discharge model and more developed results about CS capacity. We manage to show the efficacy of **FW1** through an experimental demonstration using Cognifly drones, along with various simulation results.

We tackle the research objective **O3** during our development of **FW2** in Chapter 6, where we extend ES guarantees over paths produced by path planners with the help of a simple sigmoid based path smoothing method, and apply the results on a unicycle type robot. We show the validity of **FW2** with results from an experimental run using an AgileX Scout Mini rover, as well as numerous simulation results.

8.2 Limitations

Herein we discuss some limitations of the current deliverables demonstrated in this thesis.

Energy sufficiency for multi-robot systems

Although our treatment for the ES framework for multi-robot systems presented in Chapters 4 and 5 showed that the framework offers a working solution for the problem of providing guarantees for energy sufficiency for a multi-robot systems sharing a CS, the treatment only considers robots operating in an environment without resistance, i.e. no wind fields affecting the drones. Although this could be taken into account by considering the worst discharge rate in case of an adverse wind, the result is often very conservative, and severely affects the station capacity while maintaining coordination constraints.

Another important limitation with **FW1** is that it is designed with having only one station in account. This can be very limiting in case of having an actual real life deployment in a real environment which may call for more than one station. Although in our treatment we can describe the capacity of a station, which is a useful sizing tool, we need to extend our framework to not only consider temporal coordination (who goes to station first) but also consider the spatial aspect (which station to go to). In section 7.2.2 we provide further remarks and insights of possible ways to achieve this goal.

The framework **FW1** is based on the assumption of a known discharge rate *a-priori*. Although such assumption is plausible, but highly dependent on prior experience with the robot system at hand or extensive empirical trials, which limits the applicability of our approach to a wider range of robots and can even threaten the robustness of **FW1** (otherwise a too pessimistic discharge rate needs to be adopted, which can compromise system performance).

We note that although the ES framework for multi-robot systems was developed with a simple non-cluttered environment in place, which can be perceived as a limitation, the application of our extended ES framework presented in Chapter 6 can solve this problem, moreover, the coordination framework from Chapters 4 and 5 could be used as is, with the use of voltage instead of energy in the formulation of CBF constraints.

Energy sufficiency in unknown environments

The treatment presented in Chapter 6 offers an extension to tackle to one central issue of **FW1**, via coupling ES constraint with path planning abilities to handle more complex environments (cluttered, unstructured, unknown, ...etc.). However, there are still some limitations in this work that could be mitigated in future work. The power model used in said framework is mostly empirical, and this means that it can change from one environment to another. One way this could be mitigated in the current setup of **FW2** is by adopting a more pessimistic power model, which can affect the quality of mission execution.

Another limitation is the fact that the performance of the framework is somewhat sensitive when a robot is returning back on a path. In other words, if a robot is returning back to station in compliance to the method outlined in Chapter 6 and got perturbed, the speed-power relationship can become unstable, especially at high desired return speeds.

8.3 Future work

In the light of our discussion of the current limitations for the frameworks presented in Chapters 4, 5 and 6 , we herein present some ideas and directions for further development.

- We plan to extend **FW2** to accommodate scenarios in which multiple robots coordinate the use of one charging station. This can be more challenging than the current coordination framework in **FW1** owing to the fact that robots' discharge is not uniform anymore (each robot can stop or move with arbitrary speeds during the mission) which makes it more challenging to estimate arrival times to a station.
- We plan to adopt adaptive and learning techniques in which a robot can infer the power model online during the mission to overcome the issue of inaccurate power models. Such power model should also be map dependent, in the sense of describing power consumption for every part of the map for more accurate representation of power consumption.
- The ES framework **FW2** needs to be modified to add extra robustness to the framework and to protect the system from getting to instability in case of disturbance along the path. This could be done by adopting a modified formulation that allows for having an extra amount of energy, we can call it buffer energy, that can compensate for disturbances when the robot is moving along the path. Moreover, learning techniques could be employed to estimate such energy quantity in a manner that does not affect the mission execution quality, but at the same time guarantees ES even in presence of occasional disturbances.
- Another interesting direction would be extending **FW1** and **FW2** to situations with multiple charging stations, with providing guarantees on spatiotemporal coordination.

REFERENCES

[1] N. Sturtevant, "Benchmarks for grid-based pathfinding," *Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 2, pp. 144 – 148, 2012. [Online]. Available: http://web.cs.du.edu/~sturtevant/papers/benchmarks.pdf

[2] H. Balta *et al.*, "Integrated data management for a fleet of search-and-rescue robots," *Journal of Field Robotics*, vol. 34, no. 3, pp. 539–582, 2017.

[3] M. Bajracharya, M. W. Maimone, and D. Helmick, "Autonomy for mars rovers: Past, present, and future," *Computer*, vol. 41, no. 12, pp. 44–50, 2008.

[4] S. Thrun *et al.*, "Autonomous exploration and mapping of abandoned mines," *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 79–91, 2004.

[5] P. Gonzalez de Santos* *et al.*, "Dylema: Using walking robots for landmine detection and location," *International journal of systems science*, vol. 36, no. 9, pp. 545–558, 2005.

[6] A. Botta *et al.*, "A review of robots, perception, and tasks in precision agriculture," *Applied Mechanics*, vol. 3, no. 3, pp. 830–854, 2022.

[7] G. Yang *et al.*, "Hallway exploration-inspired guidance: applications in autonomous material transportation in construction sites," *Automation in Construction*, vol. 128, p. 103758, 2021.

[8] H. Ardiny, S. Witwicki, and F. Mondada, "Construction automation with autonomous mobile robots: A review," in *2015 3rd RSI international conference on robotics and mechatronics (ICROM)*. IEEE, 2015, pp. 418–424.

[9] S. Grehl *et al.*, "Mining-rox–mobile robots in underground mining," in *Proceedings of the Third International Future Mining Conference, Sydney, Australia*, 2015, pp. 4–6.

[10] L. Marques *et al.*, "State of the art review on mobile robots and manipulators for humanitarian demining," 2016.

[11] M. K. Habib, "Humanitarian demining: reality and the challenge of technology–the state of the arts," *International Journal of Advanced Robotic Systems*, vol. 4, no. 2, p. 19, 2007.

[12] D. Portugal *et al.*, "An autonomous all terrain robotic system for field demining missions," *Robotics and Autonomous Systems*, vol. 70, no. C, pp. 126–144, 2015.

[13] Y. Li *et al.*, "Development and applications of rescue robots for explosion accidents in coal mines," *Journal of Field Robotics*, vol. 37, no. 3, pp. 466–489, 2020.

[14] P. Asgharian, A. M. Panchea, and F. Ferland, "A review on the use of mobile service robots in elderly care," *Robotics*, vol. 11, no. 6, p. 127, 2022.

[15] S. Wang and H. I. Christensen, "Tritonbot: First lessons learned from deployment of a long-term autonomy tour guide robot," in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2018, pp. 158–165.

[16] S. Thrun *et al.*, "Minerva: A second-generation museum tour-guide robot," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 3. IEEE, 1999.

[17] G. Best *et al.*, "Resilient multi-sensor exploration of multifarious environments with a team of aerial robots," in *Proceedings of Robotics: Science and Systems*, 2022.

[18] T. Zhang *et al.*, "The progress of extraterrestrial regolith-sampling robots," *Nature Astronomy*, vol. 3, no. 6, pp. 487–497, 2019.

[19] R. Doyle *et al.*, "Recent research and development activities on space robotics and ai," *Advanced Robotics*, vol. 35, no. 21-22, pp. 1244–1264, 2021.

[20] A. A. R. Newaz *et al.*, "Long-term autonomy for auvs operating under uncertainties in dynamic marine environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6313–6320, 2021.

[21] M. Veloso *et al.*, "Cobots: Robust symbiotic autonomous mobile service robots," in *Twenty-fourth international joint conference on artificial intelligence*. Citeseer, 2015.

[22] L. Sun *et al.*, "Robust and long-term monocular teach and repeat navigation using a single-experience map," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 2635–2642.

[23] H. Xie *et al.*, "Robust incremental long-term visual topological localization in changing environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–14, 2022.

[24] L. Kunze *et al.*, "Artificial intelligence for long-term robot autonomy: A survey," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4023–4030, 2018.

[25] R. Gourdeau and H. M. Schwartz, "Optimal control of a robot manipulator using a weighted time-energy cost function," in *Proceedings of the 28th IEEE Conference on Decision and Control,*. IEEE, 1989, pp. 1628–1631.

[26] R. Katoh *et al.*, "A real-time path planning of space manipulator saving consumed energy," in *Proceedings of IECON'94-20th Annual Conference of IEEE Industrial Electronics*, vol. 2. IEEE, 1994, pp. 1064–1067.

[27] Z. Sun and J. Reif, "On energy-minimizing paths on terrains for a mobile robot," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 3. IEEE, 2003, pp. 3782–3788.

[28] L. Wang and Y. Xiao, "A survey of energy-efficient scheduling mechanisms in sensor networks," *Mobile Networks and Applications*, vol. 11, no. 5, pp. 723–740, 2006.

[29] S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," in *ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01CH37240)*, vol. 2. IEEE, 2001, pp. 472–476.

[30] F. Ye *et al.*, "Peas: A robust energy conserving protocol for long-lived sensor networks," in *23rd International Conference on Distributed Computing Systems, 2003. Proceedings.* IEEE, 2003, pp. 28–37.

[31] B. Chen *et al.*, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wireless networks*, vol. 8, no. 5, pp. 481–494, 2002.

[32] J. Wu and S. Yang, "Coverage issue in sensor networks with adjustable ranges," in *Workshops on Mobile and Wireless Networking/High Performance Scientific, Engineering Computing/Network Design and Architecture/Optical Networks Control and Management/Ad Hoc and Sensor Networks/Compil*. IEEE, 2004, pp. 61–68.

[33] M. Cardei, J. Wu, and M. Lu, "Improving network lifetime using sensors with adjustable sensing ranges," *International Journal of Sensor Networks*, vol. 1, no. 1-2, pp. 41–49, 2006.

[34] Y. Mei *et al.*, "Deployment of mobile robots with energy and timing constraints," *IEEE Transactions on robotics*, vol. 22, no. 3, pp. 507–522, 2006.

[35] A. Kwok and S. Martinez, "Energy-balancing cooperative strategies for sensor deployment," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 6136–6141.

[36] W. Bentz and D. Panagou, "An energy-aware redistribution method for multi-agent dynamic coverage networks," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 2644–2651.

[37] T. Setter and M. Egerstedt, "Energy-constrained coordination of multi-robot teams," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1257–1263, 2016.

[38] D. Kularatne, S. Bhattacharya, and M. A. Hsieh, "Time and energy optimal path planning in general flows." in *Robotics: Science and Systems*, 2016.

[39] F. Michaud and E. Robichaud, "Sharing charging stations for long-term activity of autonomous robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3. IEEE, 2002, pp. 2746–2751.

[40] M. Silverman *et al.*, "Staying alive longer: Autonomous robot recharging put to the test," *Center for Robotics and Embedded Systems (CRES) Technical Report CRES*, vol. 3, p. 015, 2003.

[41] M. Rappaport, "Energy-aware mobile robot exploration with adaptive decision thresholds," in *Proceedings of ISR 2016: 47st International Symposium on Robotics*. VDE, 2016, pp. 1–8.

[42] M. Ramana, S. A. Varma, and M. Kothari, "Motion planning for a fixed-wing uav in urban environments," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 419–424, 2016.

[43] V. Berenz, F. Tanaka, and K. Suzuki, "Autonomous battery management for mobile robots based on risk and gain assessment," *Artificial Intelligence Review*, vol. 37, pp. 217–237, 2012.

[44] 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.

[45] G. Notomista and M. Egerstedt, "Persistification of robotic tasks," *arXiv preprint arXiv:1903.05810*, 2019.

[46] M. Egerstedt *et al.*, "Robot ecology: Constraint-based control design for long duration autonomy," *Annual Reviews in Control*, vol. 46, pp. 1–7, 2018.

[47] A. D. Ames *et al.*, "Control barrier functions: Theory and applications," in *2019 18th European control conference (ECC)*. IEEE, 2019, pp. 3420–3431.

[48] M. Rappaport and C. Bettstetter, "Coordinated recharging of mobile robots during exploration," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 6809–6816.

[49] M. Kenzin, I. Bychkov, and N. Maksimkin, "Self-adaptive coordination for fuel-constrained robot teams with periodic and aperiodic communications," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 236, no. 17, pp. 9730–9742, 2022.

[50] T. Gao and S. Bhattacharya, "Multirobot charging strategies: A game-theoretic approach," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2823–2830, 2019.

[51] A. R. Kan, *Machine scheduling problems: classification, complexity and computations*. Springer Science & Business Media, 2012.

[52] Y. Ding, W. Luo, and K. Sycara, "Decentralized multiple mobile depots route planning for replenishing persistent surveillance robots," in *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*. IEEE, 2019, pp. 23–29.

[53] T. Kundu and I. Saha, "Charging station placement for indoor robotic applications," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3029–3036.

[54] H. Hamann *et al.*, "Pick, pack, & survive: Charging robots in a modern warehouse based on online connected dominating sets," in *9th International Conference on Fun with Algorithms (FUN 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[55] N. Mathew, S. L. Smith, and S. L. Waslander, "A graph-based approach to multi-robot rendezvous for recharging in persistent tasks," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 3497–3502.

[56] G. Li, I. Svogor, and G. Beltrame, "Long-term pattern formation and maintenance for battery-powered robots," *Swarm Intelligence*, vol. 13, no. 1, pp. 21–57, 2019.

[57] D. Aksaray, C.-I. Vasile, and C. Belta, "Dynamic routing of energy-aware vehicles with temporal logic constraints," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3141–3146.

[58] A. Couture-Beil and R. T. Vaughan, "Adaptive mobile charging stations for multi-robot systems," in *2009 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2009, pp. 1363–1368.

[59] R. Alur, T. Dang, and F. Ivančić, "Progress on reachability analysis of hybrid systems using predicate abstraction," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2003, pp. 4–19.

[60] E. Asarin, T. Dang, and O. Maler, "The d/dt tool for verification of hybrid systems," in *International Conference on Computer Aided Verification*. Springer, 2002, pp. 365–370.

[61] A. Bemporad, F. D. Torrisi, and M. Morari, "Optimization-based verification and stability characterization of piecewise affine and hybrid systems," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2000, pp. 45–58.

[62] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on automatic control*, vol. 50, no. 7, pp. 947–957, 2005.

[63] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2004, pp. 477–492.

[64] S. Prajna, A. Jadbabaie, and G. J. Pappas, "Stochastic safety verification using barrier certificates," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 1. IEEE, 2004, pp. 929–934.

[65] S. Prajna and A. Rantzer, "On the necessity of barrier certificates," *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 526–531, 2005.

[66] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[67] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007.

[68] Z. Artstein, "Stabilization with relaxed controls," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 7, no. 11, pp. 1163–1173, 1983.

[69] E. D. Sontag, "A 'universal'construction of artstein's theorem on nonlinear stabilization," *Systems & control letters*, vol. 13, no. 2, pp. 117–123, 1989.

[70] M. Z. Romdlony and B. Jayawardhana, "Uniting control lyapunov and control barrier functions," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 2293–2298.

[71] 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*. IEEE, 2014, pp. 6271–6278.

[72] X. Xu *et al.*, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54–61, 2015.

[73] A. D. Ames *et al.*, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[74] H. K. Khalil, *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002, the book can be consulted by contacting: PH-AID: Wallet, Lionel. [Online]. Available: https://cds.cern.ch/record/1173048

[75] S.-C. Hsu, X. Xu, and A. D. Ames, "Control barrier function based quadratic programs with application to bipedal robotic walking," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 4542–4548.

[76] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 322–328.

[77] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," *arXiv preprint arXiv:1903.04706*, 2019.

[78] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 2659–2664.

[79] P. Glotfelter, J. Cortés, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *IEEE control systems letters*, vol. 1, no. 2, pp. 310–315, 2017.

[80] F. H. Clarke, *Optimization and nonsmooth analysis*. Siam, 1990, vol. 5.

[81] P. Glotfelter, J. Cortés, and M. Egerstedt, "Boolean composability of constraints and control synthesis for multi-robot systems via nonsmooth control barrier functions," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 897–902.

[82] X. Xu, "Constrained control of input–output linearizable systems using control sharing barrier functions," *Automatica*, vol. 87, pp. 195–201, 2018.

[83] J. Breeden and D. Panagou, "Compositions of multiple control barrier functions under input constraints," in *2023 American Control Conference (ACC)*. IEEE, 2023, pp. 3688–3695.

[84] W. Xiao, C. Belta, and C. G. Cassandras, "Adaptive control barrier functions," *IEEE Transactions on Automatic Control*, vol. 67, no. 5, pp. 2267–2281, 2021.

[85] S. Liu, W. Xiao, and C. A. Belta, "Auxiliary-adaptive control barrier functions for safety critical systems," *arXiv preprint arXiv:2304.00372*, 2023.

[86] P. Seiler, M. Jankovic, and E. Hellstrom, "Control barrier functions with unmodeled input dynamics using integral quadratic constraints," *IEEE Control Systems Letters*, vol. 6, pp. 1664–1669, 2021.

[87] J. Buch, S.-C. Liao, and P. Seiler, "Robust control barrier functions with sector-bounded uncertainties," *IEEE Control Systems Letters*, vol. 6, pp. 1994–1999, 2021.

[88] K. Garg and D. Panagou, "Robust control barrier and control lyapunov functions with fixed-time convergence guarantees," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 2292–2297.

[89] M. H. Cohen, C. Belta, and R. Tron, "Robust control barrier functions for nonlinear control systems with uncertainty: A duality-based approach," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 174–179.

[90] J. J. Choi *et al.*, "Robust control barrier–value functions for safety-critical control," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 6814–6821.

[91] A. J. Taylor and A. D. Ames, "Adaptive safety with control barrier functions," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1399–1405.

[92] Q. Nguyen and K. Sreenath, "L 1 adaptive control barrier functions for nonlinear underactuated systems," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 721–728.

[93] M. Maghenem *et al.*, "Adaptive safety using control barrier functions and hybrid adaptation," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 2418–2423.

[94] B. T. Lopez and J.-J. E. Slotine, "Unmatched control barrier functions: Certainty equivalence adaptive safety," in *2023 American Control Conference (ACC)*. IEEE, 2023, pp. 3662–3668.

[95] M. Krstić and P. V. Kokotović, "Control lyapunov functions for adaptive nonlinear stabilization," *Systems & Control Letters*, vol. 26, no. 1, pp. 17–23, 1995.

[96] B. T. Lopez, J.-J. E. Slotine, and J. P. How, "Robust adaptive control barrier functions: An adaptive and data-driven approach to safety," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 1031–1036, 2020.

[97] B. Dai, P. Krishnamurthy, and F. Khorrami, "Learning a better control barrier function," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 945–950.

[98] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe Learning of Quadrotor Dynamics Using Barrier Certificates," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 2460–2465, iSSN: 2577-087X.

[99] J. Choi *et al.*, "Reinforcement learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions," *arXiv preprint arXiv:2004.07584*, 2020.

[100] A. Taylor *et al.*, "Learning for Safety-Critical Control with Control Barrier Functions," in *Proceedings of the 2nd Conference on Learning for Dynamics and Control*. PMLR, Jul. 2020, pp. 708–717, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v120/taylor20a.html

[101] W. Jin *et al.*, "Neural certificates for safe control policies," *arXiv preprint arXiv:2006.08465*, 2020.

[102] R. Cheng *et al.*, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3387–3395.

[103] Y. Emam *et al.*, "Safe Reinforcement Learning Using Robust Control Barrier Functions," *IEEE Robotics and Automation Letters*, pp. 1–8, 2022, conference Name: IEEE Robotics and Automation Letters.

[104] S. McIlvanna *et al.*, "Reinforcement learning-enhanced control barrier functions for robot manipulators," *arXiv preprint arXiv:2211.11391*, 2022.

[105] M. Saveriano and D. Lee, "Learning Barrier Functions for Constrained Motion Planning with Dynamical Systems," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Macau, China: IEEE, Nov. 2019, pp. 112–119. [Online]. Available: https://ieeexplore.ieee.org/document/8967981/

[106] A. Robey *et al.*, "Learning Control Barrier Functions from Expert Demonstrations," in *2020 59th IEEE Conference on Decision and Control (CDC)*, Dec. 2020, pp. 3717–3724, iSSN: 2576-2370.

[107] M. Srinivasan *et al.*, "Synthesis of Control Barrier Functions Using a Supervised Machine Learning Approach," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2020, pp. 7139–7145, iSSN: 2153-0866.

[108] Q. Shi *et al.*, "Adaptive pid controller based on q-learning algorithm," *CAAI Transactions on Intelligence Technology*, vol. 3, no. 4, pp. 235–244, 2018.

[109] E. Squires, P. Pierpaoli, and M. Egerstedt, "Constructive barrier certificates with applications to fixed-wing aircraft collision avoidance," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 1656–1661.

[110] T. Gurriet *et al.*, "Towards a framework for realizable safety critical control through active set invariance," in *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*. IEEE Press, 2018, pp. 98–106.

[111] 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.

[112] U. Borrmann *et al.*, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68–73, 2015.

[113] A. Li *et al.*, "Formally correct composition of coordinated behaviors using control barrier certificates," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3723–3729.

[114] Q. Nguyen and K. Sreenath, "Safety-critical control for dynamical bipedal walking with precise footstep placement," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 147–154, 2015.

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

[116] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[117] C. Pinciroli *et al.*, "Argos: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.

[118] C. Pinciroli and G. Beltrame, "Buzz: An extensible programming language for heterogeneous swarm robotics," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 3794–3800.

[119] http://ftp.k-team.com/KheperaIV/ [Accessed: 2010], 2010.

[120] https://optitrack.com/ [Accessed: 2023], 2023.

[121] https://github.com/MISTLab/BuzzKH4/tree/nightly.

[122] R. de Azambuja *et al.*, "When Being Soft Makes You Tough: A Collision-Resilient Quadcopter Inspired by Arthropods' Exoskeletons," *arXiv e-prints*, p. arXiv:2103.04423, Mar. 2021.

[123] https://github.com/Hassan-Ali-Hassan/BuzzCognifly.

[124] https://global.agilex.ai/products/scout-mini [Accessed: Jun 2023], 2023.

[125] A. Kwok and S. Martinez, "Energy-balancing cooperative strategies for sensor deployment," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 6136–6141.

[126] J. Wang *et al.*, "Optimization design of wireless charging system for autonomous robots based on magnetic resonance coupling," *Aip Advances*, vol. 8, no. 5, p. 055004, 2018.

[127] C. Pinciroli, A. Lee-Brown, and G. Beltrame, "Buzz: An Extensible Programming Language for Self-Organizing Heterogeneous Robot Swarms," *arXiv e-prints*, p. arXiv:1507.05946, Jul. 2015.

[128] A. Stancovici, M. V. Micea, and V. Cretu, "Cooperative positioning system for indoor surveillance applications," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2016, pp. 1–7.

[129] R. Mendonça *et al.*, "A cooperative multi-robot team for the surveillance of shipwreck survivors at sea," in *OCEANS 2016 MTS/IEEE Monterey*. IEEE, 2016, pp. 1–6.

[130] J. Gregory *et al.*, "Application of multi-robot systems to disaster-relief scenarios with limited communication," in *Field and Service Robotics*. Springer, 2016, pp. 639–653.

[131] L. E. Parker, "Alliance: An architecture for fault tolerant multirobot cooperation," *IEEE transactions on robotics and automation*, vol. 14, no. 2, pp. 220–240, 1998.

[132] J. Cortes *et al.*, "Coverage control for mobile sensing networks," *IEEE Transactions on robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[133] S. Thrun and Y. Liu, "Multi-robot slam with sparse extended information filers," in *Robotics Research. The Eleventh International Symposium*. Springer, 2005, pp. 254–266.

[134] A. Munoz *et al.*, "Sharing a charging station in collective robotics," *LIP6 research reports*, 2002.

[135] K. Iftikhar and M. T. Khan, "Resource sharing and deadlock avoidance in multi robot systems using market based approach," in *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 2016, pp. 1–7.

[136] H. Fouad and G. Beltrame, "Energy autonomy for resource-constrained multi robot missions," in *International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[137] G. Notomista and M. Egerstedt, "Persistification of robotic tasks," *IEEE Transactions on Control Systems Technology*, 2020.

[138] Y. Mulgaonkar and V. Kumar, "Autonomous charging to enable long-endurance missions for small aerial robots," in *Micro-and Nanotechnology Sensors, Systems, and Applications VI*, vol. 9083. International Society for Optics and Photonics, 2014, p. 90831S.

[139] R. Cassinis *et al.*, "Docking and charging system for autonomous mobile robots," *Department of Electronics for Automation, University of Brescia, Italy*, 2005.

[140] A. B. Junaid *et al.*, "Autonomous wireless self-charging for multi-rotor unmanned aerial vehicles," *Energies*, vol. 10, no. 6, p. 803, 2017.

[141] A. Rohan *et al.*, "Advanced drone battery charging system," *Journal of Electrical Engineering & Technology*, vol. 14, no. 3, pp. 1395–1405, 2019.

[142] D. Lee, J. Zhou, and W. T. Lin, "Autonomous battery swapping system for quadcopter," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 118–124.

[143] N. K. Ure *et al.*, "An automated battery management system to enable persistent missions with multiple aerial vehicles," *IEEE/ASME transactions on mechatronics*, vol. 20, no. 1, pp. 275–286, 2014.

[144] A. Forsgren, P. E. Gill, and M. H. Wright, "Interior methods for nonlinear optimization," *SIAM review*, vol. 44, no. 4, pp. 525–597, 2002.

[145] Z. Arstein, "Stabilization with relaxed controls," *Nonlinear analysis*, vol. 7, no. 11, pp. 1163–1173, 1983.

[146] 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*. IEEE, 2014, pp. 6271–6278.

[147] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE transactions on robotics*, vol. 25, no. 4, pp. 912–926, 2009.

[148] G. Notomista *et al.*, "Passivity-based decentralized control of multi-robot systems with delays using control barrier functions," in *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*. IEEE, 2019, pp. 231–237.

[149] R. de Azambuja, H. Fouad, and G. Beltrame, "When being soft makes you tough: A collision resilient quadcopter inspired by arthropod exoskeletons," *arXiv preprint arXiv:2103.04423*, 2021.

[150] P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multi-agent coordination," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, vol. 2. IEEE, 2001, pp. 1150–1155.

[151] P. Glotfelter and M. Egerstedt, "A parametric mpc approach to balancing the cost of abstraction for differential-drive mobile robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 732–737.

[152] M. F. J. Morales and J. B. G. Mendoza, "Mixed energy model for a differential guide mobile robot," in *2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*. IEEE, 2018, pp. 114–119.

[153] G. E. Cushing *et al.*, "Candidate cave entrances on mars," *Journal of Cave and Karst Studies*, vol. 74, no. 1, pp. 33–47, 2012.

[154] T. Titus *et al.*, "Science and technology requirements to explore caves in our solar system," *Bulletin of the American Astronomical Society*, vol. 53, no. 4, 2021.

[155] A. Ravankar *et al.*, "Multi-robot path planning for smart access of distributed charging points in map," *Artificial Life and Robotics*, vol. 26, no. 1, pp. 52–60, 2021.

[156] F. Liu, S. Liang, and D. X. Xian, "Optimal path planning for mobile robot using tailored genetic algorithm," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, no. 1, pp. 1–9, 2014.

[157] H. Fouad and G. Beltrame, "Energy autonomy for robot systems with constrained resources," *IEEE Transactions on Robotics*, 2022.

[158] N. Mathew, S. L. Smith, and S. L. Waslander, "Multirobot rendezvous planning for recharging in persistent tasks," *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 128–142, 2015.

[159] T. Kundu and I. Saha, "Mobile recharger path planning and recharge scheduling in a multi-robot environment," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3635–3642.

[160] N. Kamra, T. S. Kumar, and N. Ayanian, "Combinatorial problems in multirobot battery exchange systems," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 2, pp. 852–862, 2017.

[161] M. Alizadeh *et al.*, “Optimized path planning for electric vehicle routing and charging,” in *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2014, pp. 25–32.

[162] F. Fu and H. Dong, “Targeted optimal-path problem for electric vehicles with connected charging stations,” *PLOS ONE*, vol. 14, no. 8, p. e0220361, Aug. 2019. [Online]. Available: https://dx.plos.org/10.1371/journal.pone.0220361

[163] M. Schneider, A. Stenger, and D. Goeke, “The electric vehicle-routing problem with time windows and recharging stations,” *Transportation science*, vol. 48, no. 4, pp. 500–520, 2014.

[164] B. Li *et al.*, “Planning large-scale search and rescue using team of uavs and charging stations,” in *2018 IEEE international symposium on safety, security, and rescue robotics (SSRR)*. IEEE, 2018, pp. 1–8.

[165] T. Wang *et al.*, “Staying-alive and energy-efficient path planning for mobile robots,” in *2008 American Control Conference*, Jun. 2008, pp. 868–873, iSSN: 2378-5861.

[166] Y. Warsame, S. Edelkamp, and E. Plaku, “Energy-aware multi-goal motion planning guided by monte carlo search,” in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2020, pp. 335–342.

[167] DARPA, “DARPA Subterranean Challenge,” https://www.subtchallenge.com, 2018, online; accessed 8 March 2023.

[168] T. Dang *et al.*, “Graph-based path planning for autonomous robotic exploration in subterranean environments,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 3105–3112.

[169] A. Bircher *et al.*, “Receding horizon" next-best-view" planner for 3d exploration,” in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1462–1468.

[170] M. Dharmadhikari *et al.*, “Motion primitives-based path planning for fast and agile exploration using aerial robots,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 179–185.

[171] H. Zhu *et al.*, “Dsvp: Dual-stage viewpoint planner for rapid exploration by dynamic expansion,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 7623–7630.

[172] C. Cao *et al.*, "Tare: A hierarchical framework for efficiently exploring complex 3d environments." in *Robotics: Science and Systems*, vol. 5, 2021.

[173] 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)*. IEEE, 2020, pp. 7006–7013.

[174] S. S. Mehta *et al.*, "Human-assisted rrt for path planning in urban environments," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, 2015, pp. 941–946.

[175] M. Fu *et al.*, "Path planning and decision making for autonomous vehicle in urban environment," in *2015 IEEE 18th international conference on intelligent transportation systems*. IEEE, 2015, pp. 686–692.

[176] K. Zhao and Y. Li, "Path planning in large-scale indoor environment using rrt," in *Proceedings of the 32nd chinese control conference*. IEEE, 2013, pp. 5993–5998.

[177] O. Souissi *et al.*, "Path planning: A 2013 survey," in *Proceedings of 2013 International Conference on Industrial Engineering and Systems Management (IESM)*. IEEE, 2013, pp. 1–8.

[178] B. Patle *et al.*, "A review: On path planning strategies for navigation of mobile robot," *Defence Technology*, vol. 15, no. 4, pp. 582–606, 2019.

[179] P. Jaroszek and M. Trojnacki, "Model–based energy efficient global path planning for a four–wheeled mobile robot," *Control and Cybernetics*, vol. 43, no. 2, 2014.

[180] V. Gruning *et al.*, "Energy-aware path planning for skid-steer robots operating on hilly terrain," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2094–2099.

[181] B. Hao *et al.*, "Automatic recharging path planning for cleaning robots," *Mobile Information Systems*, vol. 2021, 2021.

[182] A. Ravankar *et al.*, "Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges," *Sensors*, vol. 18, no. 9, p. 3170, 2018.

[183] R. Cimurs, J. Hwang, and I. H. Suh, "Bezier curve-based smoothing for path planner with curvature constraint," in *2017 First IEEE International Conference on Robotic Computing (IRC)*. IEEE, 2017, pp. 241–248.

[184] K. R. Simba, N. Uchiyama, and S. Sano, "Real-time smooth trajectory generation for nonholonomic mobile robots using bézier curves," *Robotics and Computer-Integrated Manufacturing*, vol. 41, pp. 31–42, 2016.

[185] I. Noreen, "Collision free smooth path for mobile robots in cluttered environment using an economical clamped cubic b-spline," *Symmetry*, vol. 12, no. 9, p. 1567, 2020.

[186] M. Elhoseny, A. Tharwat, and A. E. Hassanien, "Bezier curve based path planning in a dynamic field using modified genetic algorithm," *Journal of Computational Science*, vol. 25, pp. 339–350, 2018.

[187] H. A. Satai *et al.*, "Bézier curves-based optimal trajectory design for multirotor uavs with any-angle pathfinding algorithms," *Sensors*, vol. 21, no. 7, p. 2460, 2021.

[188] J. Wu and V. Snášel, "A bezier curve-based approach for path planning in robot soccer," in *Innovations in Bio-inspired Computing and Applications*. Springer, 2014, pp. 105–113.

[189] S. Zhao and Z. Sun, "Defend the practicality of single-integrator models in multi-robot coordination control," in *2017 13th IEEE International Conference on Control & Automation (ICCA)*. IEEE, 2017, pp. 666–671.

[190] H. Oleynikova *et al.*, "Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[191] T. Shan *et al.*, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5692–5698.

[192] L. Lu, "Explicit algorithms for multiwise merging of bézier curves," *Journal of computational and applied mathematics*, vol. 278, pp. 138–148, 2015.

[193] M. Mesbahi and M. Egerstedt, "Graph theoretic methods in multiagent networks," in *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[194] W. Ren *et al.*, "Experimental validation of consensus algorithms for multivehicle cooperative control," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 4, pp. 745–752, 2008.

[195] T. Halsted *et al.*, "A survey of distributed optimization methods for multi-robot systems," *arXiv preprint arXiv:2103.12840*, 2021.

[196] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on automatic control*, vol. 51, no. 3, pp. 401–420, 2006.

[197] W. Ren and N. Sorensen, "Distributed coordination architecture for multi-robot formation control," *Robotics and Autonomous Systems*, vol. 56, no. 4, pp. 324–333, 2008.

[198] M. Alighanbari and J. P. How, "Decentralized task assignment for unmanned aerial vehicles," in *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 5668–5673.

[199] D. Dionne and C. A. Rabbath, "Multi-uav decentralized task allocation with intermittent communications: The dtc algorithm," in *2007 American Control Conference*. IEEE, 2007, pp. 5406–5411.

[200] B. P. Gerkey and M. J. Mataric, "Sold!: Auction methods for multirobot coordination," *IEEE transactions on robotics and automation*, vol. 18, no. 5, pp. 758–768, 2002.

[201] Z. Yan, N. Jouandeau, and A. A. Cherif, "Multi-robot decentralized exploration using a trade-based approach," in *8th International Conference on Informatics in Control, Automation and Robotics*, 2011.

[202] L. Brunet, H.-L. Choi, and J. How, "Consensus-based auction approaches for decentralized task assignment," in *AIAA guidance, navigation and control conference and exhibit*, 2008, p. 6839.

[203] G. P. Kontoudis and D. J. Stilwell, "Decentralized nested gaussian processes for multi-robot systems," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 8881–8887.

[204] Z. Sun *et al.*, "Exponential stability for formation control systems with generalized controllers: A unified approach," *Systems & Control Letters*, vol. 93, pp. 50–57, 2016.

[205] G. Strang, "Computational science and engineering," *Optimization*, vol. 551, no. 563, pp. 571–586, 2007.

[206] V. H. Johnson, A. A. Pesaran, and T. Sack, "Temperature-dependent battery models for high-power lithium-ion batteries," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2001.

APPENDIX A SOME NOTES ON SINGLE INTEGRATOR TO UNICYCLE TRANSFORMATION

In this appendix we tackle some issues related to applying our proposed framework on differential drive robots modelled using a unicycle model as (5.43). The main difference with a unicycle model, as will be shown in the next section, is the fact that input directly affects the rate of voltage consumption. Although the voltage model used in (5.6) assumes a constant discharge rate, we show that our proposed methods can nonetheless on a full unicycle model.

We start by mathematically describing the dependence of voltage discharge rate on unicycle dynamics and the relationship between the single integrator to unicycle transformation (5.44) and voltage discharge. We then present in the section that follows a case study of applying energy sufficiency constraint on a unicycle model to serve as an example of how to adapt the energy sufficiency method encoded by the CBF in (5.7) to unicycle dynamics.

We leave the full extension of our proposed framework to a multi-robot system with unicycle model (including energy sufficiency and multi-robot coordination) as a future work, while presenting the energy augmented unicycle model and the case study as motivating points for such future work.

Effect of single integrator to unicycle transformation on power consumption

As we mentioned in remark 11, changing the value of l in the transformation (5.44) may affect the rate of power consumption. In this section we demonstrate this effect using a modified version of the energy augmented dynamic model of a differential drive robot in [152], then we apply the output of transformation (5.44) to this model for some nominal mission control action $\mathbf{u}_{nom}$, and we use the electric current value from the model to show the voltage change of a battery modelled as in [206].

In [152] the motor dynamics were neglected to produce a simpler model. However, this undermines the effect of some motor properties on overall performance. We modify the model in [152] by taking the motor dynamics into consideration. We start by stating the main result in [152] after the modification. The robot model can be described as

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{q} \\ \dot{\eta} \\ \dot{I} \end{bmatrix} = \begin{bmatrix} S\eta \\ -(S^T M S)^{-1} S^T (C + FS\eta + M\dot{S}\eta - TI) \\ -\frac{K_w}{L}\eta - \frac{R}{L} I \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{6\times 2} \\ \frac{1}{L}\mathbf{I}_{2\times 2} \end{bmatrix} \mathbf{V} \tag{A.1}$$

where $q = \begin{bmatrix} x_1 & x_2 & \phi_1 & \phi_2 \end{bmatrix}^T$ is the state vector, (x_1, x_2) are the robot's coordinates, ϕ_i are the wheels' angles, $\eta = \begin{bmatrix} \dot{\phi}_1 & \dot{\phi}_2 \end{bmatrix}^T$ is the vector of wheels' angular speeds, $I = \begin{bmatrix} I_1 & I_2 \end{bmatrix}^T$ are the motors' currents, $\mathbf{I}_{2\times 2}$ is an identity matrix, $C = \mathbf{0}_{4\times 1}$, and $\mathbf{V} \in \mathbb{R}^2$ is the vector containing motors' armature voltage values. The matrices S, M, F, T are defined as follows:

$$F = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & K \end{bmatrix}, \quad K = \begin{bmatrix} \nu & 0 \\ 0 & \nu \end{bmatrix}$$

$$T = \begin{bmatrix} 0 & 0 & K_t & 0 \\ 0 & 0 & 0 & K_t \end{bmatrix}^T$$

$$S = \begin{bmatrix} cb\cos\theta & cb\cos\theta \\ cb\sin\theta & cb\sin\theta \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$M(q) = \begin{bmatrix} m & 0 & -\alpha_1 & \alpha_1 \\ 0 & m & \alpha_2 & -\alpha_2 \\ -\alpha_1 & \alpha_2 & I_c^2 + I_w + I_s & -I_c^2 \\ \alpha_1 & -\alpha_2 & -I_c^2 & I_c^2 + I_w + I_s \end{bmatrix}$$

with $\alpha_1 = m_c cd\sin\theta$, $\alpha_2 = m_c cd\cos\theta$. Table A.1 contains definitions of different variables and their values.

To understand the effect of l in (5.44), we need to generate voltages that lead the wheels to spin with speeds that achieve a desired single integrator point speed. Suppose we have a desired velocity $\mathbf{u} \in \mathbb{R}^2$ for a single integrator robot, then the desired linear and angular velocities for the unicycle robot can be obtained from (5.44), from which the desired speeds for both motors can be expressed

Table A.1 Definitions of different parameters used in (A.1)

variable	definition	value
R	motor armature resistance	20Ω
K_t	motor torque constant	$0.88\ N.m/A$
K_w	motor voltage constant	$0.88\frac{V.sec}{rad}$
I_s	motor armature inertia	$0.0713\ \frac{Kg}{m^2}$
b	distance from wheel to center line	$0.165m$
c	$c = \frac{r}{2b}$, r being wheel radius	$0.8684m$
m_c	mass of robot without wheels or motors	$6.04Kg$
m_w	mass of each wheel and its motor	$1.48Kg$
I_c	robot's moment of inertia around its vertical axis[1]	$0.537\frac{Kg}{m^2}$
I_w	moment of inertia of a wheel and its motor around motor's axis	$0.0023\frac{Kg}{m^2}$

as

$$\eta_d = \begin{bmatrix} \dot{\phi}_{d_1} \\ \dot{\phi}_{d_2} \end{bmatrix} = \underbrace{\frac{1}{r}\begin{bmatrix} 1 & b \\ 1 & -b \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l} \end{bmatrix}}_{B}\underbrace{\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}}_{\mathcal{R}(\theta)}\mathbf{u} \tag{A.2}$$

the error between the desired and actual wheel speed is

$$e = \eta - \eta_d \tag{A.3}$$

which can be used as an input for a controller to generate the voltage required for each motor to achieve the desired rotational speed. For example, we can use a proportional controller, thus the voltage V in (A.1) is

$$V = -K_p e \tag{A.4}$$

plugging this voltage into (A.1) leads to driving the unicycle robot according to $\mathbf{u}$. The closed loop system now becomes

$$\dot{\mathbf{x}} = \begin{bmatrix} S\eta \\ -(S^T M S)^{-1} S^T (C + FS\eta + M\dot{S}\eta - TI) \\ -\frac{K_w + K_p}{L}\eta - \frac{R}{L}I \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{6\times 2} \\ \frac{Kp}{L} B\mathcal{R}(\theta) \end{bmatrix}\mathbf{u} \tag{A.5}$$

This result describes a unicycle robot model that takes single integrator velocity as input. The total current needed to cater for the desired motors' rotational speed is

$$I_t = \mathbf{1}^T |I| \tag{A.6}$$

with $\mathbf{1} \in \mathbb{R}^2$ is a vector of all ones and $|.|$ is the element-wise absolute value of a vector. This current value can be used as an input for a simple battery voltage model, as in [206] for example, to compute the voltage change of the battery. From [206], the battery voltage can be modelled as

$$\begin{aligned} \begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cc} \end{bmatrix} &= \frac{1}{R_e+R_c} \begin{bmatrix} -\frac{1}{C_b} & \frac{1}{C_b} \\ \frac{1}{C_c} & -\frac{1}{C_c} \end{bmatrix} \begin{bmatrix} V_{Cb} \\ V_{Cc} \end{bmatrix} \\ &\quad + \begin{bmatrix} \frac{-R_c}{C_b(R_e+R_c)} \\ -\frac{1}{C_c} + \frac{R_c}{C_c(R_e+R_c)} \end{bmatrix} I_t \\ V_o &= \frac{1}{R_e+R_c} \begin{bmatrix} R_c & R_e \end{bmatrix} \begin{bmatrix} V_{Cb} \\ V_{Cc} \end{bmatrix} + \left[R_t - \frac{R_e R_c}{R_e+R_c} \right] I_t \end{aligned} \tag{A.7}$$

which we can represent as $\dot{\mathbf{V}}_b = A_v \mathbf{V}_b + B_v I_t, V_o = C_v \mathbf{V}_b + D_v I_t$, with V_o being the output voltage of the battery. Values of different battery parameters are presented in Table A.2.

Table A.2 Values of battery internal resistance and capacitance values used in the battery model.

C_c	C_b	R_c	R_e	R_t
40.74	820.00	.0004	.0011	.0012

For demonstration purpose, we consider the case in which a robot moves along a square with rounded corners for which $\mathbf{u} = \mathbf{u}_{nom}$, where $\mathbf{u}_{nom}$ is defined as

$$\mathbf{u}_{nom} = K_d(x - x_d) \tag{A.8}$$

with

$$x_d(t) = \alpha \left(\cos^\beta(\omega t) + \cos^\beta(\omega t) \right)^{\frac{1}{\beta}} \begin{bmatrix} \cos(\omega t) \\ \sin(\omega t) \end{bmatrix}$$

where $x \in \mathbb{R}^2$ is the robot's position, $x_d \in \mathbb{R}^2$ is a desired position along the nominal trajectory, $\alpha > 0, \beta > 0$. For our demonstration we choose $\omega = -0.2 rad/sec$, $\alpha = 4m$ and $\beta = 100$. Figure A.1a shows the trajectory and voltage change for a robot tracking the nominal trajectory, for three different values of l in (A.2).

From Figure A.1b above we notice that the smaller the value of l gets, the less oscillatory and more responsive the robot becomes, and the higher the voltage decline rate is due to the more aggressive

nature of its response. The dotted lines in A.1b represent the lower bound of the voltage change, which can be used in our proposed framework.

We note that using a full unicycle model like (A.5) with our proposed framework gives rise to an additional aspect which is the direct effect of the input on voltage change and power consumption. One strategy to tackle this issue is to use the framework as proposed and add extra constraints on the electric current values to make sure that the change in voltage is always bounded from below. In what follows we present an example of applying this idea to the full unicycle model.

Case study

One idea to use the full unicycle model is to set a constraint on the maximum electric current being used, thus limiting the rate of voltage decrease. This constraint is set by establishing an upper bound on the velocity that corresponds to the electric current limit we impose.

Effect of limiting electric current on speed

Suppose that the robot is moving along a straight line such that the electric current is at the set limit and that it is moving in a steady state manner, we then can set all time varying terms in the equation for $\dot{\eta}$ in (A.5) to be equal to zero, thus

$$\begin{aligned} &FS\bar{\eta} = T\bar{I} \\ &\bar{\eta} = \left((FS)^T FS\right)^{-1} (FS)^T T\bar{I} \\ &\Rightarrow \bar{\eta} = \frac{K_t}{\nu}\bar{I} \end{aligned} \tag{A.9}$$

with $\bar{I}$ and $\bar{\eta}$ being the steady state values of electric currents and wheels' angular speeds. To produce the constraint on electric current, we can use the following candidate CBF

$$h_I = I_{\max} - ||I|| \tag{A.10}$$

which gives the constraint

$$\begin{aligned} &-\frac{I^T}{||I||}\dot{I} \geq -p_I h_I \\ &\Rightarrow -\tfrac{I^T}{||I||}\left(-\tfrac{K_p+K_w}{L}\eta - \tfrac{R}{L}I + \tfrac{K_p}{L}B\mathcal{R}(\theta)\mathbf{u}\right) \geq -p_I h_I \end{aligned} \tag{A.11}$$

with $p_I > 0$. Suppose that robot is moving in steady state with $||I|| = I_{\max}$ (i.e. $h_I = 0$) we can substitute (A.9) in the last inequality and we get

$$B\mathcal{R}(\theta)\mathbf{u} \leq \underbrace{\left(\frac{(K_p + K_w)K_t}{K_p\nu} + \frac{R}{K_p}\right)}_{\Gamma} \bar{I} \tag{A.12}$$

and since we consider the robot moving in a straight line, $\mathcal{R}(\theta)$ is actually an identity matrix and the last equation can be further simplified to

$$\mathbf{u} \leq \Gamma \frac{r}{2b} \begin{bmatrix} b & b \\ l & -l \end{bmatrix} \bar{I} \tag{A.13}$$

we can use the last relation to estimate the maximum magnitude of velocity attainable while respecting the electric current constraint. Supposing the robot is moving in a straight line and that the robot is symmetric with motors having same properties, then $\bar{I} = \frac{I_{\max}}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, and the last relation becomes

$$\begin{aligned} \mathbf{u} &\leq \frac{\Gamma r}{\sqrt{2}} I_{\max} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \Rightarrow ||\mathbf{u}|| &\leq \frac{\Gamma r}{\sqrt{2}} I_{\max} \end{aligned} \tag{A.14}$$

Effect of limiting electric current on voltage change

The other aspect that is affected by limiting maximum current consumption is the voltage decrease rate, noting that the primary purpose of constraining the current is to have a known worst-case rate of voltage decrease. One step in this direction is to understand the effect of a generic current input on the battery discharge rate.

We note that the system matrix in (A.7) is not full rank and it has an eigenvalue that is equal to zero, which is the dominant mode of the system, associated with an all-ones eigenvector. We focus on estimating the effect of electric current on this dominant mode by doing a similarity transformation $\mathbf{V}_b = \mathcal{T} z$, where $\mathbf{V}_b$ is the battery voltages vector in (A.7), z is a set of new states and $\mathcal{T}$ is a transformation matrix comprised of the eigenvectors of the system matrix in (A.7) to obtain the

modal canonical form. The new transformed system is

$$\begin{aligned}\dot{z} &= \overbrace{\begin{bmatrix}0 & 0\\ 0 & \lambda\end{bmatrix}}^{\mathcal{T}^{-1}A_v\mathcal{T}} + \mathcal{T}^{-1}B_v I_t \\ V_o &= C_v\mathcal{T}z + D_v I_t\end{aligned} \tag{A.15}$$

where $\lambda = -\frac{1}{R_e+R_c}\left(\frac{1}{C_b}+\frac{1}{C_c}\right)$ and the $\mathcal{T}$ matrix can be easily calculated to be

$$\mathcal{T} = \begin{bmatrix}\frac{1}{\sqrt{2}} & -\frac{C_c}{\gamma C_b}\\ \frac{1}{\sqrt{2}} & \frac{1}{\gamma}\end{bmatrix}$$

where $\gamma = \sqrt{1+\left(\frac{C_c}{C_b}\right)^2}$. The dynamics of the state associated with zero eigenvalue is

$$\dot{z}_1 = \frac{\sqrt{2}\gamma C_b}{C_c+C_b}\left(\frac{1}{\gamma}B_{v_1} + \frac{C_c}{\gamma C_b}B_{v_2}\right) I_t$$

and supposing that the robot is moving in steady state along a straight line while maintaining the electric current constraint (A.11), we observe that the effect of the input current is that it changes the slope of z_1. Since we take $I_t = \mathbf{1}^T|I|$, the maximum current for the aforementioned scenario (i.e. moving in steady state with both motors drawing the same current) is $I_t = 2\frac{I_{\max}}{\sqrt{2}} = \sqrt{2}I_{\max}$ and thus the previous relation becomes

$$\dot{z}_1 = \frac{2\gamma C_b}{C_c+C_b}\left(\frac{1}{\gamma}B_{v_1} + \frac{C_c}{\gamma C_b}B_{v_2}\right) I_{\max} \tag{A.16}$$

and the battery output becomes (taking only the dominant mode)

$$V_o = \left(\frac{C_{v_1}}{\sqrt{2}}+\frac{C_{v_2}}{\sqrt{2}}\right) z_1 + \sqrt{2}D_v I_{\max} = \sqrt{2}z_1 + \sqrt{2}D_v I_{\max} \tag{A.17}$$

Example of energy sufficiency with the unicycle model

Here we use the previous two results, i.e. the effect of limiting electric current on speed and voltage change, to add an energy sufficiency layer to the perimeter patrol task shown in Figure A.1. We use the following zeroing CBF

$$h_s = V_o - V_{\min} - \frac{k_e}{v_r}(D_{cs} - \delta) \tag{A.18}$$

where $D_{cs} = ||x - x_{cs}|| \geq 0$ is the distance from the charging station, $\delta > 0$ is a threshold distance away from the center of the charging region and k_e is the rate of voltage change from (A.16) and is represented as

$$k_e = \frac{2\sqrt{2}\gamma C_b}{C_c + C_b}\left(\frac{1}{\gamma}B_{v_1} + \frac{C_c}{\gamma C_b}B_{v_2}\right)$$

In (A.18) v_r represents the speed by which the robot returns back to the charging region and we pick its value such that it satisfies (A.14).

Remark 17. *We can also use a CBF of the form*

$$h_s = V_o - V_{\min} e^{\frac{k'_e}{v_r}(D_{cs} - \delta)} \tag{A.19}$$

which is a versatile form that can be adapted to different types of battery discharge types (linear or exponential). Since the voltage change in (A.17) *is predominantly linear with a known slope, say k_e, and starting at $V_o(0)$ we can propose an exponential discharge with the form*

$$V'_o = V_o(0)e^{-k'_e t}$$

such that it intersects with the linear discharge of $V_o = V_o(0) - k_l t$ at $t = 0$ and $t = t^*$ such that $V_o(t^*) = V_{\min} + \epsilon$, for $\epsilon > 0$. Equating the two discharge expressions gives an estimate for k'_e

$$k'_e = \frac{1}{t^*}\log\frac{V_o(0)}{V_{\min}+\epsilon} = \frac{k_e}{V_o(0) - (V_{\min}+\epsilon)}\log\frac{V_o(0)}{V_{\min}+\epsilon}$$

this estimate will make $V'_o(t) < V_o(t)$ for $t < t^*$ (thus giving a more conservative or pessimistic value of voltage). The higher the value of ϵ is, the more conservative the estimate of voltage change will be.

Remark 18. *We note that the velocity limit relation in* (A.14) *is an optimistic estimate of the maximum velocity achievable at a given electric current limit (since we considered the simplistic case of the robot moving in a straight line at maximum current, the effect of l is not taken into consideration). Choosing a lower value for v_r in* (A.18) *leads to a reduced chance of violating the constraint on the electric current when the robot is returning back to the charging station.*

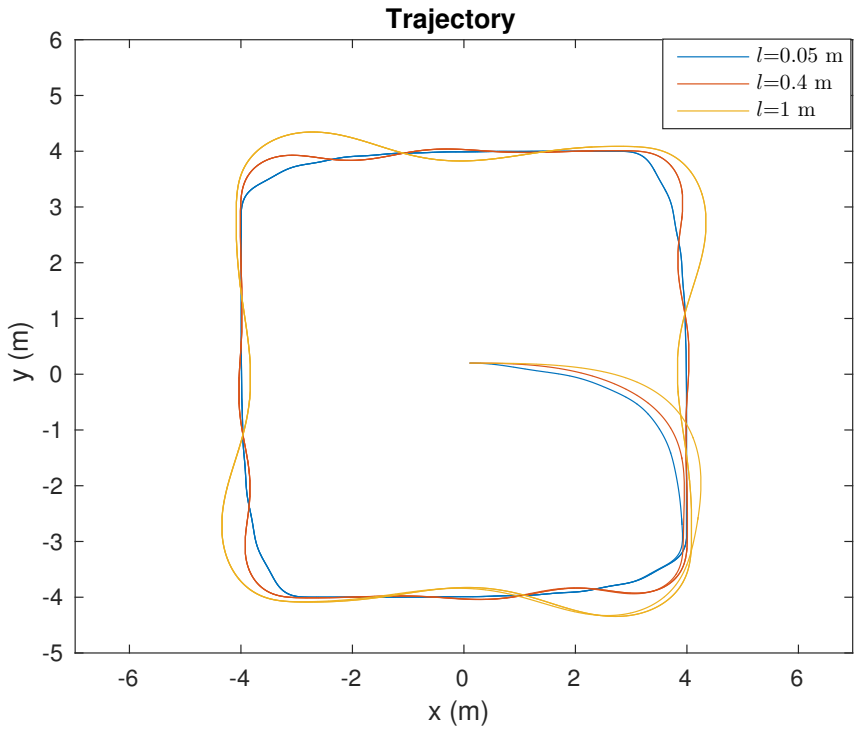
The constraint equation associated with (A.18) is

$$-k_e - \frac{k_e}{v_r}\frac{(x - x_c)^T}{D_{cs}}\mathbf{u} \geq -p_s h_s \tag{A.20}$$

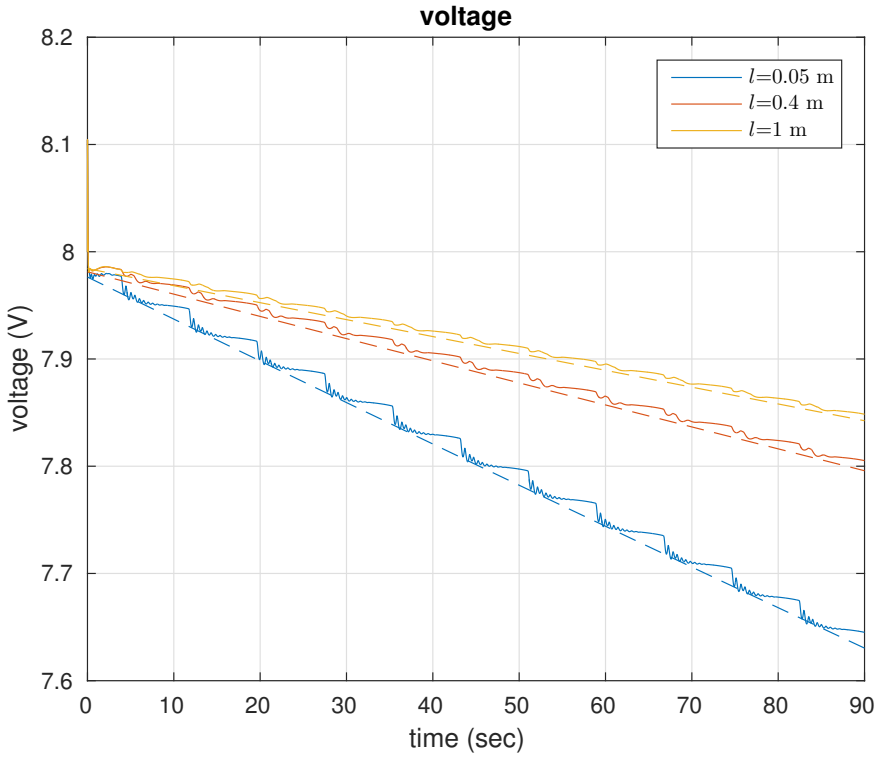
where $p_s > 0$, $x, x_c \in \mathbb{R}^2$ are the robot's position and the position of the charging region center respectively. We can use a quadratic program of the form (6.3) with (A.11) and (A.20) as constraints.

Figure A.2 shows the result of applying the aforementioned quadratic program with the proposed constraints for a task where the robot carries out a perimeter patrol task and then returns to the charging station, considering $V_{min} = 7.6$ volts, $\delta = 0.3$m, $l = 0.1$m, and we present results for two values of return velocity: $v_r = 0.8$m/s and $v_r = 1.5$ m/s. In Figure A.1a the robot is trying to minimally deviate from the task, thus falling back to the station (marked with a red dashed circle) in a spiral trajectory. We note that when the return speed decreases, the electric current requirement on the robot's way back decreases as well, which explains the decrease in the slope of the voltage change from the case of higher return speed as shown in Figure A.2b.

We note that the results presented here serve as a simple example of how to use the full unicycle model to achieve energy sufficiency, but further treatment is in order. One possible direction of development is to consider the full unicycle nonlinear model in the constraint design without the need of simplifying intermediate models (such as single integrator), aiming not to use extra constraints (like constraining the current consumption) thus leading to better utilization of the system. Another direction could be attempting to reduce the unicycle model using feedback linearization to a simpler model, on which the constraints are designed.

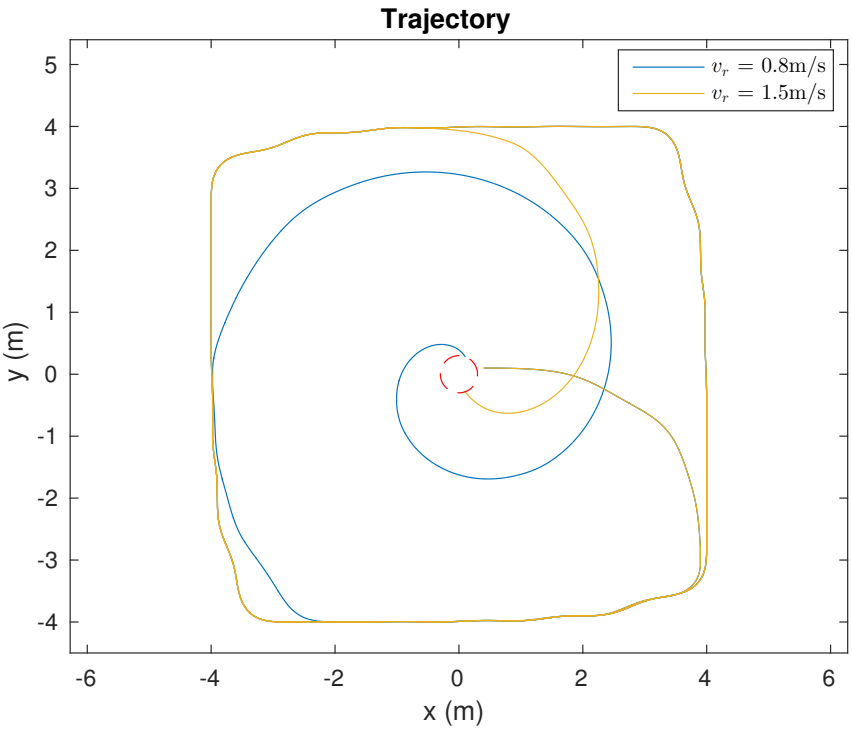


(a) robot's trajectory

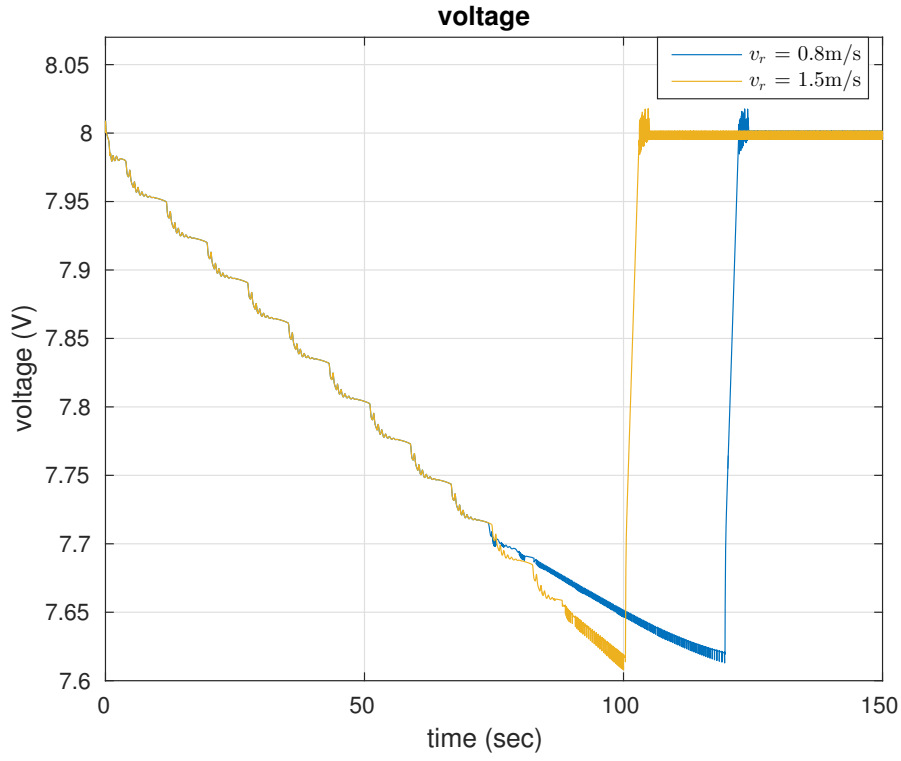


(b) battery voltage

Figure A.1 Trajectory and voltage change of a robot modelled as in (A.5) and battery described by (A.7) for different values of l in (A.2). The dotted lines in Figure A.1b represent lower bounds of voltage change.



(a) robot's trajectory



(b) battery voltage

Figure A.2 Trajectory and voltage change of a robot for two values of return speed: $v_r = 0.8m/s$ and $v_r = 1.5m/s$. The robot moves on the perimeter and falls back to the charging region, denoted by the red dashed circle.