Motion planning for multi-agent dynamical systems in a variable environment

Ngo Quoc Huy Tran

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

Spécialité : **AUTOMATIQUE - PRODUCTIQUE**

Arrêté ministériel : 25 mai 2016

Présentée par

Ngo Quoc Huy TRAN

Thèse dirigée par **Laurent LEFÈVRE,**
co-encadrée par **Ionela PRODAN**

préparée au sein du **Laboratoire de Conception et d'Intégration des Systèmes**
dans **l' École Doctorale Électronique, Électrotechnique, Automatique, Traitement du Signal EEATS (EEATS)**

Planification de mouvement pour les systèmes dynamiques multi-agents dans un environnement variable

Motion planning for multi-agent dynamical systems in a variable environment

Thèse soutenue publiquement le **19 Décembre 2019**,
devant le jury composé de :

Mme. Cristina STOICA MANIU
Professeure, Centrale Supélec, L2S, Paris-Saclay University, Rapporteure
M. Florin STOICAN
Professeur, Université Polytechnique de Bucarest, Rapporteur
M. Esten INGAR GRØTLI
Chargé de Recherche, SINTEF Digital, Mathematics and Cybernetics, Norway, Examinateur
M. Jean-Paul JAMONT
Professeur, Université Grenoble Alpes, Président
Mme. Ionela PRODAN
Maître de Conf., Institut Polytechnique de Grenoble, Co-Encadrante de thèse
M. Laurent LEFÈVRE
Professeur, Institut Polytechnique de Grenoble, Directeur de thèse

Motion planning for multi-agent dynamical systems in a variable environment

Ngo Quoc Huy TRAN

PhD Thesis 2019

Laboratoire de Conception et d'Intégration des Systèmes (LCIS)
Université Grenoble Alpes, LCIS, F-26902, France

I would like to dedicate this thesis to my beloved family, especially, my wife Phuoc Hung and two little princesses Van Ha and Van Anh. I owe you my promises, more than four years of waiting and a sincere apology, just because of my selfish decisions.

Acknowledgements

This research work has been realized for more than three years work in research team “Modélisation, commande et supervision des systèmes complexes ouverts et décentralisés (MACSY-COSY)” of the LCIS laboratory of Grenoble INP, Valence, France, with main funding from the Ministry of Education and Training (MOET - Vietnam). During this important period of my life, I received a lot of enormous help and generous support from many elegant people to make this thesis possible. I would like to take this opportunity to express my appreciation to all of them.

First of all, I wish to thank my supervisors, Mr. Laurent LEFEVRE and Ms. Ionela PRODAN. Nine years ago, in 2012, Laurent gave me a chance to do a Master 2 research course as well as an internship at LCIS, which was a massive challenge for me at that time. Four years later, in 2016, one more time, Laurent bailed me out from MOET - Vietnam and college doctoral to give me another chance for going on a Ph.D. thesis. For his supervision and enthusiastic support, there is no pressure in my studies, and this makes my thesis possible. To my nearly direct supervisor, Ionela, who has been part of all the research in the thesis, I would never forget her kindness when she spent precious times to discuss my works correct my papers, and give various valuable comments for shaping the final version of this manuscript.

To all of them, I’m grateful their high patience and availability for answering my questions and providing help on both aspects of academic side and daily life whenever I needed. It has been an honor to be their Ph.D. student. I have been taught from them, both consciously and unconsciously, how to organize my works, how to approach and solve scientific problems. They are extremely valuable experiences for my future research work.

I would like to thank the two reviewers Prof. Cristina STOICA MANIU and Prof. Florin STOICAN, for their insightful remarks and encouragement which helped in improving the quality of the manuscript. They spent their valuable time and effort to evaluate my thesis. I would also like to thank the other members of my oral defense committee, specially Prof. Jean-Paul JAMONT, Dr. Esten Ingar GRØTLI, for their time, interest, and helpful questions.

I am thankful to Mr. Esten Ingar Grøtli, who has been part of almost all the research in the thesis and co-authored most of the papers. He has provided me the information such as the map, data, figures, and control objectives of the surface vehicles in Trondheim fjord, Norway. Thanks to him, I received exciting ideas of randomized sampling algorithms as well as the critical comments for the important papers through my thesis.

During the three years, I also had the chance to discuss with many people on different research topics. I thank to Medianu SILVIU, former PhD student at LCIS, for inspirational discussions with me regarding the knowledge in the field of advanced mathematics. I thank also to Mr. El Mehdi KHALFI, former PhD student at LCIS, for the fruitful discussions on the multi-agent systems.

Furthermore, I am grateful to the LCIS assistant team who helped me easily integrate to the research life of the laboratory. Firstly, Ms. Jennyfer DUBERVILLE, Ms. Caroline PALISSE and Carole SEYVET, the LCIS secretaries, were always available to get through the administrative issues, this giving me more time to focus on the research works. I also take this opportunity to thank everyone in the computing service, Mr. Cedric CARLOTTI and Mr. Karim Oumahma, for their important technical support. I am thankful to Mr. André LAGREZE, associate professor at LCIS, for organizing sportive activities which got me out of the office and integrate into new life in France.

In addition, I enjoyed the friendship with other members in our group: Lai, Hung, Thinh, Yoann, Mehdi, Igyso, Youness, Daniel. We have shared lots of exciting moments together. A huge thanks and big hugs to them. I also acknowledge the presence and help of the Vietnamese friends, Mr. Du and Mrs. Thao, who always let me stay in their house when I arrived in Paris for the business or trip.

Last but not least, I convey special acknowledgement to my patient wife, Phuoc-Hung, my parents, as well as the rest of my family for their love and support. I thank them for always believing in me and in all my decisions.

Ngo Quoc Huy TRAN

ÉCOLE DOCTORALE EEATS

Abstract

SPECIALITE: AUTOMATIQUE - PRODUCTIQUE

Doctorate degree

by Ngo-Quoc-Huy TRAN

This thesis proposes optimization-based control solutions for the motion planning of multi-agent dynamical systems operating in a variable environment (with static/mobile obstacles and time-varying environmental disturbances).

Collision-free paths are planned for the agents through the combined use of set theory (particularly, bounded convex sets), non(-linear) Model Predictive Control (MPC), Potential Field (PF) and graph-based methods. The contributions build on the proposal of repulsive potential field constructions together with on-off barrier functions which describe and, respectively, activate/deactivate the collision-free conditions introduced in a distributed NMPC framework. These constructions are further used for connectivity maintenance conditions among the group of agents while ensuring the tracking of the a priori generated path. Furthermore, a nonlinear disturbance observer is integrated within the control scheme for environmental disturbance rejection.

Finally, the results are validated in simulation through comparisons with mixed-integer approaches and over a benchmark for the safe navigation of Unmanned Surface Vehicles (USVs) in the Trondheim fjord, Norway, using real numerical data.

Notations

Parameter and variable fonts

Element	Font
Scalar variable	regular letter
Vector	regular and bold letter
Weighting matrix	capital and bold letter
Matrix	capital letter
Set	capital and calligraphic letter
Union of set	capital and boldface letter

Operator

Notation	Description
Δ	relative position or velocity between two agents
$\gamma(x)$	sum function
$\nabla\gamma$	gradient of sum function $\gamma(x)$
$\dot{x}(t)$	time derivative of vector $x(t)$
Conv$\{\mathcal{C}\}$	convex hull of the set $\mathcal{C}$
$\partial\mathcal{C}$	boundary of the set $\mathcal{C}$
$Int(\mathcal{C})$	strict interior of the set $\mathcal{C}$
$\mathbf{A}^\top$	transpose matrix of $\mathbf{A}$
$diag\{a, b, c\}$	diagonal matrix with the diagonal elements a, b, c
$\lvert\mathbf{x}\rvert$	Absolute value of the vector x
$\lVert\mathbf{x}\rVert$	Euclidean norm
$\lVert\mathbf{x}\rVert_{\mathbf{Q}}$	Q-norm of vector x
$\lVert\Delta p^s_{i,\ell}\rVert$	Euclidean distance calculated from Chebyshev center, the largest inscribed ball of ℓ^{th} fixed obstacle and current position of i^{th} agent
$\lVert\Delta p^d_{i,j}\rVert$	Euclidean distance between the current position of i^{th} and j^{th} agents

Subscript and superscript

Notation	Description
$(.)_i$	variable of agent indexed by $i \in \mathcal{V}$
$(.)_0$	initial state of the variable
$(.)_\ell$	repulsive potential of fixed obstacle in general
$(.)_\ell^f$	repulsive potential of ℓ^{th} fixed obstacle constructed by using fractional function related to sum function
$(.)_\ell^e$	repulsive potential of ℓ^{th} fixed obstacle constructed by using exponentional function related to sum function
$\bar{(.)}_i$	predicted states or inputs of i^{th} agent
$(.)_p$	current way-point
$\hat{(.)}$	estimation of (.)

Variable

Notation	Description
z	Auxiliary binary variable, $\{0, 1\}$
$\mathbf{I}_n$	identity matrix of size $n \times n$
$\mathbf{0}_n$	zero matrix of size $n \times n$
$\mathcal{F}$	workspace of agents
J_i	cost function
$\mathcal{L}_i$	running or stage cost
E_i	terminal cost
$\mathcal{N}_i$	neighbors of i^{th} agent
$\mathcal{G}$	undirected graph
$\mathcal{V}$	set of agents or set of nodes representing all agents
$\mathcal{E}$	set of edges of the graph
$\mathbf{x}_i$	state variables of i^{th} agent
$\mathcal{X}$	state space
$\mathcal{B}_\ell$	largest inscribed ball of ℓ^{th} polytopic region
ρ_ℓ	radius of the ball $\mathcal{B}_\ell$
$\mathbb{R}$	set of real numbers
$\mathbb{N}$	set of natural numbers
$\mathbb{O}$	union of forbidden polytopic convex regions
$\mathbb{S}$	union of the set of the static repulsive potential functions
$\mathbb{M}$	union of the set of the dynamic repulsive potential functions
$\bar{\mathbf{x}}_i(\tau)$, $\bar{\mathbf{u}}_i(\tau)$	predicted states and inputs of i^{th} agent
Λ	lookahead distance
u_i	surge velocity of i^{th} ship
v_i	sway velocity of i^{th} ship
r_i	yaw rate of i^{th} ship
ψ_i	heading or yaw angle of i^{th} ship
w_i	disturbance of i^{th} ship
$\Gamma_i(p_i, r_i^a)$	view range of i^{th} agent corresponding to its current position p_i within a radius r_i^a
$\Gamma_i^c(p_i, r_i^a)$	communication range of i^{th} agent corresponding to its current position p_i within a radius r_i^c
t	time instant
N_{obs}	number of static obstacles

Variable

Notation	Description
N_a	number of agents
V_i	Lyapunov function
T_p	prediction horizon
T_e	sampling time
$\mathbf{Q}_i, \mathbf{R}_i$	weight matrices of the state and input variables of stage cost
$\mathbf{P}_i$	weight matrices of the state variables of terminal cost
N_p	number of subintervals

Nomenclatures

ACO	Ant Colony Optimization
AIS	Automatic Identification System
APF	Artificial Potential Field
AUV	Autonomous Underwater Vehicle
CBF	Control Barrier Function
COLREGS	Convention on the International Regulations for Preventing Collisions at Sea
CSII	Cybership II
GPS	Global Positioning System
LOS	Line-of-Sight
LOA	Maximum length of a ship's hull
MAS	Multi Agent Systems
NDO	Nonlinear Disturbance Observer
MINLP	Mixed-Integer Nonlinear Programming
MIP	Mixed-Integer Programming
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
RP	Repulsive Potential
OORP	On-Off Repulsive Potential
D-OORP	Distributed On-Off Repulsive Potential
PSO	Particle Swarm Optimization
OCP	Optimal Control Problem
PRM	Probabilistic Road Map
UAV	Unmanned Aerial Vehicle
USV	Unmanned Surface Vehicle
SA	Simulated Annealing
RRT	Rapidly-Exploring Random Trees
RRT*	optimal Rapidly-exploring Random Trees

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motion planning for multi-agent dynamical systems

Over the past few decades, the paradigm *Multi-Agent Systems* (MAS) used firstly in the domain of artificial intelligence, a branch of computer science, has had many outstanding developments and spread in a variety of fields, particularly, the control engineering sector. Due to their flexibility in structure, MAS are generally exploited in two ways [Wooldridge, 2009]: as an approach for modeling various systems, and as an approach for developing flexible and extensible software/hardware systems [Wang et al., 2015, Mfumu et al., 2018]. In control engineering, MAS is defined by a group of agents having a particular dynamical model and operating in an environment in the presence of constraints. Consequently, MAS in control engineering is a brief way of denoting multi-agent dynamical systems. In the control of MAS, the control decisions are derived by taking into account the system constraints and common objectives which are then implemented in real time. Feedback is considered as an effective method to deal with the uncertainties affecting the system and its environment.

The flourish of control engineering applications has played an essential role in the success of MAS for various tasks, such as improving data communication, control and coordination, task allocation, processing, and increasing the use of sustainable energy. With such a crucial position, MAS are becoming a hot trend in autonomous control systems. Some noteworthy projects are enumerated highlighting their importance in the modern world: deployment of multiple watercrafts operating without a crew, known as Unmanned Surface Vehicles (USVs), for specific tasks that are tiresome or dangerous for humans. For example, the Protector USV of Rafael Advanced Defense Systems Ltd [Rafael, 2004] is capable of performing tasks such as marine surveillance, naval warfare and force protection or the ECA robotics USVs designed for water and oceanographic survey. Not quite yet out of the laboratory, self-propelled cars, show great promise for the future of transport and infrastructure [Paolillo et al., 2018, Lim et al., 2018, Guo et al., 2019].

A group of agents can reach a common goal through coordination among them through the so called cooperative control of multi-agent systems (even if the goal is unattainable by a single agent or by multiple agents working separately). Without a doubt, this means exchanging information through an ad hoc communication network (each agent has a neighborhood of agents with which it communicates). The information may represent attitude, position, velocity, voltage, temperature, pressure, and so on, depending on the different applications. Plenty of remarkable

examples of cooperative control can be enumerated such as water distribution networks [Fontana et al., 2017], output power control of microgrids [Morstyn et al., 2015, Mortezaei et al., 2018], coverage control for a mobile sensor network [Song and Fan, 2018, Zhong and Cassandras, 2008], localization, formation control and tracking for multi-ground/surface/aerial vehicles [Pack et al., 2009, Maghenem et al., 2018, He et al., 2019a, Wang et al., 2019b]. As a result, we can summarize some essential features of these applications in order to have a general view on the use of MAS in a control engineering framework:

- the dynamical model of each agent can be linear or nonlinear, time-variant or time-invariant, etc;
- the information exchange between agents with their neighbors can be weak or strong, directed or undirected, etc;
- diverse working space (ground, air, underwater, in the presence of static or dynamic obstacles, and the like);
- to be relevant to the real world, constraints (saturation of actuators, energy consumption, collision avoidance) are mandatory.

Moreover, each agent has to own the following primary properties for ensuring an excellent performance of MAS:

- autonomy: make decisions to act as well as react by itself in the absence of human intervention in case of critical situations;
- cooperation: exchange information among agents in the group to coordinate their actions for accomplishing the common goals of MAS;
- proactiveness: the agents' actions need to be controlled actively rather than responding passively; this might come from a predictive mechanism for each agent.

This thesis mainly takes into account three critical issues in the control of multi-agent dynamical systems: path/trajectory planning of multi-agent systems in the presence of constraints; collision and obstacle avoidance and distributed control for information exchange and connectivity preservation under disturbances:

- **Path/trajectory planning** means that it is necessary to generate a collision-free feasible path/trajectory in order to steer an autonomous agent from an initial to a target position in its given working space.
- **Constraints handling** which aims at formulating the constraints in the control design.
- **Distributed control** for maintaining the connectivity of the network, handling the information exchange.
- **Robustness** ensures the operation of MAS notwithstanding disturbances or perturbations.

Next, a brief literature review on the above mentioned issues is presented. Various sources of inspiration in this effort come from [Chen and Wang, 2005], [Prodan, 2012], [Guerrero and Lozano, 2012], [Nguyen, 2016].

1.1.1 Path/trajectory planning

Autonomous multi-agents need to operate without human intervention. In order to complete their tasks, the agents have to be proactive in favor of deciding their own actions. Therefore, the risks that may affect the agents need to be minimized through optimal planning, depending on their tasks. Hence, the primary purpose of MAS motion planning is to generate collision-free paths or trajectories which minimize a cost function combining time, energy, or distance penalties for all agents in the group reaching their desired targets. In this framework, the collision avoidance between MAS and (static and dynamic) obstacles as well as among the agents in a group have to be firmly guaranteed while tracking the generated paths/trajectories and converging towards the desired destinations in real-time conditions. A number of different popular approaches found in the literature will provide an overview of the path/trajectory planning for autonomous MAS.

Evolutionary algorithms are employed to solve the path/trajectory planning problem quickly without necessarily guaranteeing a global solution. There are many variants of these algorithms, such as *Ant Colony Optimization* (ACO) [Dorigo and Birattari, 2010], *Simulated Annealing* (SA) [Aarts et al., 2005], etc. *Genetic algorithms* (GA) [Hu and Yang, 2004, Tuncer and Yildirim, 2012] and *Particle Swarm Optimization* (PSO) [Chen and Ye, 2012, Xue et al., 2013]. For example, the key idea at GA is to employ a grid-based model representing the workspace of MAS [Sugihara and Smith, 1997, Karami and Hasanzadeh, 2015]. This grid can be described in two ways, either as an orderly numbered grid or in a coordinates plane. Accordingly, the calculation of the distance between the agents and the boundary of obstacles can be determined easily by their actual boundary plus the minimum safety distance when considering the size of the agent. Furthermore, the potential path is represented by variable-length chromosomes and the agents go from the initial to the target point through intermediate points. This is a procedure based on stochastic search techniques analogous to natural evolution based on the principle of survival of the fittest [Davis, 1987]. In contrast to GA, PSO, which was initially employed by [Eberhart and Kennedy, 1995], is a stochastic evolutionary algorithm which does not incorporate survival of the fittest. It is rather inspired by the group's ability of some animal species to locate a desirable position in the given area, particularly, birds flocking and fish schooling. Each particle in PSO represents a potential solution to an optimization problem and is associated with a velocity which is dynamically adjusted based on the particle and their neighbors searching experience. As a result, each particle is attracted to a stochastically weighted average of its personal best position and the global best position of the swarm. The main advantages of PSO are their ease of implementation and a more rapidly convergence than GA, a comparison between them can be found in [Roberge et al., 2013].

Graph-based approaches, are also called randomized sampling algorithms and have achieved tremendous success in solving challenging motion planning problems of autonomous vehicles operating in complex environment. They are now very well-known thanks to the two famous methods: *Probabilistic Road Map* (PRM) and *Rapidly-Exploring Random Trees* (RRT). Firstly, the PRM as proposed in [Kavraki et al., 1994], can be employed to find a feasible path for the agent considering the start and target points within a grid map (described by a set of randomly non-collision nodes, also called milestones) constructed using a multiple-query technique. After each query, the shortest path without conflicts for static situations is obtained by using the Dijkstra algorithm. A set of the shortest paths creates a feasible path for the autonomous agent. Plenty of applications of PRM for different systems like *Unmanned Aerial Vehicle* (UAV), multi-robots, can be referred in [Hsu et al., 1998], [Clark, 2005], [Baek et al., 2018]. The PRM algorithm, firstly introduced by [LaValle, 1998], requires thousands of connecting points sampled

randomly from the state space. The fundamental idea is to initialize a tree rooted at an initial state. This tree is explored in the working space with the randomly sampled collision-free states by applying control inputs over short time intervals to reach new states. These obtained states are taken into account as vertices of the tree, and each directed edge is considered as an input applied for reaching another new state. If a new state reaches the desired region, an open-loop path from the initial state is represented by the tree. RRT has shown to be effective for UAV applications [Saunders et al., 2005], [Kothari and Postlethwaite, 2013], in the robotic domain [Krejsa and Věchet, 2005], [Moon and Chung, 2015], for *Autonomous Underwater Vehicle* (AUV) motion planning [Tan et al., 2005] and *Unmanned surface Vehicle* (USV) [Dalpe and Thein, 2017] path planning in dynamic flow-fields. There exists a number of variants such as Anytime RRTs [Ferguson and Stentz, 2006] using similar heuristics for selection and expansion of nodes, RRT* [Webb and Van Den Berg, 2013] which guarantees finding asymptotically optimal paths/trajectories in state spaces of arbitrary dimension.

Artificial Potential Field (APF) for global path planning was originally introduced by [Khatib, 1986]. It is based on a scalar function which combines a repulsive and attractive function representing the obstacles and the final destinations, respectively. The key element of this approach is the potential function which has a high value in the obstacles to prevent the agents from getting very close to them, and its value decreases towards a minimum at the goal configuration. In other words, the gradient of the potential function provides in each point the moving direction of the agents and makes it progress according to the decreasing gradient of the potential function, thus, generating safe paths. The potential function in the context of motion planning can be built by either considering the distance to the obstacles (see [Leonard and Fiorelli, 2001, Stastny et al., 2015, Woods and La, 2017], [Baillard et al., 2018]) or the specific shape of the obstacles (see [Prodan et al., 2013], [Cetin and Yilmaz, 2016], [Rasekhipour et al., 2017]). A detailed review of the potential field approach is proposed in [Koditschek, 1992].

Optimization-based control methods are widely applied for a variety of fields from simple to complex, particularly, control engineering. What makes this method stand out from other methods is the fact that it allows for the specification of an objective function which is optimized by the controller. From this, the optimal solution is found subject to the considered criterions. Due to the complexity of most applications, these approaches are most often solved numerically. Among the optimization-based approaches, MPC [Garcia et al., 1989], [Rawlings and Mayne, 2009], has a tremendous success for constraints handling in the motion planning problem due to the use of a receding finite horizon technique [Mayne et al., 2000]. This has made MPC more applicable in practice, [Qin and Badgwell, 2003] with relatively long sampling periods (measured in minutes and hours) at the beginning. However, in the recent years the development of semiconductor technology increased processing speed and memory, and the improvement of algorithms [Bock et al., 2007], [Diehl et al., 2009] have enabled the MPC to be applied for many branches of control engineering such as automotive control applications [Cairano et al., 2007, Di Cairano et al., 2008, Di Cairano et al., 2010], smart grids [Parisio et al., 2014, Pham et al., 2017] water control for irrigation canals [Aguilar et al., 2016], sewer networks [Pedersen et al., 2017], motion planning for ground/aerial vehicles [Nayl et al., 2015, Nägeli et al., 2017], area coverage problem [Nguyen and Maniu, 2016, Tzes et al., 2018] etc. Other advantageous MPC characteristics are the capability to cope with uncertainty [Scokaert and Mayne, 1998], [Mayne et al., 2005] or time-delay [Liu et al., 2006]. The detailed design of MPC can be read in some famous books [Kwon and Han, 2006], [Maciejowski, 2002], [Allgöwer and Zheng, 2012].

However, with such an approach, the collision-free optimal paths/trajectories for MAS is able only to be achieved if collision and obstacle avoidance is taken into account. Note that we need to clearly distinguish the meaning of the two expressions (i) *obstacle avoidance* (i.e., collision

avoidance with stationary obstacles), and (ii) *collision avoidance* (i.e., collision avoidance with dynamic obstacles that have predefined paths, and mutual collision avoidance among multiple agents).

1.1.2 Collision and obstacle avoidance constraints

MAS operate in a workspace in the presence of many static and dynamic obstacles, which are usually considered as convex regions. In other words, the feasible region for the operation of MAS is the complement of the convex regions, i.e., a non-convex region. Therefore, the non-convex constraints which represent a feasible region describing collision and obstacle avoidance for MAS need to be first taken into account in the design strategy. Usually, these issues are addressed as *explicit (i.e., hard) or implicit (i.e., soft) constraints*. On the one hand, *explicit constraints*, concerning constrained optimization problems with the simple cost function subjecting to the (non-convex) constraints appearing explicitly. However, this problem becomes complex if there are too many (non-convex) constraints. Typical for this approach is *Mixed-Integer Programming*. On the other hand, *implicit constraints*, usually, are formulated through *Artificial Potential Field* approaches, where the constraints are penalties in the objective function. This means that the objective function becomes complicated, whereas the constraints are straightforward.

Optimization control based using *Mixed-Integer Programming* (MIP) is a general framework which describes problems with both discrete and continuous decisions variables. The algorithm used for solving MIP problems is branch-and-bound [Linderoth and Savelsbergh, 1999]. The fundamental idea of the algorithm is to relax the non-convex constraints, maintain some simple bounds and then to branch. This algorithm has outstanding benefits when terminated early and a globally optimal solution for non-convex problems is achieved. Since the early 2000s, MIP started to be used for solving motion planning problems with collision avoidance constraints. For example, [Deits and Tedrake, 2015] and [Molinari et al., 2017] tackled the optimization of trajectories for autonomous agents in the presence of static and dynamic obstacles. The authors of [Prodan et al., 2011], [Radmanesh and Kumar, 2016] focused their works on coordinating the efficient interaction of MAS in scenarios like task assignment or trajectory planning; [Bonthu et al., 2019] designed MPC based on MIP approach to develop an optimal power management strategy for saving the cost of electricity for buildings. However, despite its unique capabilities and the availability of solvers, MIP has some shortcomings. First of all, in some cases, if the solution to a relaxed subproblem is not integral, but has a cost worse than the best MIP solution found already, that branch can be terminated as stated in [Richards and How, 2005]. Secondly, MIP is NP-hard, i.e., the computational complexity increases exponentially with the number of binary variables used in the problem formulation [Garey and Johnson, 2002]. This limits their use for real time applications. A geometrical view of MIP for non-convex, non-connected regions representation is presented in [Prodan et al., 2015]. In here, the authors also address the reduction of the computational requirements of MIP while providing examples for motion planning problems with collision avoidance constraints [Ioan et al., 2019a, Ioan et al., 2019b]. To the best of the author's knowledge, the sudden termination of the branching has not been yet much addressed.

Methods based on APF generate a Potential Field (PF) based on potential functions of static and/or dynamic obstacles or desired destinations in the operating space. The path is planned by moving in the descent direction of the field. The main advantage of this approach over the MIP approach is its low calculation cost even with complex PFs for obstacles. There are also many experimental results for motion planning which exploit the use of APF-based MPC, and

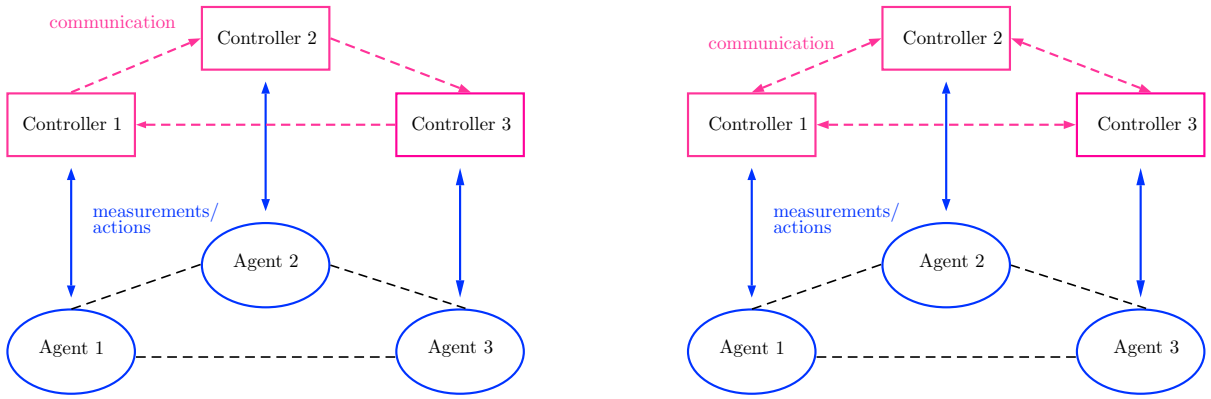
can be listed: autonomous road vehicles [Rasekhipour et al., 2017, Wang et al., 2019a], motion planning and tracking of an omnidirectional autonomous robot/two wheeled vehicle as presented in [Teatro et al., 2014, Shibata et al., 2018]. Thus, one disadvantage of APF approach compared to the MIP technique in motion planning problems is the fact that a badly chosen potential function may cause the solution to get into local minima [Goerzen et al., 2010]. This will be further discussed in the next chapters. Thus, a combination of MPC and APF, while not treated widely in the literature, seems to be worthwhile.

For all the above arguments, MPC using APF-based *soft constraints* (i.e., implicit constraints) will be considered as a fundamental approach for path planning of MAS in the current work and will be further detailed throughout this thesis.

1.1.3 Distributed control

Acknowledging the limitations which appear in a MAS (e.g., communication limits and computational resources) makes distributed control an attractive approach. In this endeavor, *distributed control* emerges as an efficient approach for controlling MAS since the computation load is reduced significantly while still ensuring the interaction among the agents when compared with *centralized control* approach. Furthermore, the cooperation among the agents of *decentralized control* is weak due to the fact that each agent's controller is only based on its own state information uniquely.

Similar to the decentralized control approach, each agent also has its own controller. However, information is transmitted among the local controllers, so each one of them has some knowledge on the behavior of the others and the local control action is influenced by this information. In other words, each controller receives the decision variables from the nearby subsystems, the so-called neighbors. It can be thus aware of its neighbors' behavior to make the right control decisions for the agent that is being controlled [Scattolini, 2009], [Cao et al., 2013]. In addition, a variant of distributed control is the *hierarchical control* in which MAS is partitioned into different layers of abstraction. The highest layer takes planning decisions regarding the MAS, whereas lower layers may model more detailed features (such as path tracking or sense and avoidance). Information is typically exchanged from the upper to the lower layers [Leitão, 2009, Hou et al., 2018].



(a) Sequential architecture.

(b) Parallel architecture.

Figure 1.1.1: Distributed control approach.

Recent advances in computing, wireless communications, or vehicular technologies are making it possible to deploy multiple ground/aerial/marine vehicles that can operate autonomously as well as cooperate with each other to achieve a global objective. The distributed scheme needs only a simple sparse communication network and fewer data to process as well as the flexibility to adapt to variations of operation conditions automatically. By considering that assumptions such as communication and information exchange among agents are ensured, the states (i.e., the variables containing information related to the agents' behaviors) are available to all the distributed controllers. There are two architectures usually used to implement the distributed algorithm for MAS: the *sequence* and *parallel* as illustrated in Fig. 1.1.1a and Fig. 1.1.1b, respectively. On the one hand, the distributed controllers are evaluated in *sequence*, this means that once at each sampling time and one-directional communication between consecutive distributed controllers is employed, i.e., controller $i+1$, for example, is only evaluated after controller i has been evaluated. On the other hand, the distributed controllers evaluated in parallel, i.e., at the same sampling time. The controllers receive and evaluate the information of each others once (i.e., *non-iterative*) or iterate (i.e., *iterative*) to achieve a solution at a sampling instance [Liu et al., 2010, Christofides et al., 2013].

The directed or undirected graph is employed for the information exchange (i.e., interaction) among agents [Godsil and Royle, 2013, Deo, 2017]. A graph, in general, comprises finite nonempty set nodes (or vertices) and edges set of nodes. Each node stands for an agent, which can have information exchange with all or several agents. A directed graph is denoted by a sequence of ordered edges where information is transmitted between adjacent agents in a single direction, but not vice-versa. Here, i.e., a node is considered the parent node while the next node is the child node, and the child node can obtain the information from the parent node. Meanwhile, an undirected graph is a sequence of unordered edges, .i.e., adjacent agents can receive the information from each other. Under this framework, it is easy to recognize that the coordination problems under a general undirected graph are more facile than those under a directed graph [Mei et al., 2015].

Some primary criterions like control, synchronization, robustness and optimization problem are presented in Table 1.1.1 to show the differences between centralized and distributed control [Negenborn et al., 2010].

Table 1.1.1: Centralized versus distributed control.

	Centralized	**Distributed**
Control	Central controller supervises all agents	Each agent is supervised by its controller
Synchronization	Agents transmit data to a central controller	Agents exchange data among each other
Robustness	Low	High
Optimization problem	Central controller performs overall optimization problem	Overall optimization problem is partitioned into subsystems

Concerning the applications in a distributed framework, the problems of event-triggered consensus control in an MAS are studied in [Ge and Han, 2017, Ding et al., 2017, Wu et al., 2018], distributed control and management strategies for the power system in the context of microgrids is provided by [Han et al., 2017, Morstyn et al., 2018], distributed formation control for multi-

ple tethered space net robots using APF and neural network [Liu et al., 2018b], a distributed control strategy presented by [Santos and Egerstedt, 2018] for a multi-robot team with heterogeneous sensing capabilities using limited communication where each robot has knowledge about the sensor capabilities of its neighbors. In recent years, distributed MPC has gained a lot of interest due to its many benefits [Camponogara et al., 2002]. The authors of [Venkat et al., 2005] has unified existing concepts in linear MPC with guaranteed nominal stability and performance properties. A design of multi-vehicle formation stabilization is detailed in [Dunbar and Murray, 2006], where a distributed optimal control is generated for each subsystem. Many studies on path planning for MAS have been investigated, for example, [Mo and Lin, 2018, Yang et al., 2018] by using non-convex control input constraints to achieve the cooperation task for second-order MAS, robotics [Mendes Filho et al., 2017, Luis and Schoellig, 2019], many vessels [Wang et al., 2017b, Chen et al., 2018a]. The combination of MIP technique and distributed MPC scheme has also attracted interest through the literature: cooperative distributed robust trajectory optimization for a set of homogeneous UAVs [Kuwata and How, 2010], feasible path generation for MAS in the presence of obstacles [Prodan et al., 2014], the cooperation of USVs at waterway intersections [Chen et al., 2018b], or reactive power dispatch [Murray et al., 2018]. Another control strategy is to combine the APF approach with MPC as in [Huang et al., 2018] which presented the cooperative driving systems accomplishing both path planning and motion control synchronously for automated vehicle platoon; [Ren et al., 2018] introduced hierarchical adaptive cruise control system to adjust the vehicle speed to keep the appropriate distance from the one in front; or [Katriniok et al., 2019] implemented distributed motion planning in road intersections.

The obstacle and collision avoidance problem as mentioned above, represented by non-convex constraints in the distributed optimization-based control scheme, will be implemented by appropriate solvers. To deal with MIP technique in MPC scheme, CPLEX software package [IBM, 1999] is used in combination with the MATLAB toolbox YALMIP [Löfberg, 2004]. For solving the problem of APF in the MPC framework, there are many efficient solvers which apply various optimization techniques to speed up the resolution of the problem: ACADO toolkit [Houska et al., 2011] implements a sequential quadratic programming (SQP) framework and solution of a convex subproblem - a QP; FORCES NLP [Zanelli et al., 2017] uses interior-point method which approximates the Hessian and has adaptive barrier rules; interior-point solver IPOPT runs [Wächter and Biegler, 2006] in the Casadi framework [Andersson et al., 2019], which is an open-source software tool for numerical nonlinear optimization problems in general and optimal control (i.e., optimization involving differential equations) in particular; PANOC [Stella et al., 2017] uses the proximal averaged Newton-type method for optimal control to solve obstacle avoidance problems.

1.2 Contributions of the thesis

The above literature review highlights that there are still challenging issues in motion planning for MAS operating in variable environments (with static/mobile obstacles and other environmental disturbances). These are tackled using various methods which come with both advantages and shortcomings from the viewpoints of implementation and objectives validation.

The work presented in this manuscript concentrates on providing effective potential field constructions that are easy to employ in a distributed, optimization-based control problem. The theoretical results are tested over a benchmark for the safe navigation of surface vehicles at sea. More specifically, we started developing the results in [Prodan et al., 2013] where repulsive

potential fields were integrated in an MPC problem for trajectory tracking of a formation of MAS. Here we go further and construct efficient *on-off barrier functions* that will be useful for weighting the APF components. This proposed method is used to deal with local minima, which is usually a drawback of the APF method. Moreover, we also provide comparisons with the classical MIP approach to show the effectiveness of the proposed method. In particular, we are focusing on generating the feasible path for MAS by using an optimization-based control scheme with polytopic constraints. More precisely, we first construct the repulsive potential fields for the fixed and moving obstacles/MAS based on their description as bounded convex regions (i.e., bounded polyhedra). Next, the *on-off barrier functions* derived from the *Logistic Regression* (often used in Machine Learning algorithms for binary classification) [Hosmer and Lemesbow, 1980] will be taken into account based on the safety distances among the agents. Such a combination (between on-off barrier function and APF) can be seen as a main contribution in the thesis as it dramatically reduces the complexity and augments the performance of the problem of motion planning. As a result, it will also allow reducing the computational cost and the efficient handling of an increased number of constraints. Furthermore, graph-based methods, specifically, rapidly-exploring random trees, are combined with tracking MPC, which will be useful for achieving an improved result for path generation of MAS.

It is worth noting that all proposals, as mentioned above, come from difficulties that may happen in practice. Hence, the contributions are briefly summarized in the following:

- In [Tran et al., 2017], we present a potential field-based control approach for multi-agent linear dynamical systems in a multi-obstacle environment. The potential field constructions are used for penalizing in the cost for the obstacle avoidance conditions. Furthermore, a barrier function is designed for the control inputs to prevent the imminent collision among the agents. Comparisons through simulations with mixed-integer formulations are provided.
- Next, in [Tran et al., 2018], we introduce an NMPC-based algorithm for safe navigation of multi-agent dynamical systems in a variable environment with fixed and moving obstacles. The contribution lies in the construction of repulsive potential fields for the fixed and moving obstacles which are introduced in a predictive control optimization problem to penalize collision. The constraints are activated only in the view range of the agent using on-off barrier functions. Collision-free motion planning of ships in Trondheim fjord harbor serves as a benchmark for testing our algorithm. Simulations and comparisons with a classical mixed-integer framework prove the efficiency of the algorithm and its feasibility to run in real-time.
- Finally, [Tran et al., 2019] presents a distributed optimization-based control algorithm for the safe navigation of multi-surface vehicles in a complex environment with fixed and moving obstacles. We first generate off-line within the a priori known environment a collision-free path using RRT* (optimal Rapidly-exploring Random Trees) algorithm. Next, PF constructions are developed through the use of on-off barrier functions to ensure vehicle connectivity and fixed/moving obstacles avoidance. These ingredients are integrated in a distributed NMPC framework which activates the constraints only in the view range of the agents. The algorithm shows good results in simulations for both connectivity maintenance and validation of the COLREGS rules.

We provide here the complete list of accepted/submitted publications:

1. **Ngo Quoc Huy Tran**, Ionela Prodan, Esten Ingar Grøtli, Laurent Lefèvre: Safe navigation in a coastal environment of surface vehicles under uncertainties: a combined use of potential field constructions and NMPC, *Ocean Engineering*, submitted in August 2019.
2. **Ngo Quoc Huy Tran**, Ionela Prodan, Esten Ingar Grøtli, Laurent Lefèvre: Distributed nonlinear optimization-based control formulti-agent systems navigation in a costal environment, *In Proceedings of the 19th IEEE European Control Conference, Naples, Italy, 2019, pp.3377-3382.*
3. **Ngo Quoc Huy Tran**, Ionela Prodan, Esten Ingar Grøtli, Laurent Lefèvre: Potential-field constructions in an MPC framework: application for safe navigation in a variable coastal environment, *In Proceedings of the 6th IFAC Conference on Nonlinear Model Predictive Control, Madison, Wisconsin, USA, 2018, pp.307-312.*
4. **Ngo Quoc Huy Tran**, Ionela Prodan, Laurent Lefèvre: Nonlinear optimization for multi-agent motion planning in a multi-obstacle environment, *In Proceedings of the 21st IEEE International Conference on System Theory, Control and Computing, Sinaia, Romania, 2017, pp.488-493.*

1.3 Organization of the manuscript

This thesis includes 5 chapters, including the introduction (see also Fig. 1.3.1).

- **Chapter 2** introduces the motion planning optimization problem formulation and details two methods for handling collision avoidance constraints: MIP (Mixed-Integer Programming) and APF (Artificial Potential Field). This requires first the description of the obstacles through set-theoretic tools. Next, some classical repulsive potential filed constructions and barrier function are presented. Comparisons between the APF method (i.e., implicit constraint) and MIP technique (i.e., explicit constraint) are carried out in order to assess the advantages and shortcomings of each method.
- **Chapter 3** proposes a collision avoidance algorithm via a coherent combination between potential field-based constructions and NMPC. In particular, we first construct on-off barrier functions which activate the associated repulsive potential for fixed and moving obstacles. Then, we develop an NMPC-based algorithm which activates the constraints in the view range of the agents. Finally, the proposed algorithm is validated through simulations over a real benchmark of safe navigation of ships in the Trondheim fjord as well as comparisons with some other approaches.
- In **Chapter 4**, we first generate a LOS guidance system via a graph-based method, which applies the RRT* (optimal Rapidly-Exploring Random Tree) algorithm. Next, we employ the on-off barrier functions presented in Chapter 3 to guarantee the necessary connectivity distance for information exchange among the agents as well as activate the associated repulsive potential for static and dynamic obstacles. Moreover, a NDO (Nonlinear Disturbance Observer) is considered to reject the disturbances from the ocean that may lead to undesirable performance for the ships. Then, we propose a distributed NDO-NMPC - based algorithm with a threefold purpose: i) track the RRT*- based feasible path through the LOS guidance system, ii) activate the constraints in the view range of the agent for on-line collision avoidance complying with the COLREGs rules and iii) exchange information for connectivity maintenance. Finally, the proposed algorithm is validated through simulations over a real benchmark for the safe navigation of ships in the Trondheim fjord.

- **Chapter 5** completes the thesis with conclusions and discussions on future directions.

The presented organization of the thesis is graphically illustrated in Fig. 1.3.1. Each rectangle box represents a chapter including various processes denoted by ellipses; the colored ellipses would like to emphasize that they are considered as important ingredients in the Ph.D. thesis. Each arrow describes the relation between two chapters, i.e., the chapter at the arrow end uses results (e.g., variables, models, methods, ideas) of the chapter at the arrow origin.

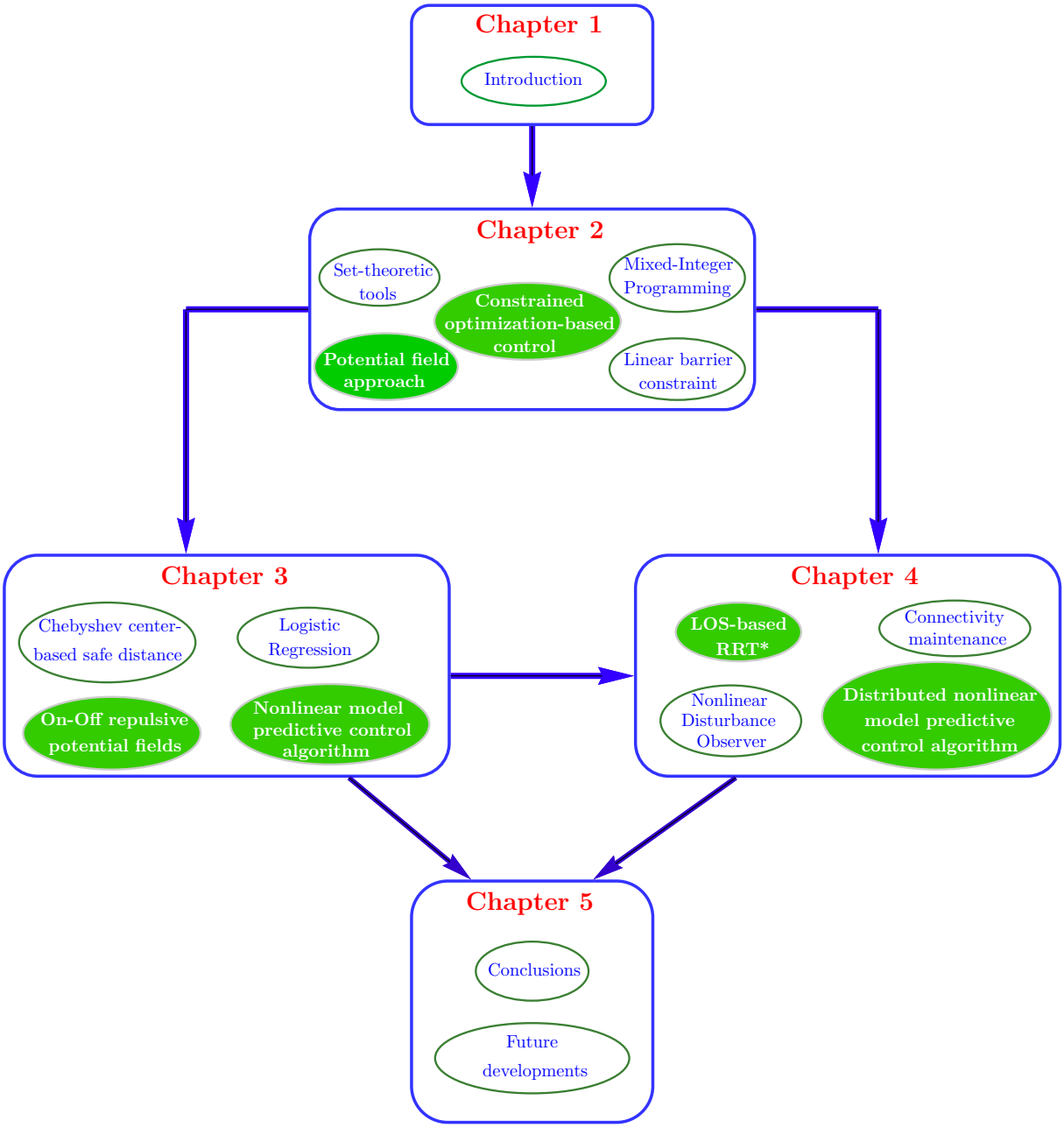


Figure 1.3.1: The organization of the thesis.

Chapter 2

Constrained optimization for motion planning

In motion planning problems it became natural to use constrained optimization-based control approaches (like Model Predictive Control) as they have the ability to handle explicitly state and input constraints for Multi-Agent Systems (MAS). In particular, non-convex constraints for collision avoidance can be explicitly taken into account. The present chapter builds on the application of set-theoretic tools in the context of safety regions around obstacles/agents to ensure collision avoidance for multi-agent systems evolving in a common environment. Furthermore, the *Control Barrier Function* (CBF)-based constraints implying forward invariance of a safe set [Wang et al., 2017a, Ames et al., 2019] is presented and used in conjunction with the a priori mentioned set constructions. This coherent combination demonstrates the effectiveness and flexibility of the optimization-based control approach in handling safety constraints under different forms (i.e., implicit (soft) and explicit (hard) constraints).

"Safety" is the notion concerning collision/obstacle avoidance in the course of normal operation for a MAS. It is worth noting that collision/obstacle avoidance for MAS is commonly one of the challenges that are tough to deal with because it naturally leads to non-convex constraints. The main principle is to guarantee that the forbidden regions bounding agent/dynamic obstacle or static obstacle do not intersect along the trajectory/path of the agents. Throughout the manuscript, the set-theoretic methods will be used to characterize the elements of interest (safety bounds, forbidden regions, etc.).

In what regards the particular set family employed, there are the two main families commonly employed in many research and application fields. On the one hand are polyhedral sets [Motzkin, 1953] and, on the other hand, are ellipsoidal sets [Kurzhanski and Varaiya, 2000]. Their strengths and weaknesses come from their representation which directly restricts their flexibility and the numerical performance of the associated algorithms (e.g., polyhedral sets approximate arbitrarily well any convex set but ellipsoidal sets are more robust wrt operations in high dimensions). Note that polyhedral sets have a dual representation (generator and half-space form) which allow to solve efficiently various problems (in the sense that they can be written more compactly/solved more easily in one representation than in the other) [Blanchini, 1999]. Consequently, in the forthcoming chapters, the polyhedral sets will be employed to describe forbidden regions around agents, static/dynamic obstacles.

Within this set-theoretic interpretation, there are two classes of methods which treat differently

the constraints within the control design scheme: i) optimization-based control combined with mixed-integer formulations [Jünger et al., 2009] which take explicitly into account the non-convex constraints stemming from the collision avoidance conditions, and ii) gradient methods combined with potential field constructions which penalize the non-convex constraints into the cost function. On the one hand, the MIP technique showed their effectiveness in modeling and solving problems subject to non-convex constraints. On the other hand, MIP still has the significant drawback that is an NP-hard problem, which implies an exponential increase in the computational complexity after the number of binary variables used in the problem formulation. The recent results of [Stoican et al., 2011] and further developed in [Prodan et al., 2015], address the problem of reducing the MIP formulations complexity through the efficient description of non-convex regions. While MIP is increasingly used in the literature, there is not much evidence of applications done in a complex environment with many obstacles.
The other significant class of solutions is the one represented by the Potential Field (PF) method, a classical approach to deal with collision/obstacle avoidance. Notably, it appears in the field of motion planning for multiple agents [Koren and Borenstein, 1991], [Latombe, 2012] due to its many benefits, for example, a simple algorithm, convenient mathematical description, and the ease of implementing it for real-time control. The PF-based MPC approach has also been studied and applied in motion planning. Nevertheless, it is still not widely available on this topic; the causes are its own weaknesses, and especially, the stall in local minima. There are several works which consider this approach. [Li et al., 2017] presents the combination of potential field method and MPC scheme for navigating a three-wheeled omnidirectional mobile robot with the goal of avoiding static obstacles. However the paper only uses the potential field to describe motion-based constraints, i.e., potential fields are not penalized in the cost function. In [Chen et al., 2016,Shibata et al., 2018], the potential field is constructed by Euclidean distance between UAV/wheelchair and fixed obstacles, i.e., the obstacles are given pointwise and their actual shape is ignored. [Wang et al., 2019a] used hyperbolic/exponential functions to build the Artificial Potential Field (APF) which are added into the cost function to achieve general obstacle avoidance and the lowest crash severity for autonomous vehicles. The authors of [Yang and Sukkarieh, 2012, Kim et al., 2017, Appapogu, 2019] penalized non-convex constraints generated from the potential field of obstacles in the cost function to establish feasible paths for autonomous car and rotary-wing UAV, respectively. However, they only employ ellipsoidal sets for convex shape approximation of obstacles, and local minima was not discussed. [Prodan et al., 2013] considered polyhedral forbidden regions for the agents and obstacles to ensure collision avoidance concerning trajectory tracking of multi-agents formation in the context of MPC, and this is the primary motivation for enhancements in the use of the potential field-based MPC in this thesis.

The organization of this chapter is as follows. Section 2.1, presents the non-convex workspace of MAS in the presence of static/dynamic obstacles or agents which are described by polyhedral sets. Section 2.2 provides background on receding horizon nonlinear optimization-based control. In Section 2.3, a control barrier function-based constraint will be introduced in the optimization problem to penalize for the distance among the agents. The simulation, comparison, and validation results are given in Section 2.4. Finally, the conclusions recall briefly the main ideas and are presented in Section 2.5.

2.1 Polytopic constraints

First, let us recall some basic notions as well as mathematical descriptions related to the construction of convex regions which will prove useful for the later description of the collision avoidance constraints.

2.1.1 Polytopic descriptions

A polytope is defined as the bounded[1] convex region created by the intersection of a finite number of hyperplanes. Alternatively, it can be seen as the convex sums of its vertices.

Definition 2.1.1 (Hyperplane [Boyd and Vandenberghe, 2004]). *A hyperplane H_k is the set of the form*

$$H_k = \{x \in \mathbb{R}^n \mid a_k x = b_k\}, \tag{2.1.1}$$

where $a_k \in \mathbb{R}^{1\times n}$ is a column vector and $b_k \in \mathbb{R}$ is scalar.

Any hyperplane partitions the space into two disjoint regions (the "half-spaces"):

$$H_k^+ = \{x \in \mathbb{R}^n \mid \quad a_k x \leq \quad b_k\}, \tag{2.1.2}$$
$$H_k^- = \{x \in \mathbb{R}^n \mid -a_k x \leq -b_k\}, \tag{2.1.3}$$

Definition 2.1.2 ($\mathcal{H}$-representation [Boyd and Vandenberghe, 2004]). *A polytope $\mathcal{P} \subset \mathbb{R}^n$ is the intersection of a finite number of half-spaces[2] (H_k^+) and described as*

$$\mathcal{P} = \{x \in \mathbb{R}^n \mid a_k x \leq b_k, \ k = 1, ..., n_h\}, \tag{2.1.4}$$

where $a_k \in \mathbb{R}^{1\times n}$, $b_k \in \mathbb{R}$, n_h is the number of half-spaces. It is worth mentioning that if $b_k > 0$, the origin will belong to the interior of the polytopic region, i.e., $0 \in Int(\mathcal{P})$. Otherwise, the origin is exterior to the polytopic region.

Definition 2.1.3 ($\mathcal{V}$-representation [Boyd and Vandenberghe, 2004]). *The convex hull of a set of a finite-sized collection of points (i.e., the vertices) $\mathcal{C} = \{s_1, \ldots, s_p\}$, with $s_n \in \mathbb{R}^n$ is defined as*

$$\text{Conv}\{\mathcal{C}\} = \Big\{x \in \mathbb{R}^n \mid x = \sum_{n=1}^{p} \alpha_n s_n, \alpha_i \geq 0, \sum_{n=1}^{p} \alpha_n = 1\Big\}, \tag{2.1.5}$$

where s_n, $n = 1, ..., p$, is n^{th} point from the collection of vertices and α_n is the weight associated to it.

Example 2.1.4. The origin is in the interior of polytope $\mathcal{P}$ in case of $b_k > 0$ as in Fig. 2.1.1 and can be constructed in two ways. Its $\mathcal{H}$-representation is

$$\mathcal{P} = \left\{ x \in \mathbb{R}^2 : \begin{bmatrix} 0.0716 & -0.3581 \\ 0.5270 & 0.4216 \\ -0.5523 & 0.0921 \end{bmatrix} x \leq \begin{bmatrix} 0.9309 \\ 0.7379 \\ 0.8285 \end{bmatrix} \right\}$$

and its $\mathcal{V}$-representation is

$$\mathcal{P} = \left\{ x \in \mathbb{R}^2 : x = \alpha_1 \begin{bmatrix} -2 \\ -3 \end{bmatrix} + \alpha_2 \begin{bmatrix} -1 \\ 3 \end{bmatrix} + \alpha_3 \begin{bmatrix} 3 \\ -2 \end{bmatrix}, \alpha_n \geq 0, \sum_{n=1}^{3} \alpha_n = 1 \right\}$$

[1] Note that in general a intersection of hyperplanes does not necessarily lead to a bounded polyhedron.

[2] Here we took by convention the sign " $+$ ", any other combination of signs would lead to a polytope (or to the empty set if the intersection of the associated half-spaces is empty)

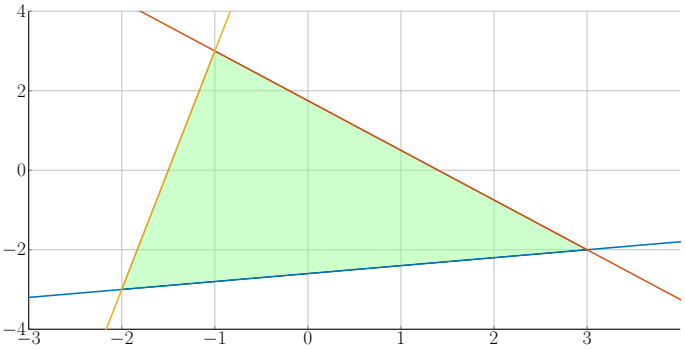


Figure 2.1.1: Polytope in case of the origin $\in Int(\mathcal{P})$.

Example 2.1.5. The origin is outside of polytope $\mathcal{P}$ in case of $b_k \leq 0$ as depicted in Fig. 2.1.2 and it can also be built in two ways. Its $\mathcal{H}$-representation is

$$\mathcal{P} = \left\{ x \in \mathbb{R}^2 : \begin{bmatrix} -0.2691 & -0.2018 \\ 0.1871 & -0.0234 \\ 0.1156 & 0.1445 \\ -0.1046 & 0.0262 \end{bmatrix} x \leq \begin{bmatrix} 0.9417 \\ -0.9821 \\ 0.9827 \\ 0.9942 \end{bmatrix} \right\}$$

and its $\mathcal{V}$-representation is

$$\mathcal{P} = \left\{ x \in \mathbb{R}^2 : x = \alpha_1 \begin{bmatrix} -4 \\ 10 \end{bmatrix} + \alpha_2 \begin{bmatrix} -8 \\ 6 \end{bmatrix} + \alpha_3 \begin{bmatrix} -5 \\ 2 \end{bmatrix} + \alpha_4 \begin{bmatrix} -6.5 \\ 12 \end{bmatrix}, \alpha_n \geq 0, \sum_{n=1}^{4} \alpha_n = 1 \right\}$$

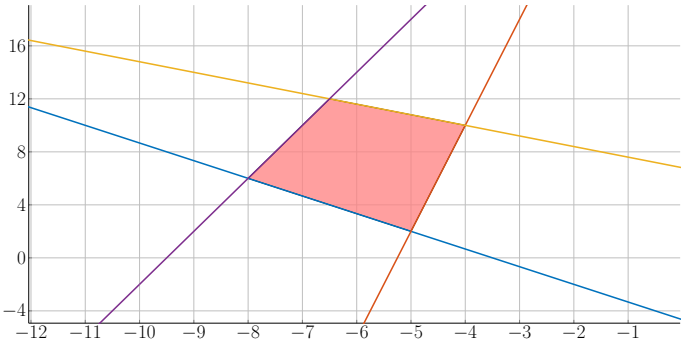


Figure 2.1.2: Polytope in case of the origin $\notin Int(\mathcal{P})$.

2.1.2 Sum function

The so-called "sum function" is piecewise linear function which is introduced by [Camacho and Bordons, 2004] and whose definition is based on the $\mathcal{H}$-representation of polytope in (2.1.4):

$$\gamma(x) = \sum_{k=1}^{n_h} (a_k x - b_k + |a_k x - b_k|). \tag{2.1.6}$$

By analyzing the sum function (2.1.6), it is obvious that this piecewise affine function is zero whenever $x \in \mathcal{P}$ and strictly positive whenever $x \notin \mathcal{P}$:

- If $\exists x \in \mathbb{R}^n$ such that: $a_k x - b_k < 0$, $\forall k = 1 \dots n_h$, then

$$\gamma(x) = \sum_{k=1}^{n_h} (a_k x - b_k - (a_k x - b_k)) = 0. \tag{2.1.7}$$

  This means that the sum function has zero value in the interior of the bounded convex region (2.1.4).

- If $\exists x \in \mathbb{R}^n$ such that: $a_k x - b_k \geq 0$, $\forall k = 1 \dots n_h$, then

$$\gamma(x) = \sum_{\forall k}^{\exists k} (a_k x - b_k + (a_k x - b_k)) = 2 \sum_{\forall k}^{\exists k} (a_k x - b_k). \tag{2.1.8}$$

  In other words, the value of the piecewise linear function will grow piecewise[3] linearly with the distance from the polytopic region (2.1.4).

Furthermore, (2.1.8) can be piecewise differentiable as long as each piece is differentiable throughout its subdomain, though the unity function may not be differentiable at the breakpoints[4] between the pieces. Therefore, the gradient of the sum function is expressed as

$$\nabla\gamma(x) = 2 \sum_{\forall k}^{\exists k} a_k^\top. \tag{2.1.9}$$

The generalized gradient in (2.1.9) is multivalued, i.e., $\nabla\gamma(x)$ has n values, one per each dimension of variable $x \in \mathbb{R}^n$. However, this would not be too complicated to compute when using the result of the gradient for the implicit/explicit methods if its range of variation is still bounded. Hence, it is possible to use such a function for applications in the sequel.

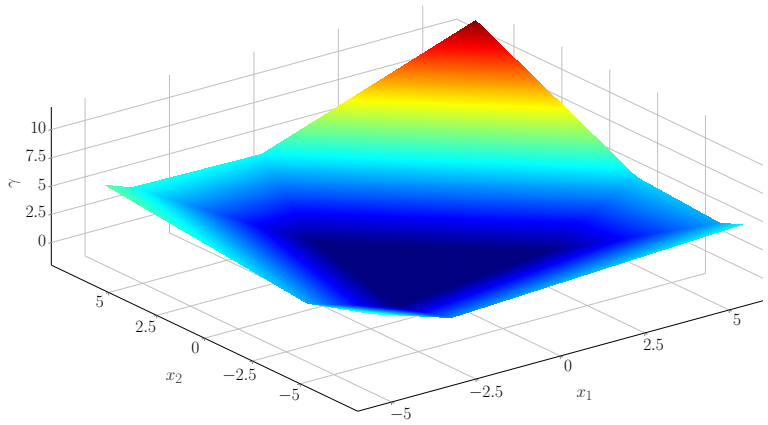


Figure 2.1.3: Sum function of the triangular polytope in Fig. 2.1.1.

Fig. 2.1.3 illustrates the sum function obtained from the polytopic region as in Fig. 2.1.1. We note that, as mentioned earlier, inside the obstacle, the value of the function is zero and outside is non-zero.

[3] As long as the combination of active/inactive indices k does not change, the formula (2.1.8) remains, the same and is linear.

[4] Those are the points where the slope of sum function changes.

2.2 Nonlinear optimization in control

Nonlinear optimization in control is used to achieve certain optimality criterions for the given (sub)-systems which are related to the nonlinear dynamic model. We arrive at such formulation via a, (non-)quadratic cost function and/or nonlinear constraints on the states and inputs. This section will recall first the formulation of nonlinear optimization-based control in the presence of constraints. Next, the two kinds of methods which tackle the collision avoidance problem with static/dynamic obstacles in explicit and implicit processes. Usually, the explicit methods reduce to the use of MIP where the constraints are explicitly taken into account, and the implicit methods generally reduce to the use of various types of potential field-based constructions. Finally, MPC is presented as a useful tool to deal with motion planning for MAS.

We present the basic formulation of nonlinear optimal control problem in relation to the dynamical system, objective function, and constraints in continuous-time. First of all, consider the nominal dynamic system of i^{th} agent in the set $\mathcal{V}$ comprising $N_a \in \mathbb{N}$ agents, which compose a multi-agent dynamical system:

$$\dot{\mathbf{x}}_i(t) = f_i(\mathbf{x}_i(t), \mathbf{u}_i(t)),\ \mathbf{x}_i(t_0) = \mathbf{x}_{i0},\ i \in \mathcal{V},\ \mathcal{V} = 1, \ldots, N_a, \tag{2.2.1}$$

subject to input and state constraints:

$$\mathbf{x}_i(t) \in \mathcal{X}_i,\ \forall t \geq 0, \tag{2.2.2}$$
$$\mathbf{u}_i(t) \in \mathcal{U}_i,\ \forall t \geq 0. \tag{2.2.3}$$

$\mathbf{x}_{i0}$ is the initial state at initial time t_0, $\mathbf{x}_i(t) \in \mathbb{R}^n$ is the current state, $\mathbf{u}_i(t) \in \mathbb{R}^m$ is the control input with respect to the vector field f_i: $\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ which we suppose to be locally Lipschitz continuous in the region of interest and satisfying $f_i(0,0) = 0$. It is worth mentioning that the sets $\mathcal{X}_i, \mathcal{U}_i$ are compact and $(0,0) \in \mathcal{X}_i \times \mathcal{U}_i$. Furthermore, $\mathcal{X}_i, \mathcal{U}_i$ are convex sets which characterize magnitude constraints[5]:

$$\mathcal{X}_i = \{\mathbf{x}_i \in \mathbb{R}^n | \mathbf{x}_{imin} \leq \mathbf{x}_i \leq \mathbf{x}_{imax}\}, \tag{2.2.4}$$
$$\mathcal{U}_i = \{\mathbf{u}_i \in \mathbb{R}^m | \mathbf{u}_{imin} \leq \mathbf{u}_i \leq \mathbf{u}_{imax}\}, \tag{2.2.5}$$

where $\mathbf{x}_{imin}, \mathbf{x}_{imax}$, $\mathbf{u}_{imin}, \mathbf{u}_{imax}$ are constant vectors.

The objective of MAS in (2.2.1) will be obtained by minimizing a cost function over a finite horizon T_p and given as:

Problem 2.2.1.

$$\text{minimize } J_i(\boldsymbol{x}_i[t, t+T_p], \boldsymbol{u}_i[t, t+T_p]) \tag{2.2.6a}$$

subject to

$$\dot{\boldsymbol{x}}_i(t) = f(\boldsymbol{x}_i(t), \boldsymbol{u}_i(t)),\ \boldsymbol{x}_i(t_0) = \boldsymbol{x}_{i0},\ \forall t \in [t, t+T_p], \tag{2.2.6b}$$
$$\boldsymbol{x}_i(t) \in \mathcal{X}_i;\ \boldsymbol{u}_i(t) \in \mathcal{U}_i,\ \forall t \in [t, t+T_p]. \tag{2.2.6c}$$

[5]Note that more complex sets are possible but it is often the case that magnitude constraints suffice in practical applications.

The physical and operational limitations of MAS is described by the box constraints (2.2.6c), (2.2.6b) presents the evolution of state $\mathbf{x}_i(t)$ depending on the control input $\mathbf{u}_i(t)$. The cost functional $J_i(\cdot)$ is typically described by

$$J_i(\mathbf{x}_i[t, t+T_p], \mathbf{u}_i[t, t+T_p]) = \int_t^{t+T_p} \big[\mathcal{L}_i(\mathbf{x}_i(t), \mathbf{u}_i(t))\big] dt + E_i(\mathbf{x}_i(t+T_p)), \tag{2.2.7}$$

where $\mathcal{L}_i(\cdot)$: $\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^+$ is called running cost or stage cost, terminal penalty term $E_i(\cdot)$: $\mathbb{R}^m \to \mathbb{R}^+$ might or might not be present, and $\mathbf{x}_i(t+T_p)$ is the terminal state. This cost is determined by the desired objective of MAS, e.g., tracking errors, energy, or time minimization.

It is evident that solving problem 2.2.1 is not trivial; in particular, nonlinear formulations are exceedingly difficult to solve. In order to cope with these difficulties, numerical methods are considered most promising and popular. They are efficient in discretization and finite parameterization and introduced in the nonlinear optimization-based control problem and will be detailed later in subsection 2.2.3.

In the following, potential field approach, mixed-integer programming, and MPC will provide us the necessary tools for accomplishing motion planning for MAS.

2.2.1 Potential field approach

From Section 2.1, let us first define the working space of MAS, $\mathcal{F} \subseteq \mathbb{R}^n$, which comprises a union of forbidden polytopic convex regions characterizing multiple static obstacles, as defined in (2.1.4):

$$\mathcal{F} \supset \mathbb{O}, \tag{2.2.8}$$

where $\mathbb{O}$ is defined as follows:

$$\mathbb{O} = \bigcup_{\ell=1}^{N_{\text{obs}}} \mathcal{O}_\ell, \tag{2.2.9}$$

where $\mathcal{O}_\ell = \{\mathbf{x}_i \in \mathbb{R}^n \mid a_k^\ell \mathbf{x}_i \leq b_k^\ell, k = 1, \ldots, n_h^\ell\}$, with $a_k^\ell \in \mathbb{R}^{1\times n}$, $b_k^\ell \in \mathbb{R}$, n_h^ℓ is the number of half-spaces describing $\mathcal{O}_\ell$, $\ell = 1, \ldots, N_{\text{obs}}$, and N_{obs} is the number of forbidden polytopic regions accounting for static obstacles.

Potential field is a classical approach for collision/obstacle avoidance which first proposed by [Khatib, 1986]. The main idea of this approach is to generate repulsive potential fields around the forbidden regions to force the MAS to avoid them and attractive potential fields around the desired target to attract the MAS. We denote by $S(\cdot) : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ the potential field which is the summation of repulsive and attractive potential fields and is given as:

$$S(\mathbf{x}_i) = \sum_{\ell=1}^{N_{\text{obs}}} \Phi_\ell(\mathbf{x}_i) + \Xi(\mathbf{x}_i), \tag{2.2.10}$$

where $\Xi(\mathbf{x}_i) : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is attractive potential and $\Phi_\ell(\mathbf{x}_i)$ denotes the repulsive potential with respect to the ℓ^{th} obstacle, $\ell = 1, \ldots, N_{\text{obs}}$.

Remark 2.2.2. The operating space of MAS is also called the feasible region. Therefore, this is a non-convex region due to the fact that it is the complement of a convex union of regions (coming from the safety regions of the agents, from the fixed or moving obstacles).

A quadratic function (see Fig. 2.2.1) is most widely used for representing an attractive potential field and given as:

$$\Xi(\mathbf{x}_i) = \|\mathbf{x}_i - \mathbf{x}_{i,ref}\|^2_{\mathbf{Q}_i}, \tag{2.2.11}$$

where $\mathbf{x}_{i,ref}$ denotes the reference destination of the i^{th} agent and $\mathbf{Q}_i$ is the weighting matrix.

Fig. 2.2.1 presents the attractive potential field in (2.2.11) at the desired point, $\mathbf{x}_{i,ref}$ is the origin.

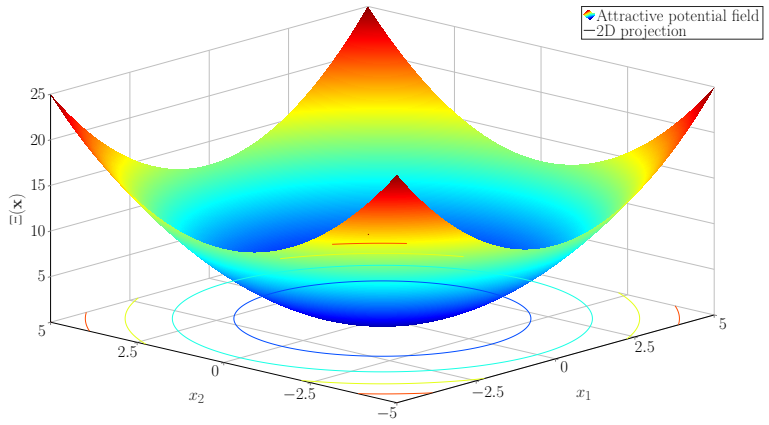


Figure 2.2.1: Attractive potential field with quadratic function.

The repulsive potential is the most challenging component because it has to both approximate the obstacles from the geometrical point of view and ensure collision avoidance. Therefore, obstacles described by polytopic constraints as (2.1.4) or (2.1.5) must have a high value inside in order to generate a downhill gradient such that the MAS path converges towards the destination. With such an idea, the repulsive potential field can be defined by making use of the sum function (2.1.6) in two ways:

- First, in combination with a standard fractional function used in the repulsive potential generation:

$$\Phi^f_\ell(\gamma_\ell(\mathbf{x}_i)) = \frac{c_{1\ell}}{(c_{2\ell} + \gamma_\ell(\mathbf{x}_i))^2}, \tag{2.2.12}$$

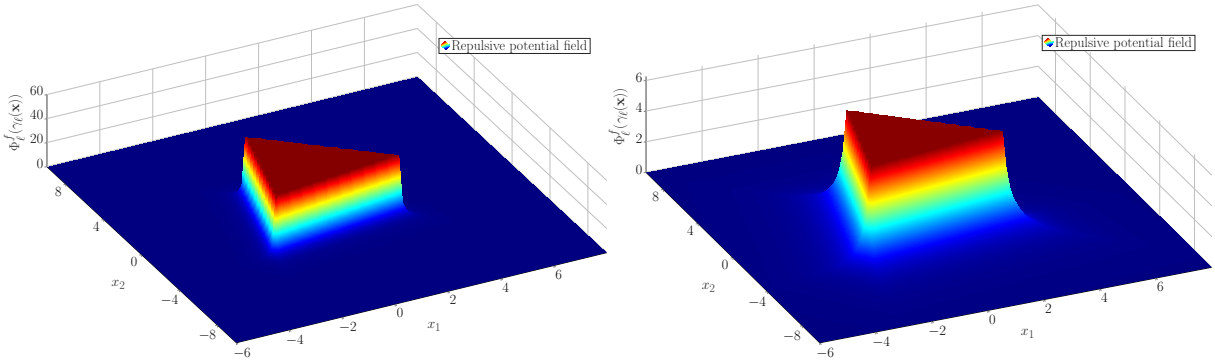
  where $c_{1\ell}$ and $c_{2\ell}$ are positive parameters representing the strength and effect ranges of repulsive potential, $\gamma_\ell(\cdot)$ denotes sum function of the ℓ^{th} polytope.

- Second, the repulsive potential field is constructed with help of an exponential function:

$$\Phi^e_\ell(\gamma_\ell(\mathbf{x}_i)) = c_{3\ell} e^{-(\gamma_\ell(\mathbf{x}_i) - c_{4\ell})^2}, \tag{2.2.13}$$

where $c_{3\ell}$ and $c_{4\ell}$ are positive parameters representing the strength and effect ranges of the repulsive potential.

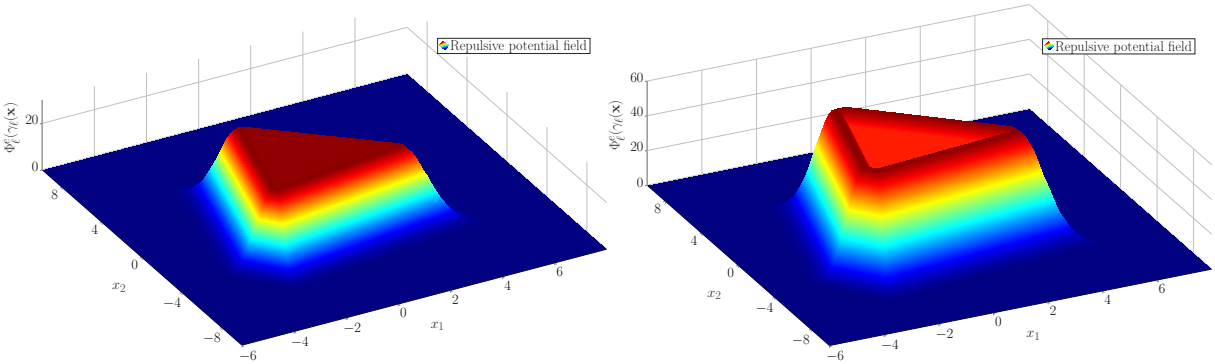
Two repulsive potentials as described in (2.2.12) and (2.2.13) map the domain $(0, \infty)$ of the sum function (2.1.6) into $\left(\frac{c_{1\ell}}{(c_{2\ell})^2}, 0\right)$ and $\left(c_{3\ell} e^{c_{4\ell}}, 0\right)$, respectively. As a result, this means the high value is in the interior of the convex region and plummets to zero as the distance from the convex region increases, i.e., there is a decrease towards 0 for $\gamma(\mathbf{x}_i) \to \infty$.



(a) Repulsive potential using fractional function with $c_1 = 0.5$, $c_2 = 0.1$.

(b) Repulsive potential using fractional function with $c_1 = 1$, $c_2 = 0.4$.

Figure 2.2.2: Repulsive potential fields using sum function (2.1.6) with the difference of strength and effect ranges (c_1, c_2) depicted as Fig. 2.1.3.



(a) Repulsive potential using exponential function with $c_3 = 30$, $c_4 = 0.1$.

(b) Repulsive potential using exponential function with $c_3 = 60$, $c_4 = 0.4$.

Figure 2.2.3: Repulsive potential fields using sum function (2.1.6) with the difference of strength and effect ranges (c_3, c_4) depicted as Fig. 2.1.3.

Fig. 2.2.2 and Fig. 2.2.3 illustrate the repulsive potential functions in two ways with various strength and effect ranges. In the sequel, we use repulsive potential field constructed by the fractional function because this method yields a nice shape from the geometrical viewpoint.

As aforementioned, the potential field is generated by the combination of repulsive and attractive potential functions. The slope (or gradient) is a superposition of the slopes generated by the attractive potential which steers the MAS towards the goal configuration, and by the slope spawned by the repulsive fields of the obstacles, which push agents away. The confluence of these influences results in a path which (under a correct parameter choice) drives the MAS to the desired destination while simultaneously avoiding the obstacles.

Therefore, the simplest and the most common method for finding the collision-free path for MAS is to use a negative gradient of the potential function (2.2.10) in order to obtain the force vector field being able to both repel agents away from the obstacles and to attract the agents towards the goal configuration. This force vector field, in space containing N_{obs} has the analytic

formula:

$$F_i(\mathbf{x}_i) = -\nabla S(\mathbf{x}_i) = -\sum_{\ell=1}^{N_{\text{obs}}} \left[\frac{c_{1\ell}}{(c_{2\ell}+\gamma_\ell(\mathbf{x}_i))^3}\right] \nabla \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i) - 2\mathbf{Q}_i(\mathbf{x}_i - \mathbf{x}_{i,ref})\nabla\mathbf{x}_i, \tag{2.2.14}$$

or

$$F_i(\mathbf{x}_i) = -\nabla S(\mathbf{x}_i) = -\sum_{\ell=1}^{N_{\text{obs}}} \left[c_{3\ell} e^{-(\gamma_\ell(\mathbf{x}_i)-c_{4\ell})^2}\right] \nabla 2 \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i) - 2\mathbf{Q}_i(\mathbf{x}_i - \mathbf{x}_{i,ref})\nabla\mathbf{x}_i, \tag{2.2.15}$$

where $\nabla \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i)$ in (2.2.14) and (2.2.15) was presented in (2.1.9).

Note that eq. (2.2.14) and (2.2.15) use fractional function (2.2.12) and exponential function (2.2.13) for repulsive potential, respectively.

Remark 2.2.3. Whenever the combination of active half-spaces does not change, the sum function remains constant. In turn, this means that the potential field gradient remains constant. In particular, if the agent enter the interior of the obstacle, there is no gradient to push towards the boundary.

Example 2.2.4 (Potential field). A potential field is established from repulsive potential of two polytopic regions as described in (2.1.5) and attractive potential (2.2.11).

The two polytopes stand for two fixed obstacles ($N_{\text{obs}} = 2$), given by their convex hull ($\mathcal{V}$-representation) as:

$$\mathcal{O}_1 = \text{Conv}\left\{\begin{bmatrix}-4\\10\end{bmatrix}, \begin{bmatrix}-8\\6\end{bmatrix}, \begin{bmatrix}-5\\2\end{bmatrix}\right\}. \tag{2.2.16}$$

$$\mathcal{O}_2 = \text{Conv}\left\{\begin{bmatrix}10\\5\end{bmatrix}, \begin{bmatrix}5\\0\end{bmatrix}, \begin{bmatrix}12\\0\end{bmatrix}\right\}. \tag{2.2.17}$$

They are also rewritten under the equivalent $\mathcal{H}$-reprsentation:

$$\mathcal{O}_1 = \left\{\mathbf{x}_i \in \mathbb{R}^2 : \begin{bmatrix}0.1871 & -0.0234\\-0.0711 & 0.0711\\-0.2691 & -0.2018\end{bmatrix}\mathbf{x}_i \leq \begin{bmatrix}-0.9821\\0.9949\\0.9417\end{bmatrix}\right\}$$

$$\mathcal{O}_2 = \left\{\mathbf{x}_i \in \mathbb{R}^2 : \begin{bmatrix}0 & -1\\0.0830 & 0.0332\\-0.1925 & 0.1925\end{bmatrix}\mathbf{x}_i \leq \begin{bmatrix}0\\0.9960\\-0.9623\end{bmatrix}\right\}$$

This allows to construct the sum function as per its definition in (2.1.6)

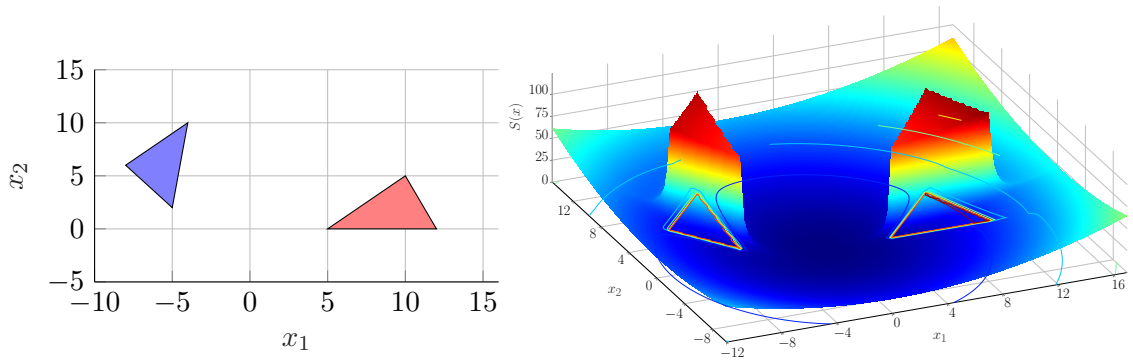
$$\gamma_1(\mathbf{x}_i) = \left(\begin{bmatrix}0.1871 & -0.0234\\-0.0711 & 0.0711\\-0.2691 & -0.2018\end{bmatrix}\mathbf{x}_i - \begin{bmatrix}-0.9821\\0.9949\\0.9417\end{bmatrix} + \left|\begin{bmatrix}0.1871 & -0.0234\\-0.0711 & 0.0711\\-0.2691 & -0.2018\end{bmatrix}\mathbf{x}_i - \begin{bmatrix}-0.9821\\0.9949\\0.9417\end{bmatrix}\right|\right)$$

$$\gamma_2(\mathbf{x}_i) = \left(\begin{bmatrix}0 & -1\\0.0830 & 0.0332\\-0.1925 & 0.1925\end{bmatrix}\mathbf{x}_i - \begin{bmatrix}0\\0.9960\\-0.9623\end{bmatrix} + \left|\begin{bmatrix}0 & -1\\0.0830 & 0.0332\\-0.1925 & 0.1925\end{bmatrix}\mathbf{x}_i - \begin{bmatrix}0\\0.9960\\-0.9623\end{bmatrix}\right|\right)$$

From the two sum functions collected, we are able to apply either fractional or exponential function as in (2.2.12), (2.2.13) for obtaining the repulsive potential. In this example, we use the formulation (2.2.12) with the coefficient of strength $c_1 = 1$ and effect range $c_2 = 0.1$.

Together with an attraction potential as aforementioned in (2.2.11) at the origin, we have that the potential field in this case is:

$$S(\mathbf{x}_i) = \sum_{\ell=1}^{2} \Phi_\ell(\mathbf{x}_i) + \Xi(\mathbf{x}_i). \tag{2.2.18}$$



(a) Two polytopic regions as in (2.2.16) and (2.2.17).

(b) Potential field generated by the two polytope in Fig. 2.2.4a and attractive potential at origin.

Figure 2.2.4: Potential field description for workspace of MAS.

Example 2.2.5 (Collision-free path for MAS). Assume that we have a MAS composed of $N_a = 1$ agents which operates in the space defined in Example 2.2.4 with dynamics in a two dimensional state-space:

$$\mathbf{A}_i = \begin{bmatrix} 0.5755 & 0.0614 \\ -0.6578 & 1.0245 \end{bmatrix}, \ \mathbf{B}_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

where $i = 1$.

Then, agent's path results by using the negative gradient for potential field (2.2.18) as in (2.2.14). In Fig. 2.2.5, the concentric circles whose center is at the origin depict the attractive potential

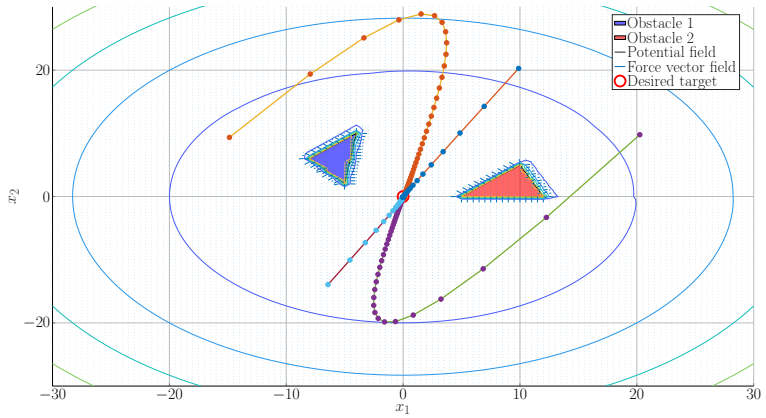


Figure 2.2.5: Motion planning of agent with different initial points.

projected onto the 2D plane. The force vectors around two obstacles tend to go out to repel the agents away. Although the potential field-based motion planning for MAS can deal with the fixed obstacles in convergence towards goal configuration from different initial points, the agent can still be trapped by local minima where the summation of repulsive force and attractive force is zero. This main drawback needs to have a relevant solution if we choose the potential

field approach for motion planning. Also, as in this example, we did not take into account the saturation of state or control input. These issues are tackled by constrained optimization-based control through minimizing a given cost over a finite horizon. Details will be presented in the next section.

2.2.2 Mixed-integer programming

As presented in the previous section, the workspace in which the MAS lies is a non-convex region due to the avoidance of convex regions describing static, dynamic obstacles and other prohibited areas where the agent is not able to enter. Besides the potential field approach, MIP techinique is an alternative method which copes with the challenges of having a non-convex feasible region.

MIP is used to describe avoidance constraints into a mixed-integer form which are further handled by optimization-based control with the aim is to prevent the collision between MAS and fixed/moving obstacles. Let us first recall a result of polytope representing the forbidden region of the obstacle in general.

Let us consider a collection of hyperplanes and, by convention, consider that their "+" half-spaces define a polytope $\mathcal{P}$ (our obstacle). Then, (2.1.4) can be rewritten:

$$\mathcal{P} = \bigcap_k H_k^+ = \{x \in \mathbb{R}^n \mid a_k x \leq b_k,\ k = 1, ..., n_h\},$$

as the intersection of the half-spaces H_k^+. The complement of $\mathcal{P}$, $\mathcal{C}(\mathcal{P})$[6], denotes the feasible space in which the agent can move unimpeded. Using the property that $\mathcal{C}(A \cap B) = \mathcal{C}(A) \cup \mathcal{C}(B)$ and that $\mathcal{C}(H_k^+) = H_k^-$ we may write:

$$\mathcal{C}(\mathcal{P}) = \bigcup_k H_k^- = \bigcup_k \{x \in \mathbb{R}^n | - a_k x \leq -b_k\}. \tag{2.2.19}$$

It is worth noting that the non-convex feasible region (2.2.19) of MAS is described by a union of the elementwise of the regions (2.1.3) associated with the k^{th} inequality of (2.1.4).

Example 2.2.6. For more clarity, go back to Example 2.1.5 for a description of the plotope $\mathcal{P}$ and is complement $\mathcal{C}(\mathcal{P})$ in detail, we have:

$$\mathcal{P} = \bigcap_k H_k^+ = \left\{ x \in \mathbb{R}^2 : \begin{bmatrix} -0.2691 & -0.2018 \\ 0.1871 & -0.0234 \\ 0.1156 & 0.1445 \\ -0.1046 & 0.0262 \end{bmatrix} x \leq \begin{bmatrix} 0.9417 \\ -0.9821 \\ 0.9827 \\ 0.9942 \end{bmatrix} \right\}$$

$$\mathcal{C}(\mathcal{P}) = \bigcup_k H_k^- = \left\{ x \in \mathbb{R}^2 : - \begin{bmatrix} [-0.2691 & -0.2018] \\ [0.1871 & -0.0234] \\ [0.1156 & 0.1445] \\ [-0.1046 & 0.0262] \end{bmatrix} x \leq - \begin{bmatrix} [0.9417] \\ [-0.9821] \\ [0.9827] \\ [0.9942] \end{bmatrix} \right\}$$

Fig. 2.2.6 shows clearly the bounded polyhedron ($\mathcal{P}$) where agents are not allowed to enter, and the feasible region H_k^- where agents operate, is the k^{th} complement of the associated inequality of prohibited region.

[6]The complement of bounded convex region $\mathcal{P}$ in (2.1.4) is defined by $\mathcal{C}(\mathcal{P}) = cl(\mathcal{F} \backslash \mathcal{P})$ where $\mathcal{F}$ was defined in (2.2.8).

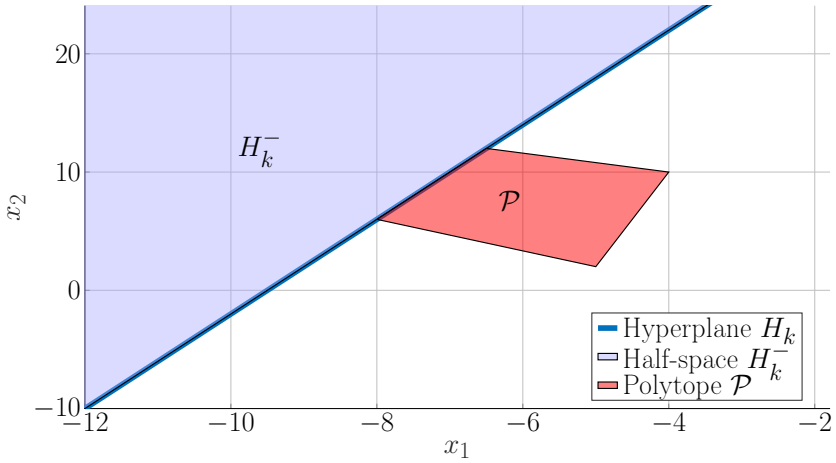


Figure 2.2.6: Polytope $\mathcal{P}$ in Fig. 2.1.2 and its complement $\mathcal{C}(\mathcal{P})$.

Imposing MAS to work in the non-convex domain (2.2.8) is infeasible for the optimization-based control problem. That is the reason why MIP technique is considered as the best solution to deal with this difficulty. The fundamental idea is to introduce auxiliary binary variables ($z \in \{0,1\}$) [Taha, 2011] and M as a sufficiently large positive constant. The mixed-integer, linear constraints describing the nonconvexity of space concerning i^{th} agent of MAS:

$$-a_k \mathbf{x}_i \leq -b_k + M z_k, \; k = 1, \ldots, n_h \tag{2.2.20}$$

$$\sum_{k=1}^{n_h} z_k \leq n_h - 1, \tag{2.2.21}$$

where $z_k \in \{0,1\}^{n_h}$.

The constraint (2.2.20) means that if $z_k = 1$, then the right-hand side of (2.2.20) is a large positive number, i.e., k-th half-space will be relaxed (i.e., ignored). The constraint (2.2.21) ensures that at least one constraint of (2.2.20) has to activate, i.e., at least one binary variable of the set $(z_1, \ldots, z_{n_h})$ is zero.

The auxiliary binary variable is used as "on-off switch" for the appearance of big M value associated to each inequality constraint in the description of the bounded convex region. However, it is necessary to choose a reasonable value for scalar M because it can cause conditioning problems and makes the MIP resolution more complicated.

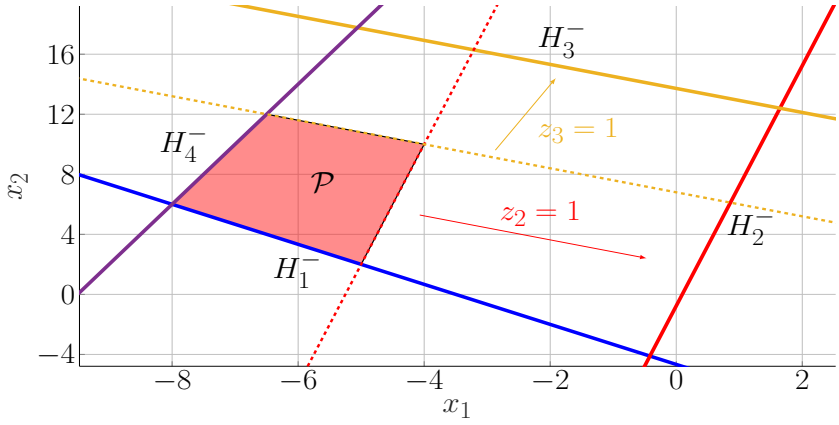


Figure 2.2.7: Offset of the two half-spaces H_2^-, H_3^- with M in Fig. 2.2.6.

Fig. 2.2.7 illustrates polytope $\mathcal{P}$ in Fig. 2.2.6 with the offsets of half-space H_2^- and H_3^- with M assumed to be 1, and $z_2 = z_3 = 1$ ($z_1 = z_4 = 0$). Note that, here, we choose $M = 1$ to be able to observe easily. In the case of M sufficiently large, the 2nd and 3rd inequalities from (2.2.20) will be far away from the region in which the obstacle lies.

A collision-free path can be found by minimizing a quadratic objective function whose variables must satisfy the state-space representation of the multi-agent dynamical systems subject to mixed-integer constraints. As aforementioned, to deal with constrained optimization-based control problem in continuous-time is not trivial due to the numerical issues. Consequently, it will be approximated by the discrete-time optimization problem and presented in the next section.

2.2.3 (Non-)linear model predictive control

Model Predictive Control (MPC) is an outstanding method for solving constrained optimization problems. One possible classification (depending on the purpose considered, other classifications may be made) is into: (i) economic MPC, which uses a "profit"-related objective function for managing the portfolio of energy usage in residential and industrial projects [Chen et al., 2013, Dombrovskii and Obyedko, 2015]; (ii) tracking MPC, which penalizes the error between actual and reference profiles such as state, input and output variables [Magni et al., 2001, Limón et al., 2008]. However, in this dissertation, we only focus on tracking MPC as it is the standard construction for the motion planning of multi-agent dynamical systems. Unlike linear MPC which only considers linear constraints and linear or quadratic costs, nonlinear MPC deals with nonlinear dynamical models and (non-)quadratic cost function as well as general nonlinear constraints on the states and inputs.

In one interpretation, MPC can be understood as a particular type of on-line optimization-based control where a finite-horizon optimal control problem is solved at each sampling instant. Due to the fact that the states of the prediction system are updated continuously at each sampling period, a new optimization problem will emerge and be solved at each sampling interval. It is worth noting that MPC always works with the nominal model (2.2.1) of MAS, i.e., the unknown disturbance components, model uncertainty, or measurement errors are not taken into account (although there are MPC variants which are robust against disturbances, e.g., stochastic or min-max MPC [Findeisen et al., 2007, Allgöwer and Zheng, 2012]). In order to implement practically MPC for a real-time control framework, one needs to pay attention to some of the following issues:

- Discretize the nominal model (2.2.1) with given initial conditions to be handled by numerical integration method[7].
- Replace continuous variables of objective, constraints function with discrete variables.
- A finite-dimensional optimization problem is defined by a finite number of decision variables that can be solved using numerical optimization.

In order to discretize the optimal control problem directly, the *direct multiple shooting* [Bock and Plitt, 1984, Leineweber et al., 2003] is preferred to *direct single shooting* [Hicks and Ray, 1971, Kraft, 1985] since the constraint objective function evaluation is much simpler and as a

[7]This is the approximate computation of an integral using numerical techniques.

result of structural properties of the equations has numerical advantages which can be further exploited [Johansen, 2011].

Therefore, applying the direct multiple shooting technique to partition a finite time horizon $[t, t+T_p]$ into N_p subintervals $[t_k, t_{k+1}]$, and t_{k+1} is determinded by $t_{k+1} = t_k + T_e$, $k = 0, \ldots, N_p - 1$, where $T_e = T_p/N_p$ is the time step and T_p gives the length of the prediction horizon. Then, we solve first a finite horizon open-loop OCP (Optimal Control Problem) (as presented in Problem 2.2.1) at discrete sampling instants t_k, based on the information of the current state measurement $\mathbf{x}_i(t_k)$. The open-loop input signal applied between the sampling instants is given by the solution of OCP:

Problem 2.2.7 (Tracking MPC)**.**

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad J_i(\mathbf{x}_i(t_k), \bar{\mathbf{u}}_i(\cdot)) \tag{2.2.22a}$$

subject to

$$\dot{\bar{\boldsymbol{x}}}_i(\tau) = f_i(\bar{\boldsymbol{x}}_i(\tau), \bar{\boldsymbol{u}}_i(\tau)),\ \bar{\boldsymbol{x}}_i(t_k) = \boldsymbol{x}_i(t_k), \tag{2.2.22b}$$

$$\bar{\boldsymbol{x}}_i(\tau) \in \mathcal{X}_i;\ \bar{\boldsymbol{u}}_i(\tau) \in \mathcal{U}_i,\ \forall \tau \in [t_k, t_k + T_p], \tag{2.2.22c}$$

$$\bar{\mathbf{x}}_i(t_k + T_p) \in \mathcal{X}_i^f, \tag{2.2.22d}$$

where $\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\tau)$ are the predicted states and inputs while $\bar{\mathbf{u}}_i(.)$ represents the predicted input trajectory along the prediction horizon T_p. The bar symbol is employed for distinguishing predicted states which are internal to the controller, i.e., $\bar{\mathbf{x}}_i(\tau)$ is the solution of (2.2.22b) at time $\tau \in [t_k, t_k + T_p]$ based on the initial condition $\mathbf{x}_i(t_k)$ by application of the input $\bar{\mathbf{u}}_i(\cdot)$: $[t_k, t_k + T_p] \to \mathcal{U}_i$. It is worth mentioning that the predicted states will not be equal to the actual closed-loop values if the system is under the presence of disturbances, model uncertainty, or measurement errors. The cost functional $J_i(\cdot)$ minimized over the prediction horizon T_p is basically presented as follows:

$$J_i(\mathbf{x}_i(t_k), \bar{\mathbf{u}}_i(\cdot)) = E_i(\bar{\mathbf{x}}_i(t_k + T_p)) + \int_{t_k}^{t_k+T_p} \big[\mathcal{L}_i(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\tau))\big] d\tau, \tag{2.2.23}$$

where the stage cost $\mathcal{L}_i(\cdot)$: $\mathcal{X}_i \times \mathcal{U}_i \to \mathbb{R}_{\geq 0}$, terminal cost $E_i(\cdot)$: $\mathcal{X}_i \to \mathbb{R}_{\geq 0}$ and $f(\cdot)$ are assumed to be continuous; for all $t \geq t_0$, $\mathcal{L}_i(0, 0) = 0$, $E_i(0) = 0$ and $f_i(0, 0) = 0$. Note that there exist class $\mathcal{K}$ functions[8] $\alpha_1(\cdot)$, $\alpha_2(\cdot)$ such that stage cost $\alpha_1(\|\mathbf{x}_i\|) \leq \mathcal{L}_i(\mathbf{x}_i, \mathbf{u}_i) \leq \alpha_2(\|\mathbf{x}_i\|), \forall (\mathbf{x}_i, \mathbf{u}_i) \in \mathcal{X}_i \times \mathcal{U}_i$.

The terminal cost $E_i(\cdot)$ and terminal constraint (2.2.22d) are time-invariant and are used to enforce nominal stability [Chen and Allgöwer, 1998]. They might or might not be available.

Due to the fact that we take only tracking MPC into account, the stage cost penalizes the errors between the actual and reference profiles. A quadratic form for $\mathcal{L}_i(\cdot)$ is commonly used as follows:

$$\mathcal{L}_i(\mathbf{x}_i(\tau), \mathbf{u}_i(\tau)) = \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i}, \tag{2.2.24}$$

and the terminal cost is defined as:

$$E(\bar{\mathbf{x}}_i(t_k + T_p)) = \|\bar{\mathbf{x}}_i(t_k + T_p) - \mathbf{x}_{i,ref}\|^2_{\mathbf{P}_i}, \tag{2.2.25}$$

[8] A continuous function $\alpha : [0, \infty) \to [0, \infty)$ is a class $\mathcal{K}$ function, if it is strictly increasing and $\alpha(0) = 0$.

where $\mathbf{Q}_i$, $\mathbf{R}_i$ and $\mathbf{P}_i$ are (semi)-positive definite weighting matrices of appropriate dimensions.

It is worth noting that the control input variations and its bounds can also be penalized in the cost, i.e., formulation (2.2.24) may be rewritten as follows:

$$\mathcal{L}_i(\mathbf{x}_i, \mathbf{u}_i) = \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\dot{\bar{\mathbf{u}}}_i(\tau)\|^2_{\Delta\mathbf{R}_i}, \tag{2.2.26}$$

where $\Delta\mathbf{R}_i$ is (semi)-positive definite weighting matrix of appropriate dimensions.

Employing the control input variations is useful as it penalizes variations and lead thus to a smoother state.

In order to predict precisely the behavior of the individual agent, its full state information has to be either measured or estimated, and then the state measurement enters the dynamical system of each agent through the initial condition (2.2.22b).

Hence, the solution of Problem 2.2.7 at sampling time t_k yields the optimal open-loop control signal $\bar{\mathbf{u}}_i^*(\tau; \mathbf{x}_i(t_k))$, computed over $\tau \in [t_k, t_k + T_p]$ for a measured state $\mathbf{x}_i(t_k)$. Then, this control input is applied to the system until the next sampling instant t_{k+1}:

$$u_i(\tau; t_k) = \bar{\mathbf{u}}_i^*(\tau; \mathbf{x}_i(t_k)),\ \tau \in [t_k, t_{k+1}). \tag{2.2.27}$$

This process is repeated at the next sampling instant t_{k+1} for solving a new finite horizon optimal control problem. The control input $u(\tau; \mathbf{x}_i(t_k))$ represents the feedback since it is recalculated at each sampling instant using new state information.

Consequently, optimal motion planning for multi-agent dynamical systems is the result of applying MPC to deal with obstacles. The particular implementation can be done either through the potential field implementation (in Section 2.2.1) or via mixed-integer programming (in Section 2.2.2).

Problem 2.2.8 (Potential field-based tracking MPC)**.** *As defined in* (2.2.10)*, potential field is a combination of repulsive potential and attractive potential functions, in which the repulsive potential* (2.2.12) *constructed from polytopic region* (2.1.4) *has a high value in the interior and a low value sufficiently far away from the obstacle, converges to the attractive potential, describing the goal configuration at a global energy minimum presented by quadratic function* (2.2.11)*. Finite prediction horizon strategies reformulate the potential field approach as in Problem 2.2.7 in order to obtain the collision-free path of the MAS through minimizing a given cost per stage* $\mathcal{L}_i(\cdot)$ (2.2.24) *which is redefined over a finite prediction horizon as follows:*

$$\begin{aligned}\mathcal{L}_i^{\mathrm{PF}}(\mathbf{x}_i, \mathbf{u}_i) &= S(\bar{\mathbf{x}}_i(\tau)) + \|\bar{\mathbf{u}}_i(\tau)\|^2_{R_i} \\ &= \sum_{\ell=1}^{N_{\mathrm{obs}}} \Phi_\ell(\bar{\mathbf{x}}_i(\tau)) + \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i},\end{aligned} \tag{2.2.28}$$

where $\Phi_\ell(\cdot)$ is the repulsive potential defined in (2.2.12). Note that the second term in (2.2.28) is the attractive potential showing the lowest value at the desired target of the potential field as can be seen in Fig. 2.2.4b of Example 2.2.4.

Therefore, the collision avoidance constraints are considered as weakened constraints, or so-called implicit (or soft) constraints by imposing a repulsive potential field constructed from the convex sets appearing in the problem. Since the cost per stage $\mathcal{L}_i^{PF}(\cdot)$ must be minimized such

that its value would be the smallest, the high value regions are inside like the repulsive potentials have to be penalized. Therefore, feasible paths for the MAS are generated which do not collide with the obstacles. CASADI framework [Andersson et al., 2019] with solver IPOPT [Wächter and Biegler, 2006] and ACADO toolkit [Houska et al., 2011] with solver qpOASES [Ferreau et al., 2014] can deal with the non-convex problem of nonlinear programming and will be used hereafter.

Mixed integer nonlinear programming (MINLP) is not easy to implement since it combines the nonlinearity of the dynamics/constraints with the use of binary variables [Jeroslow, 1973]. Hence, in this manuscript, we simplify the problem by considering a relaxed MILP (Mixed integer linear programming) formulation (i.e., MIP formulation). We will employ the CPLEX solver through the Yalmip toolbox [Löfberg, 2004] to deal with the MILP formulation in the MPC scheme.

Problem 2.2.9 (Mixed integer programming-based tracking MPC)**.** *Assuming that MAS operates in a workspace $\mathcal{F}$, comprising a set of bounded convex regions representing static obstacles as* (2.2.8)*, at every sampling instant, t_k, the predicted position state of MAS has to lie in the exterior of prohibited regions through the constraints expressed in MILP form in* (2.2.20) *and* (2.2.21)*. As a consequence, this approach is similar with Problem 2.2.7 with the MILP constraints as presented in* (2.2.20) *and* (2.2.21) *are taken into account.*

As a consequence, to deal with the non-convex problem of motion planning for MAS based on tracking optimization-based control, we presented two main approaches which handle the non-convex constraints in two ways, implicit (i,e., soft) constraints penalized through a potential field formulation as in Problem 2.2.8 and explicit (i.e., hard) constraints under MILP form as in Problem 2.2.9. Of course, each approach has its own advantages and disadvantages which have to be considered before a decision. In Section 2.4 comparisons as well as evaluations will be carried out to validate the selection done in this manuscript (i.e., the preference towards potential field methods).

2.3 Control barrier function-based constraints description

Besides the approaches mentioned earlier, collision avoidance constraints may also be implemented through a barrier certificate, i.e., control barrier function (CBF). CBF was proposed recently to certify forward invariance of a set using barrier function regardless of the details of the reachable set computation [Wieland and Allgöwer, 2007]. Therefore, this approach has been investigated and applied for a variety of domains, e.g., [Ames et al., 2014] presents the combination of CBF and control Lyapunov functions in the context of a quadratic program for adaptive cruise control and lane keeping control, whereas [Wang et al., 2017a] considers collision avoidance of multi-mobile robot systems. The fundamental idea is to define a set of safe states and then use the CBF to formally guarantee that if the system starts in the safe set, it still stays in the safe set, this property is the so-called forward invariance of the set.

2.3.1 Control barrier function

We start by introducing the notion of *safe set*. We first take dynamical systems of the i^{th} agent as in (2.2.1) into account, the objective is to find conditions such that the states of this system always lie in a defined safe set $\mathcal{C} \in \mathbb{R}^n$, i.e., the set in which the agent's states should stay, and a

controller is then generated which ensures the invariance property of the set $\mathcal{C}$, i.e., if $\mathbf{x}_i(0) \in \mathcal{C}$, then $\mathbf{x}_i(t) \in \mathcal{C}, \forall t \geq 0$. Let us define the set $\mathcal{C}$ as follows:

$$\mathcal{C} = \{\mathbf{x}_i \in \mathbb{R}^n : h(\mathbf{x}_i) \geq 0\}, \tag{2.3.1a}$$

$$\partial\mathcal{C} = \{\mathbf{x}_i \in \mathbb{R}^n : h(\mathbf{x}_i) = 0\}, \tag{2.3.1b}$$

$$Int(\mathcal{C}) = \{\mathbf{x}_i \in \mathbb{R}^n : h(\mathbf{x}_i) > 0\}, \tag{2.3.1c}$$

where a smooth function $h(\cdot) : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function, $\partial\mathcal{C}$ defines the boundary of set $\mathcal{C}$ and $Int(\mathcal{C})$ stands for the strict interior of the set $\mathcal{C}$.

From the safe set $\mathcal{C}$ is defined in (2.3.1), the invariance of $\mathcal{C}$ can be established by using a barrier function $B : \mathcal{C} \to \mathbb{R}$ (see [Prajna et al., 2007]). Motivated from the barrier function in numerical optimization in [Boyd and Vandenberghe, 2004], the barrier function can be defined in the form of the logarithmic barrier function:

$$B(\mathbf{x}_i) = -\log\left(\frac{h(\mathbf{x}_i)}{1+h(\mathbf{x}_i)}\right). \tag{2.3.2}$$

Alternatively, it may take the form of the inverse barrier function:

$$B(\mathbf{x}_i) = \frac{1}{h(\mathbf{x}_i)}. \tag{2.3.3}$$

Both forms of barrier function, $B(\mathbf{x}_i)$, must satisfy the following properties:

$$\underset{\mathbf{x}_i \in Int(\mathcal{C})}{inf} B(\mathbf{x}_i) > 0, \qquad \underset{\mathbf{x}_i \to \partial\mathcal{C}}{lim} B(\mathbf{x}_i) = \infty. \tag{2.3.4}$$

Example 2.3.1. Consider the bounded constraint, $-1 \leq x \leq 2$, one is able to obtain the safe set $\mathcal{C}$ (2.3.1) with $h_1(x) = x + 1 \geq 0$ and $h_2(x) = 2 - x \geq 0$, $B(x) = B_1(x) + B_2(x)$, where the barrier function candidate is in the form of the inverse barrier function (2.3.3), satisfying the properties (2.3.4) as can be seen in Fig 2.3.1.

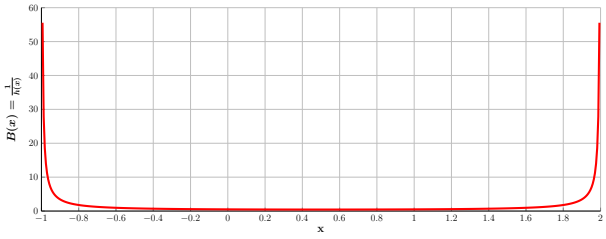


Figure 2.3.1: Barrier function for $-1 \leq x \leq 2$.

The authors in [Prajna et al., 2007], [Tee et al., 2009] have presented the condition $\dot{B}(\mathbf{x}_i) \leq 0$ to ensure that $Int(\mathcal{C})$ is forward invariant, however, this is excessively harsh due to the fact that it requires all sublevel sets of $\mathcal{C}$ to be also invariant, particularly, it will not permit a solution to leave a sublevel set even if it still remains in $Int(\mathcal{C})$. As a consequence, [Ames et al., 2014] proposed an enhancement by proposing the following relaxation:

$$\dot{B}(\mathbf{x}_i) \leq \frac{\kappa}{B(\mathbf{x}_i)}, \tag{2.3.5}$$

with $\kappa > 0$ and $B(\cdot)$ considered as a barrier function candidate from (2.3.2) or (2.3.3). The constraint (2.3.5) allows for $B(\cdot)$ to increase rapidly when $\mathbf{x}_i$ is far away from the boundary, and approaches zero as $\mathbf{x}_i$ approaches the boundary of $\mathcal{C}$, $\partial\mathcal{C}$, defined as in (2.3.1b).

For condition (2.3.5) to be feasible, the states of (2.2.1) have to be guaranteed to stay in $Int(\mathcal{C})$. We can verify this through differentiating the logarithmic barrier function (2.3.2) along solutions of (2.2.1):

$$\dot{B}(\mathbf{x}_i) = \frac{\partial B(\mathbf{x}_i)}{\partial h(\mathbf{x}_i)}\dot{h}(\mathbf{x}_i) = \frac{\dot{h}(\mathbf{x}_i)}{h(\mathbf{x}_i) + h^2(\mathbf{x}_i)}, \tag{2.3.6}$$

Hence, the condition (2.3.5) implies that $\dot{h}(\cdot)$ is bounded by:

$$\dot{h}(\mathbf{x}_i) \geq \frac{\gamma(h(\mathbf{x}_i) + h^2(\mathbf{x}_i))}{\log\left(\frac{h(\mathbf{x}_i)}{1+h(\mathbf{x}_i)}\right)}. \tag{2.3.7}$$

Using the Comparison Lemma [Khalil and Grizzle, 2002] gives

$$h(\mathbf{x}_i(t, \mathbf{x}_{i0})) \geq \frac{1}{-1 + \exp\left(\sqrt{2\gamma t + \log\left(\frac{h(\mathbf{x}_{i0})+1}{h(\mathbf{x}_{i0})}\right)}\right)}. \tag{2.3.8}$$

With the the inverse barrier function (2.3.3), we are also able to differentiate along solutions of (2.2.1):

$$\dot{B}(\mathbf{x}_i) = -\frac{\partial B(\mathbf{x}_i)}{\partial h(\mathbf{x}_i)}\dot{h}(\mathbf{x}_i) = \frac{\dot{h}(\mathbf{x}_i)}{h^2(\mathbf{x}_i)}. \tag{2.3.9}$$

Applying (2.3.5) means that $\dot{h}(\cdot)$ is bounded by:

$$\dot{h}(\mathbf{x}_i) \geq \frac{\gamma h^2(\mathbf{x}_i)}{\left(\frac{1}{h(\mathbf{x}_i)}\right)}. \tag{2.3.10}$$

The Comparison Lemma [Khalil and Grizzle, 2002] leads to

$$h(\mathbf{x}_i(t, \mathbf{x}_{i0})) \geq \frac{1}{\sqrt{2\gamma t + \frac{1}{h^2(\mathbf{x}_{i0})}}}. \tag{2.3.11}$$

Consequently, (2.3.8) and (2.3.11) show that, if $h(\mathbf{x}_{i0}) > 0$, i.e., $x_{i0} \in Int(\mathcal{C})$, then checking constraint (2.3.5) ensures that $h(\mathbf{x}_i(t, \mathbf{x}_{i0})) > 0$, $\forall t \geq 0$, which means that $\mathbf{x}_i(t, \mathbf{x}_{i0}) \in Int(\mathcal{C})$, $\forall t \geq 0$.

Definition 2.3.2 (Control Barrier function [Ames et al., 2014])**.** *For the model system as in* (2.2.1)*, let* $\mathcal{C} \subset \mathbb{R}^n$ *be defined by* (2.3.1) *for a continuously differentiable function* $h : \mathbb{R}^n \to \mathbb{R}$*, then a function* $B : \mathcal{C} \to \mathbb{R}$ *is a control barrier function (CBF) if if there exist class* $\mathcal{K}$ *functions,* α_1*,* α_2*, and* $\kappa > 0$ *such that,*

$$\frac{1}{\alpha_1(h(\mathbf{x}_i))} \leq B(\mathbf{x}_i) \leq \frac{1}{\alpha_2(h(\mathbf{x}_i))}, \tag{2.3.12a}$$

$$\dot{B}(\mathbf{x}_i) - \frac{\kappa}{B(\mathbf{x}_i)} \leq 0. \tag{2.3.12b}$$

Remark 2.3.3. The bound of $B(\cdot)$ in (2.3.12a) signifies that along with solutions of (2.2.1), $B(\cdot)$ basically acts like $\frac{1}{\alpha(h(\mathbf{x}_i))}$ for some class $\mathcal{K}$ function α with

$$\inf_{\mathbf{x}_i \in Int(\mathcal{C})} \frac{1}{\alpha(h(\mathbf{x}_i))} > 0, \qquad \lim_{\mathbf{x}_i \to \partial\mathcal{C}} \frac{1}{\alpha(h(\mathbf{x}_i))} = \infty. \tag{2.3.13}$$

Remark 2.3.4. (2.3.12a) and (2.3.12b) ensure that if $\mathbf{u}_i(\mathbf{x}_i) \in \left\{\mathbf{u}_i \in \mathcal{U}_i : \ \dot{B}(\mathbf{x}_i) - \frac{\kappa}{B(\mathbf{x}_i)} \leq 0\right\}$, then the system (2.2.1) renders the set $\mathcal{C}$ forward invariant.

It is worth noting that the CBF, as presented, only works in the case when $h(\cdot)$ has relative degree $r_d = 1$ as conclusion of [Borrmann et al., 2015].

2.3.2 Potential field for anti-collision constrains using the control barrier function

This subsection presents the method for constructing the control barrier function constraint (in the case $h(\cdot)$ has a relative degree $r_d = 1$ (i.e., a pairwise safety constraint between two agents can obtain from the first derivative of $h(\cdot)$)) in order to prevent collision among the agents while operating. A safe set $C_{ij}(\cdot)$ is defined over the state space of the MAS, which yields control barrier functions which are subsequently used to guarantee that the agents still stay in this set for all time.

We particularize the generic dynamical system defined in (2.2.1) by assuming the case of double-integrator dynamics where the state of the i^{th} agent is defined by position, $\mathbf{p}_i = \begin{bmatrix} \mathbf{x}_i & \mathbf{y}_i \end{bmatrix}^\top \in \mathbb{R}^2$, and velocity, $\mathbf{v}_i = \begin{bmatrix} \mathbf{v}_{xi} & \mathbf{v}_{yi} \end{bmatrix}^\top \in \mathbb{R}^2$, and the control input, $\mathbf{u}_i = \begin{bmatrix} \mathbf{u}_{xi} & \mathbf{u}_{yi} \end{bmatrix}^\top \in \mathbb{R}^2$, defined by the acceleration.

Assuming any two agents i and j, their relative position and relative velocity are given by $\Delta\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ and $\Delta\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$. As also stated in [Borrmann et al., 2015], we are interested only in the relative velocity between the agents denoted as:

$$\Delta\bar{v} = \|\Delta\dot{\mathbf{p}}_{ij}\| = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij}, \tag{2.3.14}$$

that causes collision. Consequently, $\Delta\bar{v}$ can be controlled to avoid imminent collisions if the maximum relative braking force is applied. Any two agents could always keep a safety distance D_s due to the following pairwise safety constraint:

$$\|\Delta\mathbf{p}_{ij}\| - \frac{\Delta\bar{v}^2}{c_{ij}\Delta a_{max}} \geq D_s, \ \forall i \neq j, \tag{2.3.15}$$

which can be rewritten as:

$$-\frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} \leq \sqrt{c_{ij}\Delta a_{max}(\|\mathbf{p}_{ij}\| - D_s)}, \forall i \neq j, \tag{2.3.16}$$

where $\Delta a_{max} = 2a_{max}$ is the maximum relative braking acceleration and $c_{ij} \geq 0$ is the collision avoidance coefficient, which scales the maximum relative braking acceleration. Note that the constraint (2.3.16) is activated when the two agents are moving closer towards each other ($\frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} < 0$) and inactivated when the two agents are moving away from each other ($\frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} \geq 0$).

We can now define the pairwise safe set $\mathcal{C}_{ij}$ as:

$$\mathcal{C}_{ij} = \{(\mathbf{p}_i, \mathbf{v}_i) \in \mathbb{R}^4 | h_{ij}(\Delta\mathbf{p}_{ij}, \Delta\mathbf{v}_{ij}) \geq 0\}, \ \forall i \neq j, \tag{2.3.17}$$

where h_{ij} is derived from (2.3.16):

$$h_{ij} = \frac{\Delta \mathbf{p}_{ij}^T}{\|\Delta \mathbf{p}_{ij}\|} \Delta \mathbf{v}_{ij} + \sqrt{c_{ij} \Delta a_{max} (\|\Delta \mathbf{p}_{ij}\| - D_s)}. \tag{2.3.18}$$

Taking into account (2.3.17) and (2.3.18), the inverse control barrier function[9], $B_{ij}(\Delta \mathbf{p}_{ij}, \Delta \mathbf{v}_{ij})$, which satisfies the pairwise safety constraint (2.3.16) is constructed as follows[10]:

$$B_{ij}(\Delta \mathbf{p}_{ij}, \Delta \mathbf{v}_{ij}) = \frac{1}{h_{ij}(\Delta \mathbf{p}_{ij}, \Delta \mathbf{v}_{ij})}, \ \forall i \neq j. \tag{2.3.19}$$

Furthermore, by combining the condition (2.3.5) to ensure that $Int(\mathcal{C}_{ij})$ is forward invariant, (2.3.18) and (2.3.19), a linear constraint on the control variable $\mathbf{u}_{ij}$ can be formulated:

$$-\Delta \mathbf{p}_{ij}^T \Delta \mathbf{u}_{ij} - \frac{\gamma}{B_{ij}} h_{ij}^2 \|\Delta \mathbf{p}_{ij}\| + \frac{(\Delta \mathbf{v}_{ij}^T \Delta \mathbf{p}_{ij})^2}{\|\Delta \mathbf{p}_{ij}\|^2} - \|\Delta \mathbf{v}_{ij}\|^2 - \frac{c_{ij} \Delta a_{max} \Delta \mathbf{v}_{ij}^T \Delta \mathbf{p}_{ij}}{2\sqrt{c_{ij} \Delta a_{max} (\|\Delta \mathbf{p}_{ij}\| - D_s)}} \leq 0, \ \forall i \neq j. \tag{2.3.20}$$

(2.3.20) is the CBF constraint for guaranteeing collision avoidance for each pair of agents in MAS and it will be employed as an explicit constraint in the potential field-based MPC (Problem 2.2.8) or MIP-based MPC (Problem 2.2.9).

2.4 Simulation results for collision avoidance using repulsive potentials

This section first presents some comparison[11] of the two approaches, PF and MIP-based optimization control for motion planning of MAS. The approaches discussed here are paramount for the rest of the manuscript and will be used and enhanced throughout it. Lastly, potential field-based NMPC is combined with CBF to ensure collision and obstacle avoidance.

Consider a multi-agent system with double-integrator dynamics, with N_a agents

$$\begin{bmatrix} \dot{\mathbf{p}}_i \\ \dot{\mathbf{v}}_i \end{bmatrix} = \mathbf{A}_i \begin{bmatrix} \mathbf{p}_i \\ \mathbf{v}_i \end{bmatrix} + \mathbf{B}_i \mathbf{u}_i, \tag{2.4.1}$$

where

$$\mathbf{A}_i = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\zeta_i}{m_i} & 0 \\ 0 & 0 & 0 & -\frac{\zeta_i}{m_i} \end{bmatrix}, \ \mathbf{B}_i = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m_i} & 0 \\ 0 & \frac{1}{m_i} \end{bmatrix}, \tag{2.4.2}$$

denote the simplified dynamics of unmanned surface vessels, as used in [Prodan, 2012], where $\mathbf{p}_i = [x_i \ y_i]^\top \in \mathbb{R}^2$, $\mathbf{v}_i = [v_{x_1} \ v_{x_2}]^\top \in \mathbb{R}^2$ and $\mathbf{u}_i = [u_{x_i} \ u_{y_i}]^\top \in \mathbb{R}^2$ stand for positions, velocities and and control inputs $u_{x_1} \ u_{x_2}$ represents accelerations. The parameters of $\zeta_i = 3$, $m = 60$ are the damping factor and the mass[12].

[9] The choice of the inverse control barrier function is not restrictive, one may use other types of control barriers, for example a logarithmic control barrier function as defined in (2.3.2).

[10] Note that for any initial distance between two agents i and j, i.e., $\|\Delta \mathbf{p}_{ij}\| \leq D_s$, $i \neq j$, h_{ij} can not be defined, leading to B_{ij} being undefined. As a consequence, the initial distance between any two agents must be greater than D_s.

[11] Note that both methods are verified with the same scenario.

[12] It is worth noting that the real dynamical model of the vessel is trivially underactuated, which comes from a lower number of actuators than outputs to be steered. In this chapter, we use only simplified linear dynamics to facilitate illustrating the problems being presented. In future chapters, a realistic ship model will be applied.

2.4.1 Potential field and Mixed-integer programming approach

To demonstrate the effectiveness of this approach, a comparison of MIP and PF is implemented in Matlab 2015a on a computer with the following configuration: Intel Core i7-4790CPU, 3.60GHz, 8GB RAM.

This comparison is based on three criteria: i) capability of ensuring obstacle avoidance; ii) feasibility (i.e., returning a solution for each step of the simulation) and iii) execution time.

It is worth noting that in this subsection, we only concentrate on the obstacle avoidance issue when applying the potential field and MIP approaches.

Scenario 1: one static obstacle

Consider one agent (i.e., $N_a = 1$) in two spatial dimensions described by the dynamics in (2.4.1), operating in workspace $\mathcal{F} \in \mathbb{R}^2$ in the presence of only one obstacle defined as in (2.1.4). Making use of the polytopic region as described in Example 2.1.5 accounts for the fixed obstacle. We assume that the agent starts from initial point $[-18, -18]^\top$ to a destination target $p_{ref} = [-6,\ 6]^\top$ which is the interior of the obstacle. The working space is a box constraint $\{\mathbf{p}_i \in \mathbb{R}^2 | -100 \leq \mathbf{p}_i \leq 100\}$ and the control input's constraint for the agent is $\{\mathbf{u}_i \in \mathbb{R}^2 | -20 \leq \mathbf{u}_i \leq 20\}$.

- Concerning the *potential field-based MPC*, we first obtain the sum function (2.1.6) from a given polytopic region representing the static obstacle. Then, repulsive potential function in favor of one obstacle is derived from (2.2.12) with the parameter of strength $c_1 = 0.5$ and effect range $c_2 = 0.01$, one can also derive the attractive potential function as presented in (2.2.11) with desired destination of agent, p_{ref}. Finally, we solve Problem 2.2.8 to accomplish the collision-free path for agent.
- With regard to *MIP optimization-based control approach*, the non-convex working space is described as in Example 2.2.6. Then, the mixed integer linear constraint is constructed as in (2.2.20) and (2.2.21):

$$-\begin{bmatrix} -0.2691 & -0.2018 \\ 0.1871 & -0.0234 \\ 0.1156 & 0.1445 \\ -0.1046 & 0.0262 \end{bmatrix} \mathbf{p}_i \leq -\begin{bmatrix} 0.9417 \\ -0.9821 \\ 0.9827 \\ 0.9942 \end{bmatrix} + M \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix},$$
$$z_1 + z_2 + z_3 + z_4 \leq 3,$$

  where M is chosen sufficiently large, 10^4 and $(z_1, \ldots, z_4) \in \{0, 1\}^4$.
  With all these elements in place, we solve Problem 2.2.9 to obtain the obstacle-avoiding path for the agent.

The prediction horizon for both approaches are $T_p = 10s$, time step $T_e = 1s$ and the weighting matrices of both methods are chosen similarly:

$$\mathbf{Q}_i = 0.1 \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{P}_i = 0.5 \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Delta\mathbf{R}_i = 0.1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Fig. 2.4.1 illustrates the actual motion of the agent through both approaches. None of the trajectories is able to reach the desired target (which is unreachable since it lies inside the obstacle). However, the MIP implementation leads to positions which are closer to the target point than the APF approach. They thus stay on the hyperplane of the fixed obstacle (the red one). This is to be expected: the potential field grows in the vicinity of the obstacle (as governed by the effect range parameter), and the agent's trajectory naturally avoids the "hill" regions of the field. From a practical point of view, the behavior exhibited by the APF methods is more than acceptable: the trajectory gets close to the obstacle but is still sufficiently far away to avoid an accidental collision. In particular, if the trajectory would reach the boundary, the APF would have failed since inside the obstacle the field has a constant value and, thus, the trajectory could travel inside without any additional penalty.

The control inputs of the two approaches are also depicted in Fig. 2.4.2, It becomes evident that the potential field method shows less oscillation and more smoothness than the MIP-based method in the control input.

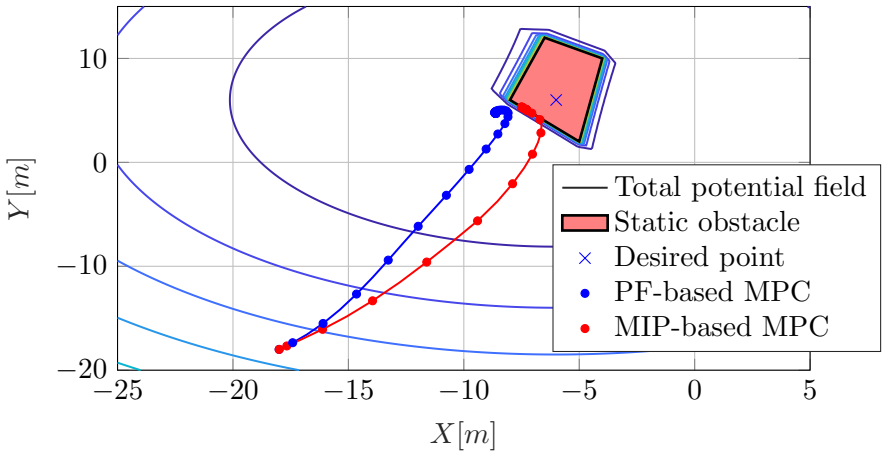


Figure 2.4.1: Motion planning of agent with PF-based MPC and MIP-based MPC.

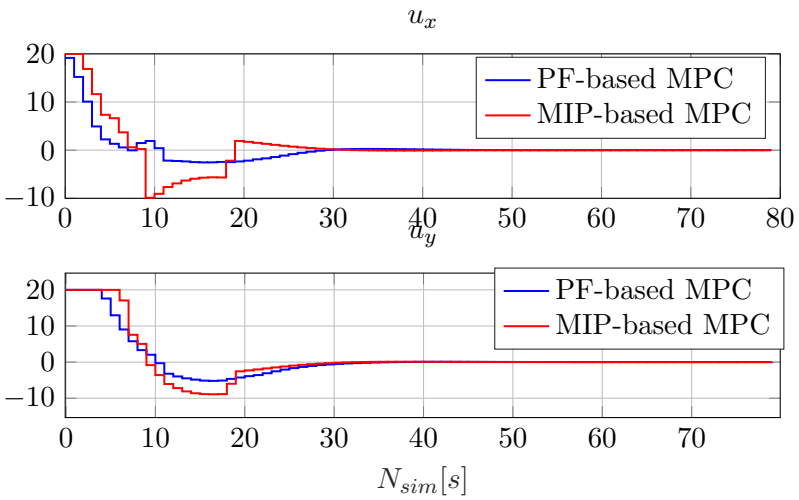


Figure 2.4.2: Comparison of control inputs of agent with PF-based MPC and MIP-based MPC.

Scenario 2: two static obstacles

Table 2.4.1: Performance criteria for the motion planning of agent implemented by two approaches in Scenario 1.

	Potential field approach	**MIP approach**
Prediction horizon $[s]$	10	10
Sampling time, T_e [s]	1	1
Simulation number	80	80
CPU time $[s]$	20.43	20.39
Length of trajectory $[m]$	25.9	27.2
Feasibility	✓	✓
Obstacle avoidance	✓	✓

Consider the Example 2.2.5 again, and let the position component of the agent (2.4.1) in the working space $\mathcal{F} \in \mathbb{R}^2$ be constrained by the two triangular obstacles, illustrated in Fig. 2.2.4a. We assume various initial points (50 different initial points) and a common destination target (at coordinates $p_{ref} = [0,\ 16]^\top$). We take the initial points as close as possible to the obstacles in order to verify the effectiveness of the obstacle avoidance schemes.

- *Potential field optimization-based control approach:* we first obtain the sum function (2.1.6) from a given polytopic region representing the static obstacle defined by (2.1.4). Then, repulsive potential function for each obstacle is derived from (2.2.12) with the parameters of strength $c_{11} = c_{12} = 100$ and effect range $c_{21} = c_{22} = 0.5$, the attractive potential function is taken into account by (2.2.11) with desired the destination p_{ref}. Finally, solving Problem 2.2.8 for obtaining the collision-free path for agent.

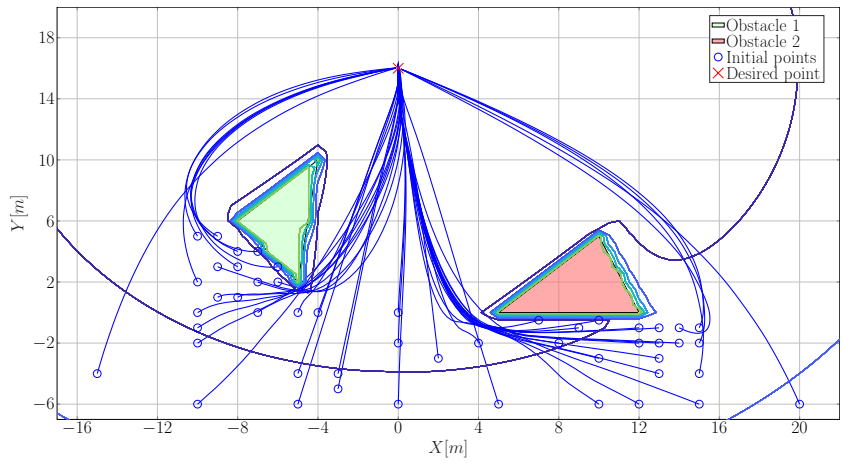


Figure 2.4.3: Agent trajectories generated by potential field-based MPC.

Similar to Scenario 1 with PF-based MPC, Scenario 2 shows an outstanding result where 50 trajectories avoid the two obstacles; even when the initial points stay very close to the obstacle, as can be seen in Fig. 2.4.3.

- *MPC with MIP-based obstacle avoidance:* the non-convex workspace of the agent, $\mathcal{F} \in \mathbb{R}^2$ in the presence of two triangular obstacles described as in Fig. 2.2.4a. Then, the mixed integer linear constraints for two static obstacles are constructed as in (2.2.20) and (2.2.21).

For the 1^{st} static obstacle (the green one):

$$-\begin{bmatrix} 0.1871 & -0.0234 \\ -0.0711 & 0.0711 \\ -0.2691 & -0.2018 \end{bmatrix} \mathbf{p}_i \leq -\begin{bmatrix} -0.9821 \\ 0.9949 \\ 0.9417 \end{bmatrix} + M \begin{bmatrix} z_1^1 \\ z_2^1 \\ z_3^1 \end{bmatrix},$$
$$z_1^1 + z_2^1 + z_3^1 \leq 2.$$

For the 2^{nd} static obstacle (the red one):

$$-\begin{bmatrix} 0 & -1.0000 \\ 0.0830 & 0.0332 \\ -0.1925 & 0.19258 \end{bmatrix} \mathbf{p}_i \leq -\begin{bmatrix} 0 \\ 0.9960 \\ -0.9623 \end{bmatrix} + M \begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix},$$
$$z_1^2 + z_2^2 + z_3^2 \leq 2.$$

where M for both fixed obstacles is chosen sufficiently large, 10^4, and z_k^1, z_k^2 $(k = 1, 2, 3)$ are the auxiliary binary variables representing the 1^{st} and 2^{nd} obstacle.

Ultimately, solving Problem 2.2.9 with these explicit constraints leads to the agent's path.

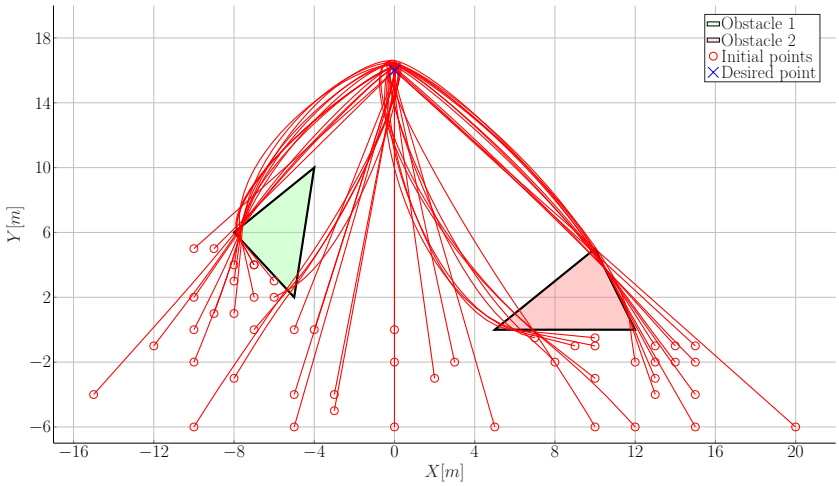


Figure 2.4.4: Agent trajectories generated by MIP-based MPC.

The results of this method are not as good as the potential field approach. Although the execution time of two methods is approximately equal, almost all the trajectories generated via mixed-integer representation are not able to avoid the fixed obstacles at the corners as can be observed in Fig. 2.4.4. The ratio of success of collision-free trajectories generated is 28%.

The obstacle avoidance of MIP optimization-based control approach can be enhanced by decreasing the sampling time $T_e = 0.1$. However, this leads to a remarkable increase of simulation number (for reaching the target point) and computation time of solver (3 times as opposed to potential field method). Moreover, trajectories exhibit a chaotic behavior while converging towards the desired point (the red one). In particular, there are six unfeasible trajectories (the black ones) (12%) due to the sudden termination of the branching (MIP technique based on a branch-and-bound algorithm). It is the main shortcoming of this technique, which was mentioned at the beginning of this Chapter and in Chapter 1. These illustrations are depicted in Fig. 2.4.5.

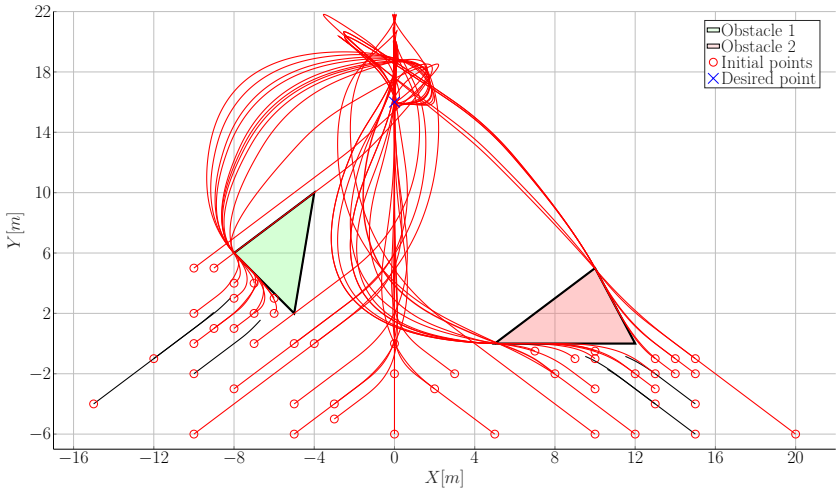


Figure 2.4.5: Agent trajectories generated by MIP-based MPC.

Table 2.4.2: Performance criteria for the motion planning of agent implemented by two approaches in Scenario 2.

50 various initial points	**PF approach** (Fig. 2.4.3)	**MIP approach** (Fig. 2.4.4)	**MIP approach** (Fig. 2.4.5)
Prediction horizon $[s]$	20	20	20
Sampling time, T_e [s]	1	1	0.1
Simulation number	100	100	250
Average CPU time $[s]$	25.43	25.45	75.42
Feasibility	100%	100%	88%
Obstacle avoidance	100%	28%	100%

Performance criteria for the motion planning of agent of both approaches are compared as presented in detail in the Table 2.4.2.

The comparisons between the two approaches, MIP-based MPC, and potential field-based MPC show us the better result of the latter one in connection with the feasibility, ability of collision avoidance as well as the computation time. As a result, the potential field optimization-based control approach is taken into account as the primary method for motion planning of the multi-agent dynamical systems. Note that, at a first glance MIP is the more tempting approach: the resulting mixed-integer model is "nicer" as it appears to be pseudo-linear in its constraints and, writing explicitly the constraints give the impression that the problem is well-behaved. Still, the numerical issues which afflict MIP formulations lead to results which are often inferior to other approaches (as proved to be the case here, when compared with the APF method).

2.4.2 Potential field-based MPC approach using the control barrier constraint

Since the capability of obstacle avoidance and the feasibility of motion planning for MAS with the PF approach are better and more realistic than the MIP-based MPC, the PF approach is combined with CBF to ensure the collision-free path for MAS in the operating space.

Consider two agents (i.e., $N_a = 2$) described by the dynamics in (2.4.1), which move in opposite directions in a workspace $\mathcal{F} \in \mathbb{R}^2$ in the presence of two triangular obstacles as illustrated in Fig. 2.2.4a. We consider two agents in a decentralized control framework (i.e., each agent owns a cost function as presented in Problem 2.2.8. An explicit CBF constraint is established as in (2.3.20) and taken into account by Problem 2.2.8 (potential field-based tracking MPC) for ensuring a safe distance between the two agents. The weighting matrices of each cost function, the prediction horizon T_p and sampling time T_e are aforementioned at the beginning of this section, the parameter κ as in (2.3.5) is chosen equal to 1, and the collision avoidance coefficient $c_{12} = c_{21} = 1$.

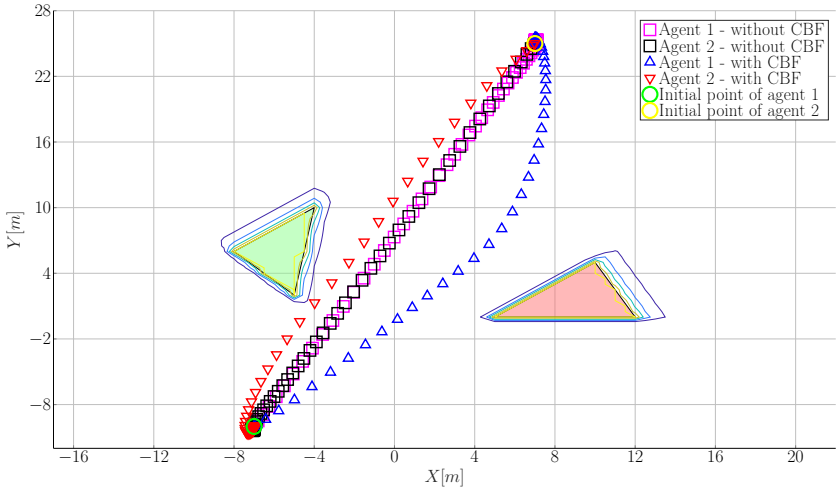


Figure 2.4.6: Motion planning of two agents generated by the PF approach with and without CBF.

Due to the scenario's nature, the two agents (the violet and black one, account for agent 1 and 2 in case of without using CBF) collided (they met midway) as illustrated in Fig. 2.4.6 and Fig. 2.4.7. In the presence of explicit CBF constraint, the safety distance $D_s = 5[m]$ between them is ensured as we are also able to observe their trajectories in Fig. 2.4.6 (the blue and the red colors represent agent 1 and 2 in case of using CBF). As a consequence, MAS can implement both obstacle and collision avoidance with the combined algorithm (PF with the additional control barrier function constraint). The control input of the two agents in this scenario is also illustrated in Fig. 2.4.8.

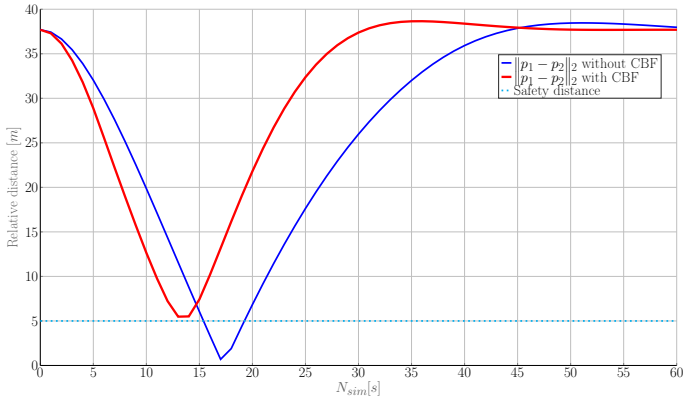


Figure 2.4.7: Relative distance of two agents generated by the PF approach with and without CBF.

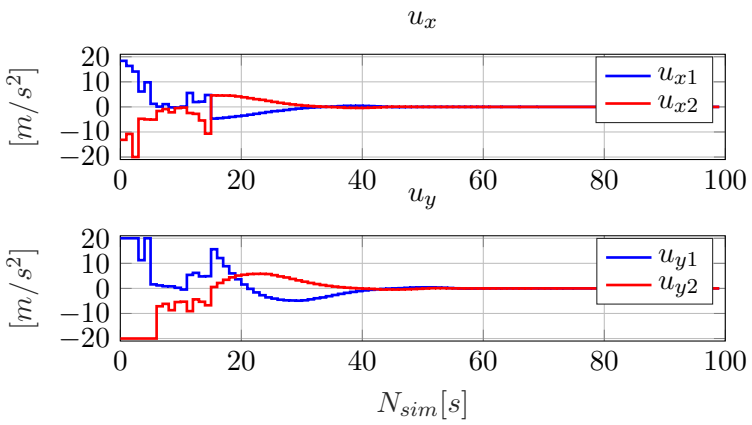


Figure 2.4.8: CBF constraint (2.3.17) of two agents generated by the PF approach with CBF.

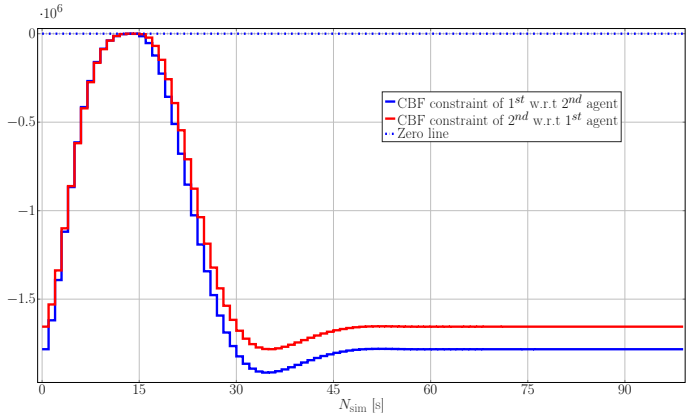


Figure 2.4.9: Control barrier constraints of two agents.

Note that the explicit CBF constraint (2.3.20) for the two agents which are based on the Euclidean distance is fully verified in Fig. 2.4.9. In other words, CBF constraint guarantees that all the states of the agents start in the pairwise safe set $\mathcal{C}_{ij}$ (2.3.17), and still stay in the safe set $\mathcal{C}_{ij}$, i.e., the property of forward invariance is guaranteed.

2.5 Conclusions

In this chapter, we first recalled the polyhedral set notion, which is used to describe the prohibited regions representing the static obstacles where MAS are not allowed to enter. In another interpretation, the workspace in which MAS operates is non-convex (the complement of a union of convex regions). Next, optimization-based control techniques, in particular, (non-)linear model predictive control, were presented for dealing with these constraints. We detail two approaches, each of them treating the obstacle avoidance constraint implementation in a different manner. First, the potential field method considers an implicit description (where the constraint-induced potential is penalized in the cost). Second, the mixed-integer programming method considers an explicit description (where the constraints are modeled with the help of binary variables and integrated into a MIP optimization problem). Finally, simulations are carried for a double-integrator dynamic to show the benefits of using the PF-based MPC approach and justify its subsequent use throughout the manuscript.

In addition, barrier certificates (i.e., control barrier functions) were also presented and taken into account as the explicit constraints which aim at ensuring no collision among agents in the context of potential field-based MPC. Although this combined algorithm has shown its effectiveness in generating the collision-free path for MAS in the variable environment, it also gives out some issues which have to be further investigated.

First, the admissible control space permitted by the safety barrier certificates is reduced. As can be seen in Fig. 2.4.9, it is challenging to perform the desired control actions of the nominal controller. Consequently, if there is an increase in the number of agents, we may arrive at a deadlock in some particular configurations. Second, the approach, as presented, only applies for the 1^{st} order relative degree of the smooth function $h(\cdot)$. As a result, it is difficult to find out a safe set for ensuring forward invariance in the case of the complex dynamical system of MAS, e.g., have nonlinearities in their dynamics. Although exponential control barrier functions as presented in [Nguyen and Sreenath, 2016, Ames et al., 2017] can deal with the higher relative degrees, CBF approach considers the agent pointwise, which is far from the conditions encountered in practical applications. Finally, using the PF approach means that we have to study and overcome local minima problems, into which the agents may be trapped when having many repulsive potential fields.

Considering these issues, the future work will first ignore the CBF approach as was aforementioned for collision avoidance. This is acceptable from a practical point of view because each agent has its own specific shape. Second, we will investigate and propose an enhanced method to deal with the local minima. Whether the agents are trapped into the local minima, and if so, the question is posed *how to deal with local minima for motion planning of MAS in potential field-based MPC approach?* For answering this question, Chapter 3 will provide an algorithm to deal with this issue.

Chapter 3

Potential-field constructions for motion planning with collision avoidance

As presented in Chapter 2, particularly in the implementation of the tracking MPC scheme, the total potential field (additively composed from repulsive, accounting for forbidden region avoidance, and attractive, accounting for goal convergence, potential fields) has been used as a penalty term in the cost. This generic potential field-based MPC construction can be applied in a wide range of applications, each of them with its own particularities. Hence, in this chapter, we concentrate on the safe navigation for MAS in a maritime field. This is a challenging research topic with many open problems. We consider thus, that proving the potential field approach in this application, will convincingly validate the approach and show its versatility for applications in other areas.

In recent years, there has been increasing interest in path planning for marine vehicles in both military and civilian applications with the dual aim of attenuating accidents at sea and of increasing efficiency. In particular, for *Unmanned Surface Vehicles* (USVs), it is essential to generate collision-free paths with respect to both fixed obstacles (e.g., islands, shorelines or other ships at anchor) and moving obstacles (e.g., other ships while under operation). Not in the least, the collision avoidance actions for both manned and unmanned surface vessels must respect the rules which have been given by the Convention on the International Regulations for Preventing Collisions at Sea (COLREGS) [Commandant, 1999].

For MAS motion planning in a coastal environment, we may classify some fundamental methods which cope with the collision avoidance problem with static/dynamic obstacles:
i) *Evolutionary algorithms* which exploit their population-based search nature to yield near-optimal solutions through successive iterations. Particle Swarm Optimization and Genetic algorithm are two well-known algorithms that are generally used to solve motion planning for USV at sea such as in [Aghababa et al., 2012], [Hu et al., 2017], [Arzamendia et al., 2019] and [Xiong et al., 2019];
ii) *Roadmap-based methods* are employed to diminish the N-dimensional configuration workspace to a set of one-dimensional paths, which are then enumerated to determine the shortest path for USV, as presented in [Wu et al., 2013], [Niu et al., 2019];
iii) *Randomized sampling algorithms* where the two typical approaches are Probabilistic Roadmaps (PRM) and Rapidly-exploring Randomized Trees (RRT). Their fundamental ideas are based on

taking randomly sampled states to search a collision-free path between a priori given two states (i.e., starting and final point) [Larson et al., 2007], [Cui et al., 2015] and [Chiang and Tapia, 2018];
iv) *Constrained optimization-based control* which is often implemented as (non-)linear model predictive control has been applied to USV in the problem of path following [Oh and Sun, 2010, Liu et al., 2015, McNinch et al., 2008] whereas collision avoidance in a simple external operating space has been considered by [Johansen et al., 2016, Breivik et al., 2017, Sun et al., 2018a].

From a control point of view, to handle the constraints describing obstacle and collision avoidance, as presented in Chapter 2, two approaches were presented: explicit (mixed-integer based) and implicit (potential-field based). Usually, the explicit methods reduce to the use of a MIP implementation where the constraints are explicitly taken into account but at the cost of high computation times since it typically demands searching through a large binary tree. Hence, the problem is NP-hard, which means that, quite probably, the result of the optimization will either be infeasible or mired into numerical issues. In addition, using the nonlinear vehicle dynamics leads to nonlinear mixed-integer nonlinear optimization programs which are substantially more challenging than standard MIP implementations [Abichandani et al., 2015, Ragi and Mittelmann, 2017, Cafieri and Rey, 2017]. The implicit methods usually reduce to the use of various types of potential field-based constructions (presented in detail in Chapter 2). Ideally, a perfect potential function may lead the USVs along globally optimal collision-free paths. Nonetheless, an inadequate potential function may cause the agent to get into local minima. This happens, e.g., whenever the agent reaches a null-potential field location where the various potential components cancel each other. If this location is attractive (the gradient of the potential field points towards it on a non-empty neighborhood) it means that the agent oscillates around it indefinitely (i.e., it enters a limit cycle).

Much of the literature has, in recent years, concentrated on enhancements to the classical APF method: this is done by adding a new component to the field, modeled either through a virtual obstacle near the local minima to repel the agent [Lee and Park, 2003], [Sfeir et al., 2011] or by providing an additional control term for the USV to escape from local minima [Li et al., 2012b], [Luo et al., 2012]. This approach is tractable for real-time path planning but incurs a high computational cost. Other methods also proposed for dealing with the local minima are to combine potential field approach with heuristic algorithms such as ant colony optimization [Nazzal, 2017]; simulated annealing algorithm [Park et al., 2001], [Zhu et al., 2006] or its variant, so-called deterministic annealing (with noise inserted into the cost function) [Doria et al., 2013]. However, the common drawback of these approaches is their demand of a tremendous amount of computation power while still being at risk of failing to find a suitable solution. An improved APF based on regression search methodology employed to solve the local minima in a priori known environments has been addressed by [Li et al., 2012a], [Li et al., 2013] and [Yang et al., 2016]. Nevertheless, an operating space with many obstacles may lead to an incomplete path in the navigation task.

In general, the above methods are either complex in computation or challenging to solve in complex scenarios. We thus propose a methodology which avoids adding virtual repulsive potentials. This proposed method is the combination between the *logistic regression* and repulsive potential field whose goal is to diminish the number of repelling areas. Furthermore, the MAS's workspace is no longer constant but rather a variable environment depending on the agents' current positions. This reduces the null-potential regions, i.e., the probability of entering into a local minima is also decreased. The *logistic regression* notion is often found in mathematical applications as it allows to describe smooth transitions between binary values: *yes vs. no, positive*

vs. negative, dead vs. alive, etc. It is the go-to method for binary classification problems, and it's fundamental concepts are also used in deep learning [Huang et al., 2012], [Nadkarni, 2016].

The idea is for the MAS to avoid taking into consideration all of the repulsive potential fields representing static/dynamic obstacles along the prediction horizon. Instead, only the repulsive fields generated by regions (obstacles or other ships) sufficiently close are considered. This is handled by weighting each of the repulsive potential fields with a logistic function. The continuous variation of the function will ensure that a local minima area (if any) where the total repulsion and attraction forces are zero will be eliminated. Moreover, reducing the number of repulsive potential components, to those in the view range of the agent, reduces the computation time.

The present chapter proposes the application of the *logistic regression* function to repulsive potential functions to solve the local minima problem as well as to enhance the MAS performance. Specifically, the contributions of this chapter are the following:

- Use a logistic regression, as a *on-off barrier function* to activate the associated repulsive potentials in the range view of the agent for fixed and moving obstacles. The variation of the logistic regression function depend on the risk of collision as a function of the relative distance between agents and obstacles.
- The proposed method is validated through simulations over a real benchmark for the safe navigation of ships in the Trondheim fjord while complying to a subset of COLREGS rules. Comparisons will be provided to show the effectiveness of the proposed algorithm.

This chapter is organized as follows. Section 3.1 describes the potential field-based workspace where static/dynamic obstacles and USVs safe regions are considered as repulsive convex regions. Section 3.2 introduces and analyses the on-off barrier function. Section 3.4 shows the construction of on-off barrier functions in conjunction with repulsive potential. Lastly, all the elements are gathered and introduced into a nonlinear model predictive control framework to ensure motion planning with collision avoidance. Section 3.5 shows the simulation results over a real benchmark as well as comparisons with some alternative approaches. Section 3.6 draws the conclusions and discusses future work.

3.1 Potential field-based description of a costal environment

The tools presented in Chapter 2, e.g., the polytopic constraints, the sum function and the repulsive potential field for static obstacles will be further used for the motion planning of autonomous ships in a coastal environment. The final purpose is to develop and apply these tools over a real benchmark for the safe navigation of ships in the Trondheim fjord complying to COLREGS rules (see Appendix A), using real numerical data.

3.1.1 Dynamic repulsive potential field

To correctly characterize the dynamic repulsive potential fields, we first provide a simplified model of the USVs. This is followed by a description of the actual potential field, in terms of the associated polytopic sets.

Dynamical model of a USV

An autonomous surface vehicle experiences motions along 6 distinct degrees of freedom: *surge, sway, yaw, heave, roll, and pitch.* However, this is only considered for the case of a small waterplane area and low metacentric stability [Breivik, 2003]. In this manuscript, we concentrate on large ship dynamics where the low-frequency vertical-plane ship dynamics and the thruster action do not affect each other. Hence, we consider a simplified horizontal 3-DOF nonlinear dynamics model with *surge, sway* and *yaw* by neglecting the vertical motion [Fossen, 2011]. Cybership II (CSII): scale 1:70 of a supply ship, is such a typical reduced model (as shown in Fig. 3.1.1). It was developed in the Marine Cybernetics Laboratory at NTNU (Norwegian University of Science and Technology). In order to identify the physical and hydrodynamical quantities for this ship, many experimental tests have been carried out [Skjetne et al., 2004].

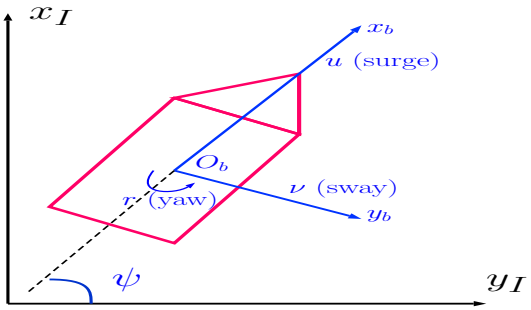


Figure 3.1.1: Earth-fixed (x_I, y_I) and body-fixed (x_b, y_b) frames.

As a consequence, from the nominal model in the general form of i^{th} agent as in (2.2.1), the mathematical model of the i^{th} underactuated surface vehicle[1] moving in a horizontal plane is described as in [Fossen, 2002]:

$$\dot{\eta}_i = R(\psi_i)\nu_i, \tag{3.1.1a}$$

$$M_i\dot{\nu}_i = -C_i(\nu_i)\nu_i - D_i\nu_i + \mathbf{u}_i + w_i, \tag{3.1.1b}$$

where $\eta_i = \begin{bmatrix} p_i & \psi_i \end{bmatrix}^\top \in \mathbb{R}^3$, composed from position $p_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^\top$ and yaw angle, ψ_i, gives the orientation relative to an Earth-fixed reference frame[2]. The vector $\nu_i = \begin{bmatrix} u_i & v_i & r_i \end{bmatrix}^\top \in \mathbb{R}^3$ includes the surge, sway and yaw rates. The additive disturbances are given by $w_i = \begin{bmatrix} w_{u_i} & w_{v_i} & w_{r_i} \end{bmatrix}^\top \in \mathbb{R}^3$. The input vector, $\mathbf{u}_i \in \mathbb{R}^3$, with $\mathbf{u}_i = \begin{bmatrix} \mathcal{T}_{u_i} & Y_\delta \delta_i & N_\delta \delta_i \end{bmatrix}^\top$ is the control input corresponding to surge thrust, rudder force and moment respectively. The system matrices M_i, $C_i(\nu_i)$ and $D_i \in \mathbb{R}^{3\times 3}$ are the mass, Coriolis and damping matrices, respectively and are given as:

$$M_i := \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix}, D_i := \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix} \text{ and } C_i := \begin{bmatrix} 0 & 0 & c_{13} \\ 0 & 0 & c_{23} \\ c_{31} & c_{32} & 0 \end{bmatrix}, \tag{3.1.2}$$

where $c_{31} = -c_{13} = m_{22}v_i - m_{23}r_i$ and $c_{23} = -c_{32} = m_{11}u_i$. The rotation matrix $R(\psi_i)$ is described by the following expression:

$$R(\psi_i) := \begin{bmatrix} \cos(\psi_i) & -\sin(\psi_i) & 0 \\ \sin(\psi_i) & \cos(\psi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.1.3}$$

[1] Underactuation means that the ship has a lower number of actuators (i.e., control inputs) than the number of states (degrees of freedom). Hence not all state components can be brought simultaneously at a desired value.

[2] Note that in the model (3.1.1a), the side-slip is neglected as presented in [Eriksen et al., 2019], therefore yaw and course are the same.

The control objective is to drive the autonomous surface vehicle state $\mathbf{x}_i = \begin{bmatrix}\eta_i & \nu_i\end{bmatrix}^\top \in \mathbb{R}^6$ to follow the reference state $\mathbf{x}_{i,ref}$, i.e., steer the ship from harbor to harbor while avoiding static and dynamic obstacles or other ships obeying the COLREGS rules.

Parametrization of polytopic constraints

To construct the repulsive potential functions for agents as well as the moving obstacles, it is necessary to parametrize their polytopic regions.

Hereinafter we consider the nominal model for ship dynamics (3.1.1) to characterize a team of N_a agents (presented first in a generic form in (2.2.1). Then, we represent the safety regions (attached to either the agents or to the moving obstacles), as polytopic regions parametrized by their current positions. Building upon from the Definition 2.1.2, which describes the convex regions of static obstacles, we provide now a parametrization for the moving obstacles.

Definition 3.1.1. *A polytope $\mathcal{A}_i$, $i \in \mathcal{V}$, characterizing the safety region (i.e., shape) of agent i, centered at position $p_i(t)$ is described by:*

$$\mathcal{A}_i = \{p \in \mathbb{R}^2 : a_m^i(p - p_i(t)) \leq b_m^i, m = 1, ..., n_h^i\}, \tag{3.1.4}$$

where $a_m^i \in \mathbb{R}^{1\times 2}$, $b_m^i \in \mathbb{R}$ define the n_h^i support hyperplanes of the polytopic set. Another interpretation of (3.1.4) is to say that $\mathcal{A}_i = \{p_i(t)\} \oplus \{p \in \mathbb{R}^2 : a_m^i p \leq b_m^i,\ m = 1 \dots n_h^i\}$, that is, the safety region is a polytopic set translated by the current position $p_i(t)$ of the agent.

Next, a piecewise linear function, i.e., the sum function defined as in (2.1.6) is applied to the safety region of agent i:

$$\theta_i(p_i(t)) = \sum_{k=1}^{n_h^i} (a_m^i(p - p_i(t)) - b_m^i + |a_m^i(p - p_i(t)) - b_m^i|). \tag{3.1.5}$$

The RP (Repulsive Potential) field for the agent is similarly described as in (2.2.12) by using the sum function (3.1.5)

$$\Upsilon_i(\theta_i(p_i(t))) = \frac{c_{1i}}{(c_{2i} + \theta_i(p_i(t)))^2}, \tag{3.1.6}$$

where c_{1i} and c_{2i} are positive parameters representing the strength and effect ranges of repulsive potential of agent i.

The repulsive field (3.1.6) is parametrized after the i-th agent position. In other words, (3.1.6) describes a repulsive potential function of agent i that changes over time. Similarly to the repulsive potential of fixed obstacles (2.2.12), it must satisfy two following essential properties: to have a high value inside the agent's safety region and decreases linearly as the distance from its bound increase. With construction (3.1.6) we observe that indeed these properties hold:

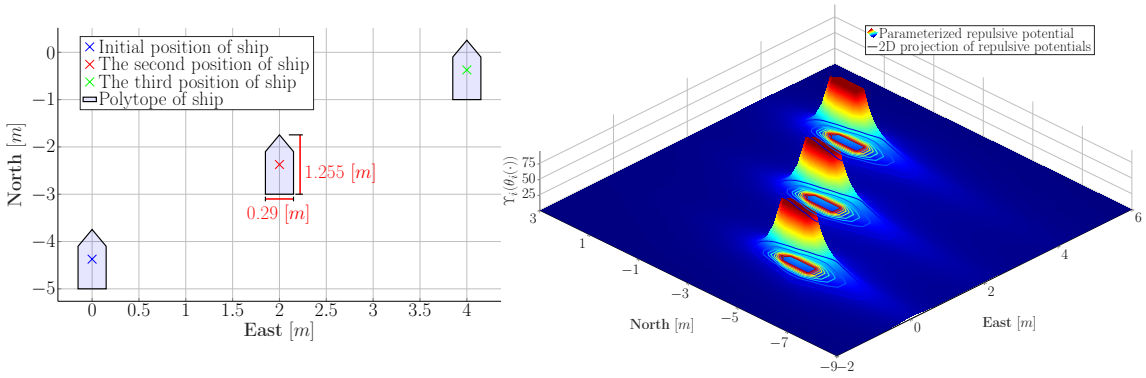
$$\inf_{p_i \notin Int(\mathcal{A}_i)} \Upsilon_i(\cdot) = 0, \qquad \sup_{p_i \in Int(\mathcal{A}_i)} \Upsilon_i(\cdot) = \frac{c_{1i}}{(c_{2i})^2}. \tag{3.1.7}$$

It is worth mentioning that the choice of strength and effect range in (3.1.6) must be carefully made with respect to the actual situation, the repulsive potential field of the agent cannot be too small (collision risk, i.e., non-compliance with COLREGS) or too large (low navigation efficiency).

Example 3.1.2. Consider a CyberShip II (a 1 : 70 scale of a real ship). Its length is $1.255m$ and its beam (breadth) is $0.29m$. In order to obtain the repulsive potential function of this ship, we first have to construct the ship's polytopic region w.r.t the given dimension.

$$\mathcal{A}_i = \left\{ p_i(t) \in \mathbb{R}^2 : \begin{bmatrix} 0.0 & -0.1961 \\ 0.9889 & 0.0 \\ 0.5211 & 0.2202 \\ -0.5211 & 0.2202 \\ -0.9889 & 0.0 \end{bmatrix} (p - p_i(t)) \leq \begin{bmatrix} 0.9806 \\ 0.1483 \\ -0.8246 \\ -0.8246 \\ 0.1483 \end{bmatrix} \right\}$$

Then, applying the the formula (3.1.5) and (3.1.6) for polytope $\mathcal{A}_i$ and parameters $c_1^i = 1$, $c_2^i = 0.1$, we obtain the repulsive potential field.



(a) Polytope is parametrized by relative coordinates of ship.

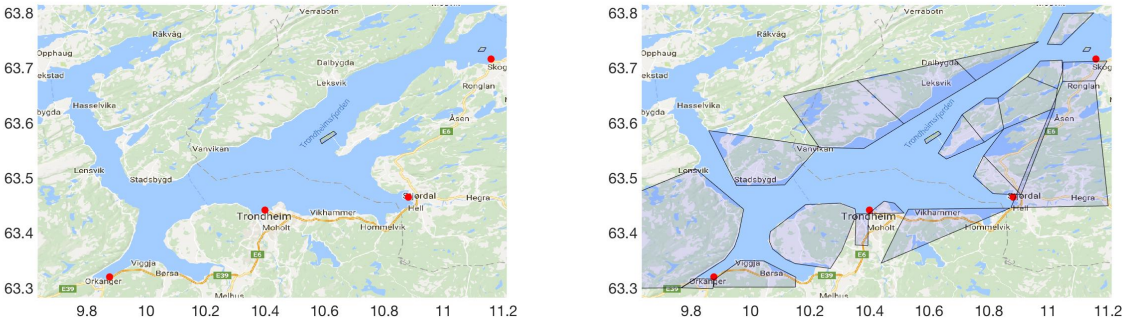
(b) Moving RP field following relative coordinates of ship in Fig. 3.1.2a.

Figure 3.1.2: Illustration of the polytopic region and dynamic RP field.

The safety region of the ship is represented by polytope $\mathcal{A}_i$, parametrized after the current position p_i at different times t ($p_i(t)$ is the center of the vessel), as depicted in Fig. 3.1.2a. Fig. 3.1.2b illustrates the corresponding repulsive potential functions w.r.t its position $p_i(t)$: high inside of the polytopic region and having a sudden drop to zero outside of the convex region.

3.1.2 Static repulsive potential field

In what follows we consider a workspace in which multiple ships travel. This is illustrated through *M_MAP: a mapping package for Matlab* [Pawlowicz, 2019]. It is based on real data of the coordinates at geographic coordinate system with the *longitude* (180^o East $\leftrightarrow$ 180^o West) and *latitude* (90^o North $\leftrightarrow$ 90^o South) display in degree units for the Trondheim fjord, as can be seen in Fig. 3.1.3a, where the ground is represented by light green and the sea is light blue.



(a) Trondheim fjord based on real data.

(b) Set of forbidden polytopic convex regions.

Figure 3.1.3: Trondheim fjord and bounded convex regions based on real data.

In this scenario, the mainland/islands are considered as the static obstacle, given as a union of bounded convex regions. It is worth noting that we only need to cover the coastline, in our description as this is the part which is actually in danger of collision, as illustrated in Fig. 3.1.3b. The four red dots in Fig. 3.1.3 stand for the harbors.

In order to proceed, we first have to convert longitude/latitude measurements into position coordinates in the horizontal plane. The cosine-Haversine formula[3] [Robusto, 1957] is used to determine the great-circle distance between two points on a sphere given their longitudes and latitudes, and is defined by the following expression:

$$d = 2R_{earth} \sin^{-1}\left(\sqrt{\sin^2\left(\tfrac{\varphi_2-\varphi_1}{2}\right) + \cos(\varphi_1)\cos(\varphi_2)\sin^2\left(\tfrac{\lambda_2-\lambda_1}{2}\right)}\right) \tag{3.1.8}$$

where φ_1, φ_2 are the latitudes of point 1 and 2, λ_1, λ_2 are longitude values for point 1 and 2; R_{earth} is the radius of the earth in kilometers. Thus, the terrain shown in Fig. 3.1.3b, is

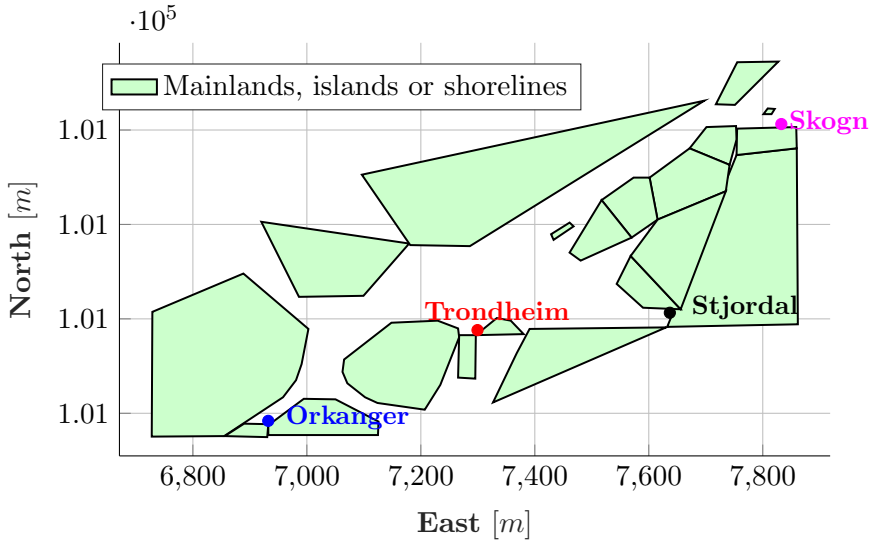


Figure 3.1.4: 1:70 scale of workspace after using the cosine-Haversine formula.

over-approximated by a union of 22 bounded convex regions in the horizontal plane with a 1:70

[3]There are two ways to determine this distance, one is the cosine-Haversine formula, and the other is the Pythagorean theorem. Haversine formula, however, is more accurate especially for longer distances.

scaling[4] of the real map after applying the cosine-Haversine formula, as illustrated in Fig. 3.1.4.

$\mathbb{S} \in \mathbb{R}$ is the set of static repulsive potential fields, each of them computed as in (2.2.12) for the polytopic regions covering the mainland in Fig. 3.1.4:

$$\mathbb{S} = \bigcup_{\ell=1}^{N_{\text{obs}}} \Phi_\ell^f(\gamma_\ell(p_i)), \tag{3.1.9}$$

and is depicted in Fig. 3.1.5.

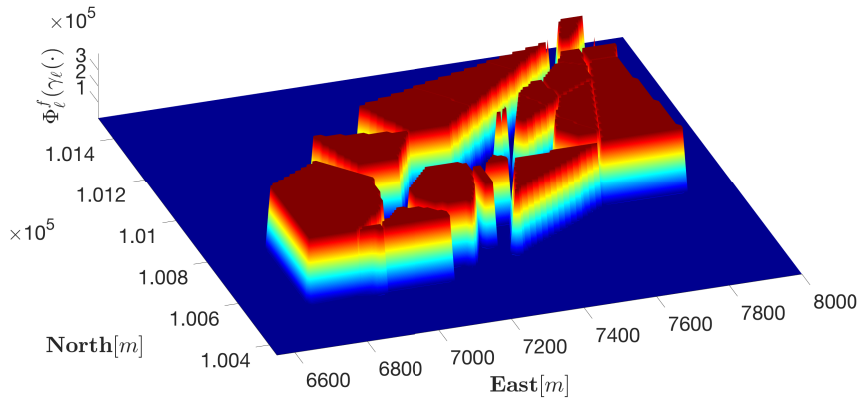


Figure 3.1.5: The static repulsive potential fields for the mainland in Trondheim fjord.

3.2 Logistic Regression analysis

The paradigm *logistic regression* is technique borrowed by machine learning algorithm. It is used to classify two-class values (yes vs. no, positive vs. negative, dead vs. alive, etc.). It is the go-to method for binary classification problems and its fundamental concepts are also constructive in deep learning [Huang et al., 2012].

Logistic regression is named for the function which is employed at the core of the method, the logistic function, also called the sigmoid function was developed by statisticians and was first applied for population growth studies in ecology as well as reaching the limit of the carrying capacity of the environment [Kyurkchiev and Markov, 2016]. It has an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1. The sigmoid function can be described as:

$$sig(x) = \frac{L}{1 + e^{-(x - x_o)}}, \tag{3.2.1}$$

where x_o is the value of the sigmoid's midpoint and L is the curve's maximum value.

For values of $x \in \mathbb{R}$, the value of sigmoid function $sig(x)$ is obtained by approaching L as x approaches $+\infty$ and approaching zero as x approaches $-\infty$.

[4] A 1:70 scale of the map is chosen in order to match the 1:70 scale of the ship considered in these simulations.

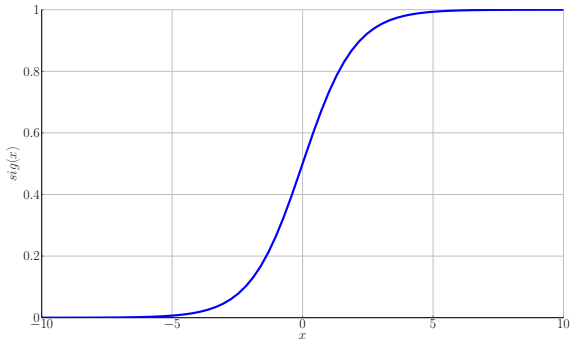


Figure 3.2.1: The standard logistic function.

The standard logistic function is the logistic function with parameters $L = 1$, $x_o = 0$ and is illustrated in Fig. 3.2.1.

Logistic regression is a technique for analyzing problems in which there are one or more independent variables (i.e., input value, x) used to predict a dependent variable (i.e., output value, y) by using the design parameter, β. From statistical decision theory, it is well known that the optimal dependent variable decision is to choose the class label y maxing out the posterior distribution $F(y|x,\beta)$ [Stork et al., 2006]. In most cases, class label y representing a dependent variable is assigned class labels 0 or 1 to a corresponding independent variable.

As a consequence, a logistic regression representation based on logistic function (3.2.1), which models the probability of the default class (e.g., the first class in two-class values (0 and 1)). In other words, we are modeling the probability that an input variable x belongs to the default class ($y = 1$); one can write this formally as:

$$F(x,\beta) = F(y=1|x,\beta) = \frac{L}{1+e^{-\beta(x-x_o)}} \approx \begin{cases} [0,L), & \text{if } x < x_o \ , \\ L, & \text{if } x \geq x_o \ , \end{cases} \tag{3.2.2}$$

where β is the design parameter for the single independent variable x and L must be chosen equal to 1.

Its derivative is:

$$\frac{\partial F(x,\beta)}{\partial x} = \frac{L\beta e^{(-\beta(x-x_o)}}{(1+e^{-\beta(x-x_o)})^2}. \tag{3.2.3}$$

In case of the default class ($y = 0$), one can present this as follows:

$$\overline{F}(x,\beta) = F(y=0|x,\beta) = \frac{L}{1+e^{\beta(x-x_o)}} \approx \begin{cases} 0, & \text{if } x > x_o \ , \\ (0,L], & \text{if } x \leq x_o \ . \end{cases} \tag{3.2.4}$$

where L is chosen to be 1.

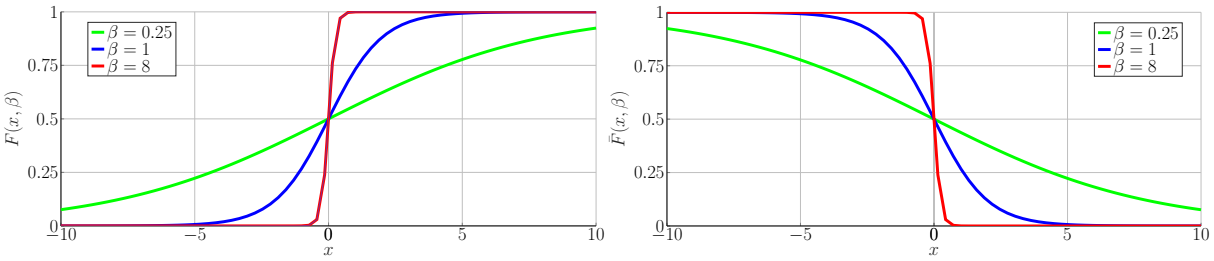
Equivalently, we may consider relation: $\overline{F}(x,\beta) = 1 - F(x,\beta)$.

Its derivative is:

$$\frac{\partial \overline{F}(x,\beta)}{\partial x} = -\frac{L\beta e^{(\beta(x-x_o)}}{(1+e^{\beta(x-x_o)})^2}. \tag{3.2.5}$$

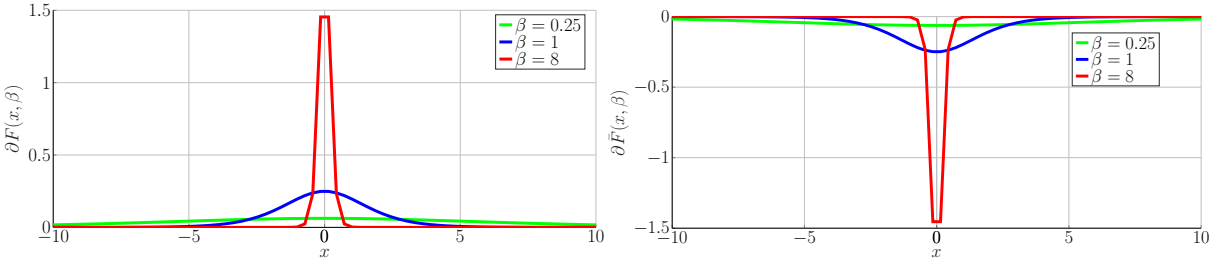
Remark 3.2.1. It's worth noting that L is chosen to be 1 to guarantee the output values of logistic function stay into the range $[0, 1]$ (i.e., logistic function is considered as classify two-class values).

The hyperplane of all points x satisfying the equation $\beta(x - x_o) = 0$ forms the decision boundary between the two classes, these are the points for which $F(x, \beta) = \bar{F}(x, \beta) = \frac{L}{2} = 0.5$ (assuming that we take $L = 1$).



(a) Logistic regression $F(x, \beta)$ (3.2.2) with different β. (b) Logistic regression $\overline{F}(x, \beta)$ (3.2.4) with different β.

Figure 3.2.2: Illustration of logistic regression.



(a) $\partial F(y = 1|x, \beta)$ (3.2.3) with different β. (b) $\partial F(y = 0|x, \beta)$ (3.2.5) with different β.

Figure 3.2.3: Illustration of derivative of logistic regression.

Fig. 3.2.2 and 3.2.3 illustrate the logistic regression and its derivative with $L = 1$, $x_0 = 0$ with the different design parameters β. The change of β plays a decisive role in the logistic regression, i.e., increasing β means that the evolution towards one of the limit values (either 0 or 1) becomes faster.

By regulating the design parameter β, the logistic regression can be considered as an *on-off barrier function* (0 accounts for *"on"* level and 1 accounts for *"off"*) while in the same time being "well-behaved" (in the sense that it is a smooth function with reasonable derivative values). In the next section, the combination of on-off barrier functions, i.e., logistic regression and repulsive potential functions will be discussed in detail.

Gathering all the elements presented above, we will concentrate next on the NMPC implementation for motion planning with collision avoidance. Particular attention is given to the formulation of the cost function through which we penalize the collision avoidance constraints with fixed and moving obstacles.

3.3 NMPC implementation

A typical approach for the use of potential field methods is to construct the total potential (attractive and repulsive) and add it into an optimization framework, possibly looking over a prediction horizon. However, this approach is computationally cumbersome, especially for demanding applications when the number of fixed and moving obstacles is large. This may lead to infeasible solutions for the optimization problem, usually due to local minima. Therefore, this section introduces first the Chebyshev center to describe the safe distance between an agent and static obstacles as well the distance to another agent. On-off barrier functions are used to weight the repulsive potential fields such as to simplify the overall formulation. This means that the various repulsive terms appearing in the objective function will be activated only when needed. Next, the NMPC optimization problem and the implementation algorithm are delineated.

3.3.1 Chebyshev center for static polytope configuration

Consider a bounded convex polyhedron described by its $\mathcal{H}-$*representation* as in Definition 2.1.2, $\mathcal{P} = \{x \in \mathbb{R}^n \mid a_k x \leq b_k,\ k = 1, ..., n_h\}$. The Chebyshev center x_c of polytope $\mathcal{P}$ is defined as the center of the largest inscribed ball $\mathcal{B}_\ell = \{x \in \mathbb{R}^n \mid \|x - x_c\| < \rho\}$ inside $\mathcal{P}$. The values of (x_c, ρ) are yielded by solving the optimization:

$$\max \quad \rho \tag{3.3.1a}$$

subject to

$$a_k x_c + \|a_k\|\rho \leq b_k, \tag{3.3.1b}$$
$$\rho \geq 0. \tag{3.3.1c}$$

It's worth noting that (3.3.1b) can be exploited through Karush-Kuhn-Tucker condition [Boyd and Vandenberghe, 2004], the equation describing the center x_c and the radius ρ can be defined as follows:

$$a_{l^*} x_c + \|a_{l^*}\|\rho = b_{l^*}, l^* \in \{1, \ldots, n_h\}$$

where l^* is the index of the activated constraints. Consequently, if $\begin{bmatrix} a_k & \|a_k\| \end{bmatrix} \in \mathbb{R}^{n \times n}$ is invertible, one can describe directly the relationship of Chebyshev center x_c and its radius ρ in terms of the $\mathcal{H}-$*representation* of $\mathcal{P}$, i.e.,

$$\begin{bmatrix} x_c \\ \rho \end{bmatrix} = \begin{bmatrix} a_k & \|a_k\| \end{bmatrix}^{-1} b_k. \tag{3.3.2}$$

Example 3.3.1. Let us consider the polytope $\mathcal{P}$ in Example. 2.1.5. Its $\mathcal{H}-$*representation* is

$$\mathcal{P} = \left\{ x \in \mathbb{R}^2 : \begin{bmatrix} -0.2691 & -0.2018 \\ 0.1871 & -0.0234 \\ 0.1156 & 0.1445 \\ -0.1046 & 0.0262 \end{bmatrix} x \leq \begin{bmatrix} 0.9417 \\ -0.9821 \\ 0.9827 \\ 0.9942 \end{bmatrix} \right\}$$

Solve the problem (3.3.1) for obtaining the Chebyshev center $x_c = \begin{bmatrix} -6.16 \\ 6.37 \end{bmatrix}$ (red dot in Fig. 3.3.1) and its radius $\rho = 1.69$ (blue line in Fig. 3.3.1).

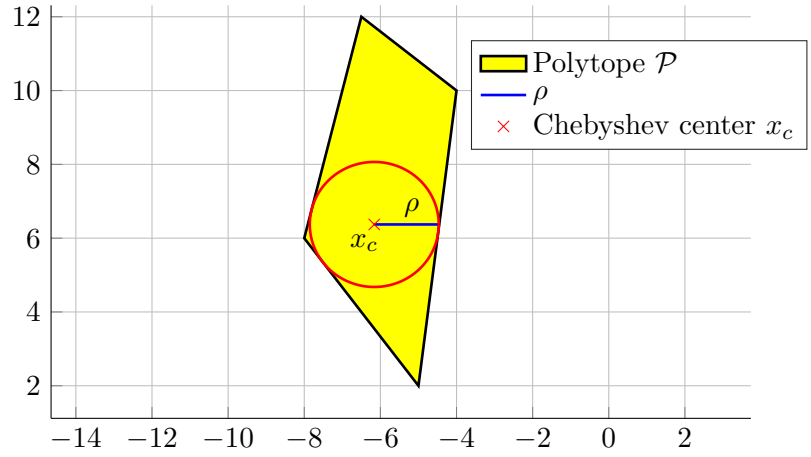


Figure 3.3.1: Chebyshev center of bounded convex polyhedron $\mathcal{P}$.

Equivalently, we may compute directly x_c and ρ by using (3.3.2):

$$\begin{bmatrix} -0.2691 & -0.2018 & \| -0.2691 & -0.2018 \| \\ 0.1871 & -0.0234 & \|0.1871 & -0.0234 \| \\ 0.1156 & 0.1445 & \|0.1156 & 0.1445 \| \\ -0.1046 & 0.0262 & \| -0.1046 & 0.0262 \| \end{bmatrix} \begin{bmatrix} x_c \\ \rho \end{bmatrix} = \begin{bmatrix} 0.9417 \\ -0.9821 \\ 0.9827 \\ 0.9942 \end{bmatrix}.$$

3.3.2 On-off repulsive potential description

As aforementioned in Chapter 2, although there are some advantages relating to the collision avoidance capabilities and the problem's feasibility in the nonlinear model predictive control scheme, the main shortcoming of potential field approach is the existence of local minima. In particular, MAS operates in a complicated environment in the presence of plenty of repulsive potential fields representing static and dynamic obstacles as can be observed in Fig. 3.1.2b and 3.1.5. Consequently, we propose a combination of the on-off barrier function (3.2.4) with the two types of repulsive potential functions considered: for fixed obstacles (2.2.12) and for moving obstacles (3.1.6). The goal is to consider only the repulsive potential fields for which the safe distances between agents and (static and dynamic) obstacles are violated.

The Chebyshev center and its radius are fundamental in the definition of a safe distance between i^{th} agent and ℓ^{th} fixed obstacle as it serves in activating the repulsive potential through an associated on-off barrier function (i.e., the logistic regression discussed earlier).

Consider a set $\mathcal{V}$ of N_a agents, given in a generic form in (2.2.1), operating in working space $\mathcal{F} \in \mathbb{R}^n$ in the presence of a set $\mathbb{O}$ (2.2.9), the union of forbidden polytopic convex regions representing N_{obs} fixed obstacles, $\mathcal{O}_\ell = \{p_i \in \mathbb{R}^n \mid a_k^\ell p_i \leq b_k^\ell, k = 1, \ldots, n_h, \ell = 1, \ldots, N_{\text{obs}}\}$.

As a result, workspace $\mathcal{F}$ of MAS contains two sets, one is the set $\mathbb{S}$ defined in (3.1.9) of N_{obs} static repulsive potential fields which stand for the prohibited convex regions of fixed obstacles and another is the set $\mathbb{M}$ of N_a dynamic repulsive potential fields representing the safe convex regions of dynamic obstacles or other agents, i.e.,

$$\mathbb{S}, \mathbb{M} \subset \mathcal{F}, \tag{3.3.3}$$

where $\mathbb{M}$ is

$$\mathbb{M} = \bigcup_{i=1}^{N_a} \Upsilon_i(\theta_i(p_i(t))), \tag{3.3.4}$$

with $\Upsilon_i(\cdot)$ mentioned in (2.2.12) and (3.1.6).

Assumption 3.3.2. *Assume that agent $i \in \mathcal{V}$ can perceive information from the other agents and the forbidden cells (fixed obstacles[5]) as well as transmit relevant information to others within the n-th dimensional range $\Gamma_i(p_i, r_i^v)$, where $r_i^a \in \mathbb{R} > 0$ is the radius of the ball centered in $p_i \in \mathbb{R}^n$, the current position of agent i.*

The range $\Gamma_i(p_i, r_i^v)$, in this manuscript, is considered as the *view range* of i^{th} agent (i.e., the zone where the ship can detect all objects).

For static obstacles

Definition 3.3.3 (Safe distance between agent and static obstacle)**.** *The safe distance $D_s^{i,\ell} \in \mathbb{R}_{>0}$ between the ℓ^{th} static obstacle and i^{th} agent is defined based on the expanded ball $\mathcal{B} \in \mathbb{R}^n$ which is obtained from Chebyshev center so that it can cover completely the ℓ^{th} static obstacle and the view range of i^{th} agent with the following expression:*

$$D_s^{i,\ell} = \varepsilon_\ell \rho_\ell + \Gamma_i, \tag{3.3.5}$$

where $\varepsilon_\ell > 0$ is a scaling parameter, ρ_ℓ is the radius of the circle defined from Chebyshev center of the ℓ^{th} forbidden cell, Γ_i is the view range of agent i.

The on-off barrier function for default class 0, presented in (3.2.4) with $L = 1$ is redefined here for the ℓ^{th} static obstacle and i^{th} agent as:

$$\overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_{i\ell}) = \frac{1}{1 + e^{\beta_{i\ell}(\|\Delta p_{i,\ell}^s\| - D_s^{i,\ell})}} \approx \begin{cases} 0, & \text{if } \|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}, \\ (0,1], & \text{if } \|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}, \end{cases} \tag{3.3.6}$$

where $\|\Delta p_{i,\ell}^s\|$ denotes Euclidean distance between current position of i^{th} agent and Chebyshev center of ℓ^{th} static obstacle (see Fig. 3.3.2).

Definition 3.3.4 (Activation of the static repulsive potential field)**.** *Given a set $\mathbb{S}$ of $N_{obs} \in \mathbb{N}$ static repulsive potential fields as defined in* (3.1.9)*, the ℓ^{th} static repulsive potential field will be activated by the i^{th} on-off barrier function if the Euclidean distance between Chebyshev center of the ℓ^{th} static obstacle and the current position of i^{th} agent, $\|\Delta p_{i,\ell}^s\|$, is smaller than or equal to a given safe distance $D_s^{i,\ell}$.*

This can be described by the following expression:

$$\begin{aligned} \Omega_{\text{static}}^{i,\ell}(p_i, x_{c\ell}) &= \sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_{i\ell}) \Phi_\ell^f(\gamma_\ell(p_i)) \\ &\approx \begin{cases} 0, & \text{if } \|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}, \\ (0,1]\Phi_\ell^f(\gamma_\ell(p_i)), & \text{if } \|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}, \end{cases} \end{aligned} \tag{3.3.7}$$

[5]Fixed obstacles such as islands and shorelines should be known from map information. The ship's position in relation to the obstacles is known using Global Navigation Satellite Systems.

where $\Phi_\ell^f(\gamma_\ell(p_i))$ is given in (2.2.12), $\overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_i)$ in (3.3.6) and $x_{c\ell}$ is the Chebyshev center of ℓ^{th} fixed obstacle solved by (3.3.1).

By using the on-off barrier function for default class 0 (i.e., value 0 is prioritized) we classify two-class values, "0" and "1" based on a defined safe distance $D_s^{i,\ell}$ with regard to the current position in time-variant of i^{th} agent. There are two possibilities for (3.3.7):

- If $\|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}$, then $\overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_{i\ell})$ will classify in class "0", i.e., the probability of risk of collision is 0. This means that the static repulsive potential function will be relaxed.
- If $\|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}$, then $\overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_{i\ell})$ will classify in class "1", i.e., the probability of collision will start increasing from 0 to 100% depending on $\|\Delta p_{i,\ell}^s\|$. This means that the static repulsive potential function will be activated in order to push the agent further away. As a result, $\overline{F}_{i,\ell}(\|\Delta p_{i,\ell}^s\|, \beta_{i\ell})$ returns to class "0".

Proposition 3.3.5 (Dealing with local minima)**.** *If a set $\mathbb{S}$ of $N_{obs} \in \mathbb{N}$ representing static repulsive potential fields* (3.1.9) *are only partially activated by the agent's view range through on-off barrier functions as defined in* (3.3.7)*, then a null-potential field*[6] *that causes local minima can be excluded.*

Proof. See Appendix C. □

For other agents/dynamic obstacles

Definition 3.3.6 (Safe distance of agent and dynamic obstacle/the other agents)**.** *The safe distance of the i^{th} agent with respect to j^{th} agent (or dynamic obstacle), $D_d^{i,j} \in \mathbb{R}_{>0}$ is also its view range with the following expression:*

$$D_d^{i,j} = \Gamma_i. \tag{3.3.8}$$

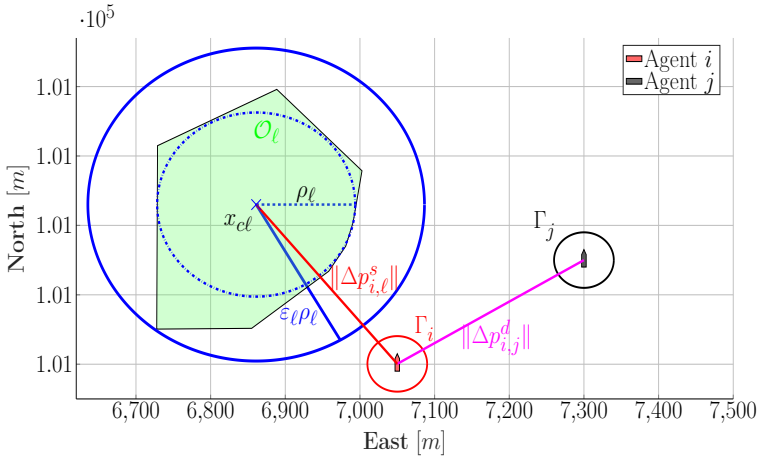


Figure 3.3.2: Description of the safe distance of the agent i w.r.t. the ℓ^{th} cell and agent j.

The on-off barrier function for default class "0", presented in (3.2.4) with $L = 1$ is redefined here for the i^{th} and j^{th} agent (or dynamic obstacle) as:

$$\overline{F}_{i,j}(\|\Delta p_{i,j}^d\|, \beta_{ij}) = \frac{1}{1 + e^{\beta_{ij}(\|\Delta p_{i,j}^d\| - D_d^{i,j})}} \approx \begin{cases} 0, & \text{if } \|\Delta p_{i,j}^d\| > D_d^{i,j}, \\ (0,1], & \text{if } \|\Delta p_{i,j}^d\| \leq D_d^{i,j}, \end{cases} \tag{3.3.9}$$

[6] Where the direction of total force and the movement direction of planning path are on the same straight line.

where $\|\Delta p^d_{i,j}\|$ denotes the Euclidean distance between the current position of i^{th} and j^{th} agent.

Definition 3.3.7 (Activating the dynamic repulsive potential field)**.** *Given the set $\mathbb{M}$ of N_a dynamic repulsive potential fields, the i^{th} dynamic repulsive potential field will be activated by i^{th} on-off barrier function if the Euclidean distance between the current position of i^{th} and j^{th} agent (or dynamic obstacle), $\|\Delta p^d_{i,j}\|$, is smaller than or equal to a given safe distance $D^{i,j}_d$, $\forall i, j \in \mathbb{M}$, $i \neq j$.*

This can be described by the following expression:

$$\begin{aligned}\Omega^{i,j}_{\text{dynamic}}(p_i, p_j) &= \sum_{i=1}^{N_a} \overline{F}_{i,j}(\|\Delta p^d_{i,j}\|, \beta_{ij})\Upsilon_i(\theta_i(p_i)) \\ &\approx \begin{cases} 0, & \text{if } \|\Delta p^d_{i,j}\| > D^{i,j}_d \ , \\ (0,1]\Upsilon_i(\theta_i(p_i)), & \text{if } \|\Delta p^d_{i,j}\| \leq D^{i,j}_d \ , \end{cases}\end{aligned} \tag{3.3.10}$$

where $\Upsilon_i(\theta_i(p_i))$ is given in (3.1.6), $\overline{F}_{i,j}(\|\Delta p^d_{i,j}\|, \beta_{ij})\Upsilon_i(\theta_i(p_i))$ in (3.3.9).

By employing the on-off barrier function for default class 0 (i.e., value 0 is prioritized) we classify two-class values, "0" and "1" based on a defined safe distance $D^{i,j}_d$ between the time-varying positions of the i^{th} and j^{th} agent. There are two possibilities for (3.3.10):

- If $\|\Delta p^d_{i,j}\| > D^{i,j}_d$, then $\overline{F}_{i,j}(\|\Delta p^d_{i,j}\|, \beta_{ij})$ will classify in class "0", i.e., the probability of risk of collision is 0%. This means that the static repulsive potential function will be relaxed.
- If $\|\Delta p^d_{i,j}\| \leq D^{i,j}_d$, then $\overline{F}_{i,j}(\|\Delta p^d_{i,j}\|, \beta_{ij})$ will classify in class "1", i.e., the probability of collision will start increasing from 0 to 100% depending on $\|\Delta p^d_{i,j}\|$. This means that the static repulsive potential function will be activated in order to push the agent further away. As a result, $\overline{F}_{i,j}(\|\Delta p^d_{i,j}\|, \beta_{ij})$ returns to class "0".

With a dichotomous mechanism as implemented through the on-off barrier function for the static/dynamic repulsive potential fields many of the repulsive field components are relaxed (as they are out of the agent's view range).

Example 3.3.8 (On-off repulsive potential fields are relaxed). Let us consider two agents ($N_a = 2$) operating in the working space $\mathcal{F} \in \mathbb{R}^2$ in the presence of the ℓ^{th} forbidden polytopic fixed obstacle $\mathcal{O}_\ell$ described by its $\mathcal{H} - representation$:

$$\mathcal{O}_\ell = \left\{ \mathbf{p}_i \in \mathbb{R}^2 : 10^{-3} \begin{bmatrix} 0.0001 & -0.0100 \\ 0.0084 & -0.0105 \\ 0.0177 & -0.0112 \\ 0.0455 & -0.0131 \\ 0.1092 & -0.0175 \\ 0.0095 & 0.0093 \\ -0.0052 & 0.0103 \\ -0.1584 & 0.0007 \end{bmatrix} \mathbf{p}_i \leq \begin{bmatrix} -1.0 \\ -1.0 \\ -1.0 \\ -1.0 \\ -1.0 \\ 1.0 \\ 1.0 \\ -1.0 \end{bmatrix} \right\}$$

Solving Problem (3.3.1) gives the Chebyshev center $x_{c\ell} = 10^5 . \begin{bmatrix} 0.0686 \\ 1.0073 \end{bmatrix}$ (red dot in Fig. 3.3.3) and its radius $\rho_\ell = 132.6$ (dot line in Fig. 3.3.3). The dot blue circle is created by Chebyshev center and ρ_ℓ.

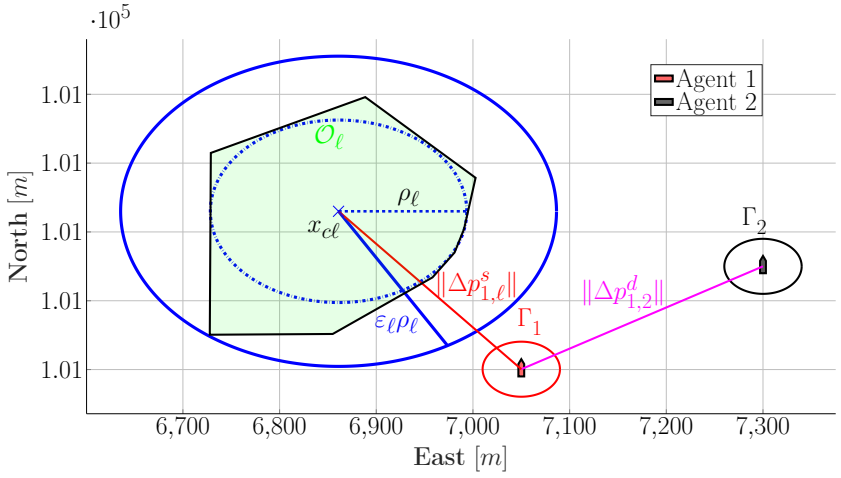


Figure 3.3.3: Chebyshev center of bounded convex polyhedron $\mathcal{O}_\ell$.

The scaling parameter, ε_ℓ is taken 1.8 so that the scaled Chebyshev circle completely covers the fixed obstacle (the blue circle). The view range of agent 1 and 2, Γ_1 and Γ_2 are equal to $40[m]$. Assume that the current position of agent 1, $p_1 = \begin{bmatrix}7050 & 100500.5\end{bmatrix}^\top$ and agent 2, $p_2 = \begin{bmatrix}7300 & 100650\end{bmatrix}^\top$. It is straightforward to compute the Euclidean distance between agent 1 and ℓ^{th} fixed obstacle:

$$\|\Delta p^s_{1,\ell}\| = \|p_1 - x_{c\ell}\| = 297.3[m]$$

and between agent 1 and agent 2:

$$\|\Delta p^d_{1,2}\| = \|\Delta p^d_{2,1}\| = \|p_1 - p_2\| = 291.29[m]$$

as well as safe distances as defined in (3.3.5) and (3.3.8), $D_s^{1,\ell} = 278.68[m]$ and $D_s^{1,2} = D_s^{2,1} = 40[m]$.

As a consequence, $\|\Delta p^s_{1,\ell}\| > D_s^{1,\ell}$, $\|\Delta p^d_{1,2}\| > D_d^{1,2}$ and $\|\Delta p^d_{2,1}\| > D_d^{2,1}$, leading to the static and dynamic repulsive potential field for are relaxed following (3.3.7) and (3.3.10).

3.4 NMPC optimization problem

This section presents a collision avoidance algorithm via a coherent combination between potential field-based constructions and NMPC. It is applied over a benchmark using real numerical data for the safe navigation of multiple ships in Trondheim fjord complying to COLREGS rules.

Assume that each agent (i.e., ship) has an associated dynamic repulsive potential field (3.1.6) constructed by a polytopic safety region as defined in (3.1.4). The probability distribution of collision is described by a dynamic RP for the agent. It will prevent any intrusion of the other agents through the choice of the repulsive field components, i.e., the i^{th} agent's current position has to be exterior of the repulsive potential field of j^{th} agent and vice-versa, i.e.,

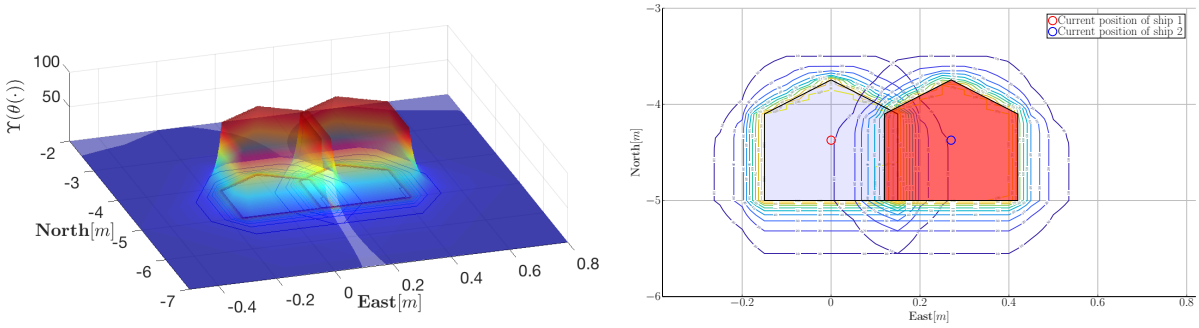
$$p_i \notin \Upsilon_j(\cdot), \tag{3.4.1a}$$
$$p_j \notin \Upsilon_i(\cdot), \tag{3.4.1b}$$

where $p_i \in \mathbf{x}_i$, $p_j \in \mathbf{x}_j$ note the current positions of agent i and j $(i, j \in N_a)$, $\Upsilon_i(\cdot)$ and $\Upsilon_j(\cdot)$ as defined in (3.1.6).

Inequalities (3.4.1) ensure that collision avoidance is guaranteed, i.e., the intersection of their bounded convex regions standing for shapes of agents is empty, i.e., $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$. The action of collision avoidance is soon or late, quick or tardy, depends on the strength and effect range parameters of $\Upsilon_i(\cdot)$ and $\Upsilon_j(\cdot)$.

Example 3.4.1. Consider two CyberShip II, as presented in Example 3.1.2.

- *Case 1*: The strength and effect range of $\Upsilon_1(\cdot)$ and $\Upsilon_2(\cdot)$ are chosen by $c_{11} = c_{12} = 1$ and $c_{21} = c_{22} = 0.1$,

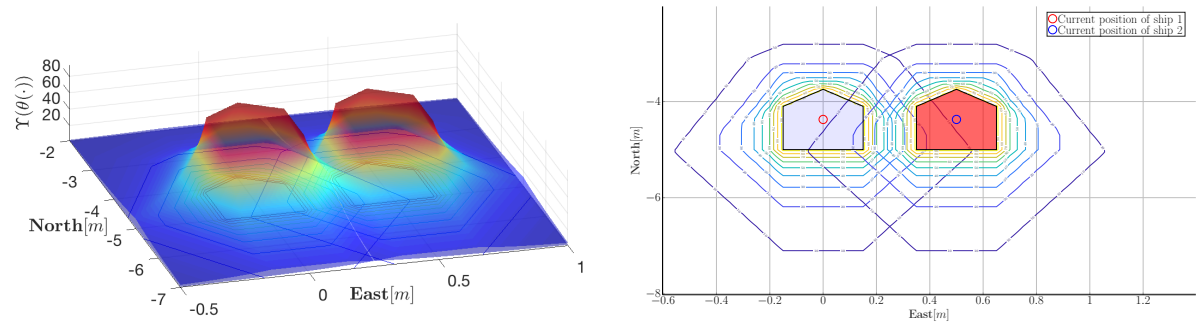


(a) Dynamic RP field of ship 1 and 2.

(b) 2D projection of dynamic RP field of ship 1 and 2.

Figure 3.4.1: Illustration of dynamic RP fields and their 2D projections of two ships in *Case 1*.

- *Case 2*: The strength and effect range of $\Upsilon_1(\cdot)$ and $\Upsilon_2(\cdot)$ are chosen by $c_{11} = c_{12} = 15$ and $c_{21} = c_{22} = 0.4$,



(a) Dynamic RP field of ship 1 and 2.

(b) 2D projection of dynamic RP field of ship 1 and 2.

Figure 3.4.2: Illustration of dynamic RP fields and their 2D projections of two ships in *Case 2*.

Fig. 3.4.1a and 3.4.2a illustrate the deep red parts representing the shape of ships (i.e., the bounded convex regions of ships, $\mathcal{A}_1$ and $\mathcal{A}_2$) are the regions of the absolute collision. Risk of collision will be reduced in intensity as the color becomes shallower. The deep blue parts are the regions of absolute safety. In *Case 1*, although the current positions of ship 1 and 2 do not violate repulsive regions of each other, a collision happened between two ships since their effect ranges are not large enough, as illustrated in Fig. 3.4.1b. *Case 2* as can be seen in Fig. 3.4.2b, when the parameters of dynamic RP such as strength and effect range have been expanded, leads to an increased distance. In other words, the collision avoidance actions between two ships was taken sooner, meaning that the risk of collision was greatly reduced.

Problem 3.4.2 (OORP (On-Off Repulsive Potential)-based tracking MPC)**.** *The potential field approach, at this point, is the combination of the on-off RP fields representing static and dynamic obstacles (or the other agents)* (3.3.7) *and* (3.3.10) *and attractive potential function describing the goal configuration at a global energy minimum presented by quadratic function* (2.2.11)*. This is reformulated by finite prediction horizon strategies as in Problem 2.2.8 in order to consider only the repulsive potential fields in the view ranges of MAS that lead to an enhancement of the feasibility of the obtained solution in connection with the collision-free path for MAS through minimizing a given cost function taking stage cost* $\mathcal{L}_i^{\mathrm{OORP}}(\cdot)$ *into account:*

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) + \int_{t_k}^{t_k+T_p} \left[\mathcal{L}_i^{\mathrm{OORP}}(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\tau))\right] d\tau, \tag{3.4.2a}$$

subject to

$$\dot{\bar{\boldsymbol{x}}}_i(\tau) = f(\bar{\boldsymbol{x}}_i(\tau), \bar{\boldsymbol{u}}_i(\tau)),\ \bar{\boldsymbol{x}}_i(t_k) = \boldsymbol{x}_i(t_k), \tag{3.4.2b}$$

$$\bar{\boldsymbol{x}}_i(\tau) \in \mathcal{X}_i;\ \bar{\mathbf{u}}_i(\tau)) \in \mathcal{U}_i,\ \forall \tau \in [t_k, t_k+T_p], \tag{3.4.2c}$$

$$\bar{\mathbf{x}}_i(t_k+T_p) \in \mathcal{X}_i^f, \tag{3.4.2d}$$

where $T_p \in \mathbb{R}_{>0}$ is the prediction horizon and t_k is discrete sampling instant, $\bar{\mathbf{x}}_i(\tau)$ and $\bar{\mathbf{u}}_i(\tau)$ are the predicted state and input while $\bar{\mathbf{u}}_i(\cdot)$ accounts for the whole predicted input trajectory along the prediction horizon. $\bar{p}_i(\tau)$ is the predicted position. (3.4.2b) is the state-space formulation in the general form of the ship (3.1.1), the set $\mathcal{X}_i$ and $\mathcal{U}_i$ presented in (2.2.4) and (2.2.5). Terminal constraint (3.4.2d) used to enforce nominal stability [Chen and Allgöwer, 1998], might or might not be available.

The cost per stage $\mathcal{L}_i^{\mathrm{OORP}}(\cdot)$ in (3.4.2a) is a composite of multiple terms with the following expression:

$$\mathcal{L}_i^{\mathrm{OORP}}(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\cdot)) = \Omega_{\mathrm{static}}^{i,\ell}(\bar{p}_i(\tau), x_{c\ell}) + \Omega_{\mathrm{dynamic}}^{i,j}(\bar{p}_i(\tau), \bar{p}_j(\tau)) + \Xi(\bar{\mathbf{x}}_i(\tau) + \|\bar{\mathbf{u}}_i(\tau)\|_{\mathbf{R}_i}^2. \tag{3.4.3}$$

In (3.4.3), $\Omega_{\mathrm{static}}^{i,\ell}(\bar{p}_i(\tau), x_{c\ell})$ and $\Omega_{\mathrm{dynamic}}^{i,j}(\bar{p}_i(\tau), \bar{p}_j(\tau))$ ($j \neq i$, j is considered as a moving obstacle) are on-off repulsive potential functions, which were presented in (3.3.7) and (3.3.10), $\Xi(\bar{\mathbf{x}}_i(\tau)$ was defined in (2.2.11) and represents the attractive potential which has the lowest value at the desired target, $\mathbf{x}_{i,ref}(\cdot)$ (in this particular application, the Trondheim harbor). This term also appears in the terminal cost of (3.4.2a):

$$E_i(\bar{\mathbf{x}}_i(t_k+T_p)) = \|\bar{\mathbf{x}}_i(t_k+T_p) - \mathbf{x}_{i,ref}\|_{\mathbf{P}_i}^2. \tag{3.4.4}$$

The predicted state $\bar{\mathbf{x}}_i(\tau)$ of MAS will be determined by predicted input $\bar{\mathbf{u}}_i(\tau)$ so that the cost per stage $\mathcal{L}_i^{\mathrm{OORP}}(\cdot)$ given in (3.4.2a) and illustrated in Fig. 3.4.3 is minimized. The constraint violation, i.e., $\bar{\mathbf{x}}_i(\tau) \notin \mathcal{X}_i$ and $\bar{\mathbf{u}}_i(\tau)) \notin \mathcal{U}_i$ must be penalized with a higher cost in order to prevent MAS approaching the forbidden regions.

The classical APF approach, theoretically, is able to achieve a feasible collision-free path for MAS if the underlying potential function is well-chosen. However, since the variable environment of MAS is a formulation of many repulsive potential fields which also have to be considered over the

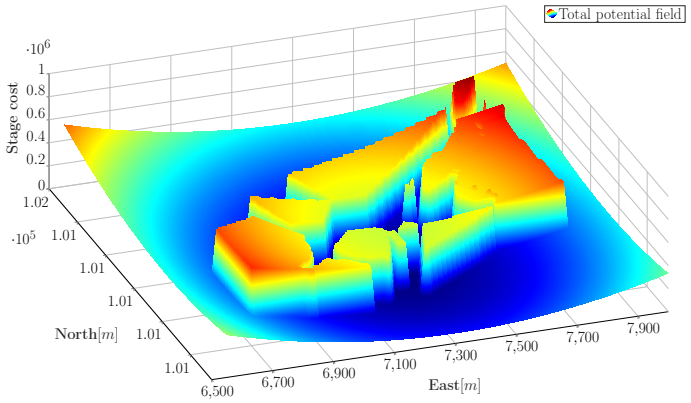


Figure 3.4.3: Cost per stage (3.4.2a) in the case in which all the repulsive potential fields are activated, i.e., the classical APF approach.

entire prediction horizon, it becomes hard to choose and interpret the parameters appearing in the RP fields. Consequently, MAS may get stuck in local minima leading to infeasibilities. The illustration of classical APF approach can be understood in general in Fig. 3.4.3, all repulsive potentials are always activated despite the position of MAS.

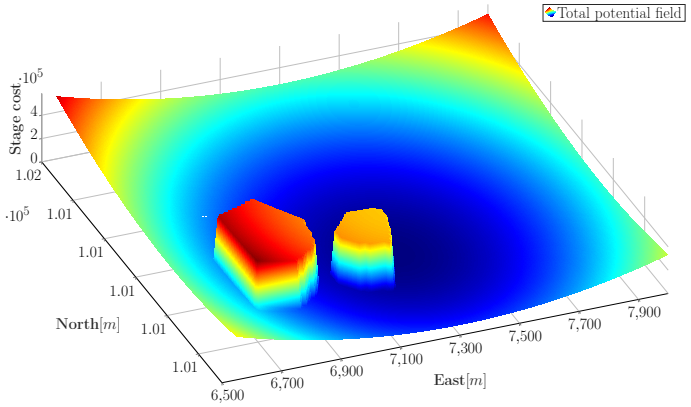


Figure 3.4.4: Cost per stage (3.4.2a) in case two repulsive potential fields activated.

Applying the on-off barrier functions in combination with repulsive potential fields as shown in Problem 3.4.2 is considered next. Therefore, the potential field changes continuously depending on the current position of MAS, i.e., APF is not a constant but a function that changes over time. Furthermore, the number of repulsive potentials representing obstacles are also diminished since they are only activated within the view range of the agent, as can be seen in Fig. 3.4.4, for example. The switching behavior modeled by the on-off barriers, depends on the design parameter β. High values of $\beta_{i\ell}$ and β_{ij} give us a speedy switch, and vice-versa, as can be seen in Fig. 3.2.2. However, arbitrarily high values should be avoided as they lead to a more challenging, prone to numerical issues, formulation.

All the information in connection with the location of the mainland, islands (i.e., static obstacles) and positions, velocities of other surface vehicles operating on the sea which are either mapped a priori, updated by GPS[7] or AIS[8], respectively.

[7] A Global Positioning System, or GPS, uses satellites orbiting the Earth to relate absolute location.

[8] The Automatic Identification System (AIS) is an automated, autonomous tracking system which is extensively used in the maritime world for the exchange of navigational information between AIS-equipped terminals.

At each sampling instance τ, the current state of i^{th} ship measured and calculated in order to yield the predicted output (i.e., state) $\bar{\mathbf{x}}_i(\tau)$, in particular, its predicted position $\bar{p}_i(\tau)$ over the prediction horizon T_p. At this point, depending on the Euclidean distance between i^{th} agent and ℓ^{th} fixed obstacle, between i^{th} agent and other agents (or moving obstacle), there will be four possible situations with regard to solving Problem 3.4.2 for each agent in order to yield the optimal control sequence:

- If both safe distances, $D_s^{i,\ell}$ and $D_d^{i,j}$, as defined in (3.3.5) and (3.3.8), are guaranteed, i.e., all obstacles are exterior of the i^{th} ship's view range ($\|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| > D_d^{i,j}$), then all the repulsive potential fields with regard to static and dynamic obstacles are disabled by on-off barrier function in cost function (3.4.2a).
- If only one safe distance, ($D_d^{i,j}$) between agent and dynamic obstacle is violated, i.e., dynamic obstacle is interior of the i^{th} ship's view range ($\|\Delta p_{i,j}^d\| \leq D_d^{i,j}$), then the dynamic repulsive potential field is activated by its associated on-off barrier function, $\overline{F}_{i,j}(\cdot)$, in cost function (3.4.2a). Depending on the violation magnitude, the value of dynamic RP will be within the range $(0,1]\Upsilon_i(\cdot) \approx \left(0, \frac{c_{1i}}{(c_{2i})^2}\right]$, i.e., $0 < \Upsilon_i(\cdot) \leq \frac{c_{1i}}{(c_{2i})^2}$, where $\frac{c_{1i}}{(c_{2i})^2}$ is maximum value of $\Upsilon_i(\cdot)$.
- If only one safe distance, ($D_s^{i,\ell}$) between agent and fixed obstacle is violated, i.e., static obstacle is interior of the i^{th} ship's view range ($\|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}$), then the static repulsive potential field is activated by its associated on-off barrier function, $\overline{F}_{i,\ell}(\cdot)$, in cost function (3.4.2a). Depending on the violation magnitude, the value of static RP will be within the range $(0,1]\Phi_\ell^f(\cdot) \approx \left(0, \frac{c_{1\ell}}{(c_{2\ell})^2}\right]$, i.e., $0 < \Phi_\ell^f(\cdot) \leq \frac{c_{1\ell}}{(c_{2\ell})^2}$, where $\frac{c_{1\ell}}{(c_{2\ell})^2}$ is maximum value of $\Phi_\ell^f(\cdot)$.
- If both safe distances, $D_s^{i,\ell}$ and $D_d^{i,j}$ are violated, i.e., all obstacles are interior of the i^{th} ship's view range ($\|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| \leq D_d^{i,j}$), then both static and dynamic RP are activated by on-off barrier functions, $\overline{F}_{i,\ell}(\cdot)$ and $\overline{F}_{i,j}(\cdot)$, in cost function (3.4.2a). Depending on the magnitude of the violation, the value of static and dynamic RP will be within the range $0 < \Phi_\ell^f(\cdot) \leq \frac{c_{1\ell}}{(c_{2\ell})^2}$ and $0 < \Upsilon_i(\cdot) \leq \frac{c_{1i}}{(c_{2i})^2}$.

To close the loop, the first component of the optimal control sequence computed over $\tau \in [t_k, t_k + T_p]$ for a measured state $\hat{\mathbf{x}}_i(t_k)$, $\bar{\mathbf{u}}_i^*(\tau; \hat{\mathbf{x}}_i(t_k))$, is applied to the nominal system of ship (3.1.1). This process will be repeated at the next sampling instance.

The details of motion planning of MAS of the proposed approach in NMPC scheme are presented in Algorithm 1 as follows:

Algorithm 1 NMPC implementation.

Require: Consider nominal model of ship (3.1.1), the set $\mathbb{S}$ of static RP of the fixed obstacles (3.1.9) and $\mathbb{M}$ of moving objects (3.3.4), the safe distance between agent and fixed obstacle $D_s^{i,\ell}$ (3.3.5), the safe distance between agent and moving obstacle $D_d^{i,j}$ (3.3.8) and the attractive potential (3.4.4).

1: $\tau \in [t, t + T_p]$;
2: Measure the current states $\mathbf{x}_i(t_k)$ of the ship and give the predicted output, in particular, its position over the prediction horizon T_p;

3: Solve the nonlinear programming problem (3.4.2) and yield the optimal control sequence $\bar{\mathbf{u}}_i(\cdot)$;

4: **if** $\|\Delta p^s_{i,\ell}\| > D^{i,\ell}_s$ and $\|\Delta p^d_{i,j}\| > D^{i,j}_d$ **then**

5: inactivate static and dynamic repulsive potential fields of ℓ^{th} fixed and and j^{th} moving obstacle (or other agent) in (3.4.2a), i.e., cost function (3.4.2a) in this case is:

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) + \int_{t_k}^{t_k+T_p} \Big[\|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i}\Big] d\tau;$$

6: **end if**

7: **if** $\|\Delta p^s_{i,\ell}\| > D^{i,\ell}_s$ and $\|\Delta p^d_{i,j}\| \leq D^{i,j}_d$ **then**

8: inactivate the ℓ^{th} static RP of fixed obstacles and activate the dynamic RP of j^{th} agent (or moving obstacle) in (3.4.2a) whose value stays within the range $(0,1]\Upsilon_i(\cdot)$, i.e., cost function (3.4.2a) in this case is:

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) + \int_{t_k}^{t_k+T_p} \Bigg[\Bigg(\sum_{i=1}^{N_a} \overline{F}_{i,j}(\|\Delta \bar{p}^d_{i,j}(\tau)\|, \beta_{ij}) \Upsilon_i(\theta_i(\bar{p}_i(\tau))) \Bigg) + \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i} \Bigg] d\tau;$$

9: **end if**

10: **if** $\|\Delta p^s_{i,\ell}\| \leq D^{i,l}_s$ and $\|\Delta p^d_{i,j}\| > D^{i,j}_d$ **then**

11: activate the ℓ^{th} static RP of fixed obstacle whose value stays within the range $(0,1]\Phi^f_\ell(\cdot)$ and inactivate the dynamic RP of j^{th} agent (or moving obstacle) in (3.4.2a), i.e., cost function (3.4.2a) in this case is:

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) + \int_{t_k}^{t_k+T_p} \Bigg[\Bigg(\sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta \bar{p}^s_{i,\ell}(\tau)\|, \beta_{i\ell}) \Phi^f_\ell(\gamma_\ell(\bar{p}_i(\tau))) \Bigg) + \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i} \Bigg] d\tau;$$

12: **end if**

13: **if** $\|\Delta p^s_{i,\ell}\| \leq D^{i,\ell}_s$ and $\|\Delta p^d_{i,j}\| \leq D^{i,j}_d$ **then**

14: activate the ℓ^{th} static RP of fixed obstacle whose value stays within the range $(0,1]\Phi^f_\ell(\cdot)$ and activate the dynamic RP of j^{th} agent (or moving obstacle) whose value stays within the range $(0,1]\Upsilon_i(\cdot)$ in (3.4.2a), i.e., cost function (3.4.2a) in this case is: $\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) +$

$$\int_{t_k}^{t_k+T_p} \Bigg[\Bigg(\sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta \bar{p}^s_{i,\ell}(\tau)\|, \beta_{i\ell}) \Phi^f_\ell(\gamma_\ell(\bar{p}_i(\tau))) \Bigg) + \Bigg(\sum_{i=1}^{N_a} \overline{F}_{i,j}(\|\Delta \bar{p}^d_{i,j}(\tau)\|, \beta_{ij}) \Upsilon_i(\theta_i(\bar{p}_i(\tau))) \Bigg) + \|\bar{\mathbf{x}}_i(\tau) - \mathbf{x}_{i,ref}(\tau)\|^2_{\mathbf{Q}_i} + \|\bar{\mathbf{u}}_i(\tau)\|^2_{\mathbf{R}_i} \Bigg] d\tau;$$

15: **end if**

16: Apply only the first sample of the optimal control sequence (i.e., $\bar{\mathbf{u}}^*_i(\tau; \mathbf{x}_i(t_k))$) to the nominal system of ship (3.1.1) over the interval $[t_k, t_k+T_p]$;

17: Continue to the next sampling instance;

18: RETURN to step 1;

3.5 Simulation results for collision avoidance using on-off repulsive potentials

This section continues considering the application example started at the beginning of this chapter. Here we combine all the earlier elements and prove the effectiveness of the proposed algorithm for the safe navigation of a ship in the Trondheim fjord. We use the USV model (3.1.1) with the following parameters:

$$M_i = \begin{bmatrix} 25.8 & 0 & 0 \\ 0 & 33.8 & 1.0115 \\ 0 & 1.0115 & 2.76 \end{bmatrix}, D_i = \begin{bmatrix} 0.9257 & 0 & 0 \\ 0 & 2.8909 & -0.2601 \\ 0 & -0.2601 & 0.5 \end{bmatrix}.$$

For simplicity, the Coriolis matrix, wind and wave forces are neglected.

The hydrodynamic parameters, $Y_\delta = -0.2$ and $N_\delta = 1$ lead to control input $\mathbf{u}_i = \begin{bmatrix} \mathcal{T}_{u_i} & Y_\delta \delta_i & N_\delta \delta_i \end{bmatrix}^\top$ $= \begin{bmatrix} \mathcal{T}_{u_i} & -0.2\delta_i & \delta_i \end{bmatrix}^\top = \begin{bmatrix} \mathcal{T}_{u_i} & -0.2\mathcal{T}_{r_i} & \mathcal{T}_{r_i} \end{bmatrix}^\top$.

We consider constraints on the actuation force $\mathcal{T}_{u_i} \in [-2, 2]^\top$ [N], on the yaw moment $\mathcal{T}_{r_i} \in [-1.5, 1.5]^\top$ [Nm]. The velocity of the ship is chosen randomly, particularly, the surge velocity $u_i \in [-0.5, 0.5]$ [m/s], the sway velocity $v_i \in [-0.1, 0.1]$ [m/s] and yaw rate $r_i \in [-0.2, 0.2]$ [rad/s].

The parameters used in (3.3.5) and (3.3.8) are: the ship's view range $\Gamma_i = 20$ $[m]$ and the expansion coefficient is $\varepsilon \in [1.2, 1.8]$ depending on the geometric shape of each static obstacle.

The steepness of the curve, i.e., design parameter used as in (3.3.6) and (3.3.9) is $\beta = 1.2$.

Relevant values for the other parameters of the NMPC optimization problem in (3.4.2a) are weighting matrices: $\mathbf{Q}_i = 0.1\mathbf{I}_6$, $\mathbf{R}_i = 0.1\mathbf{I}_2$, $\mathbf{P}_i = [0.5\mathbf{I}_2 \quad \mathbf{0}_2 \quad \mathbf{0}_2; \mathbf{0}_2 \quad \mathbf{I}_2 \quad \mathbf{0}_2; \mathbf{0}_2 \quad \mathbf{0}_2 \quad \mathbf{I}_2]$.

Finally, the number of cells considered from partitioning the map, as illustrated in Fig. 3.1.3b is $N_{\text{obs}} = 22$.

Simulations are realized by using FORCE PRO via Matlab R2016a on a computer with the following configuration: Intel Core i7-4790CPU, 3.60GHz, 8GB RAM.

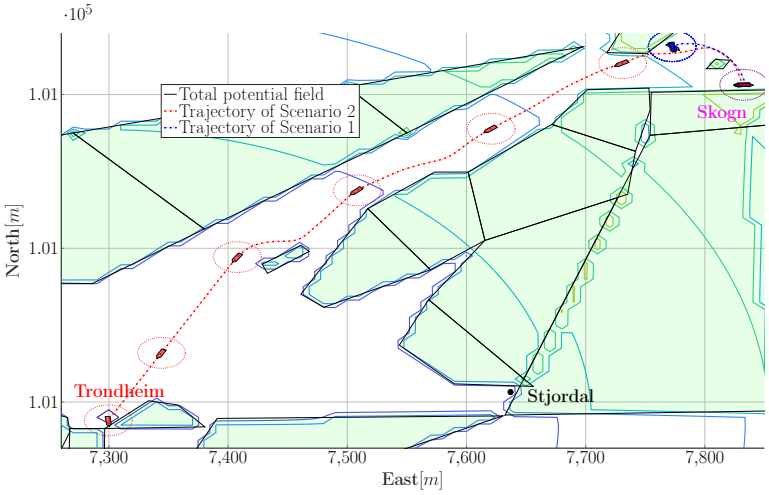
The travel from Orkanger and Skogn to Trondheim harbor while avoiding collision with static and moving obstacles is considered under the following scenarios:

- *Scenario 1:* consider one ship (i.e., $N_a = 1$), use the classical approach where the total potential field appearing in the NMPC's cost is active for the entire simulation;
- *Scenario 2:* consider one ship (i.e., $N_a = 1$), use Algorithm 1 where the constraints are activated when inside the view range of the agent;
- *Scenario 3:* consider one ship (i.e., $N_a = 1$), use the classical mixed-integer programming (MIP) approach for taking into account the collision avoidance constraints. A comparison with Algorithm 1 in terms of obstacle avoidance, feasibility or solver time is performed.
- *Scenario 4:* consider one ship (i.e., $N_a = 1$) and the three moving obstacles, use Algorithm 1 under the COLGRES rules 8 and 16 given in Annex A. In this case, only the ship takes action to avoid colliding with the three dynamic obstacles.

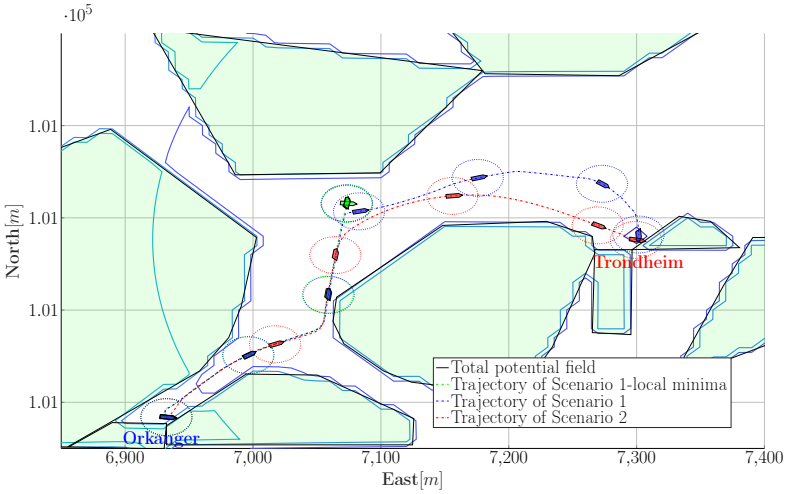
- *Scenario 5:* consider two ships (i.e., $N_a = 2$) and COLGRES rules 8, 13 and 16, make a comparison between using Algorithm 1 and the Breivik's approach in [Breivik et al., 2017].

3.5.1 Comparison of Scenario 1 (classical APF) and 2 (proposed APF)

Fig. 3.5.1a illustrates the actual motion of the agent under *Scenario 1* (blue dash dotted line) which gets stuck in a local minimum because that is a critical area surrounded by many repulsive potentials and their interaction creates a null-potential field. This is the main drawback of the classical potential field approach. This is not the case in *Scenario 2* (red dash dotted line), which is also represented in Fig. 3.5.1a and it can be observed that the ship converges towards the Trondheim harbor (its destination) without getting stuck in a local minima, this is due to the fact that the number of non-convex constraints (coming from the repulsive potential) is reduced since they are not in the view range of the agent. Notice that the circle around the ship represents its view range.



(a) Skogn to Trondheim harbor.



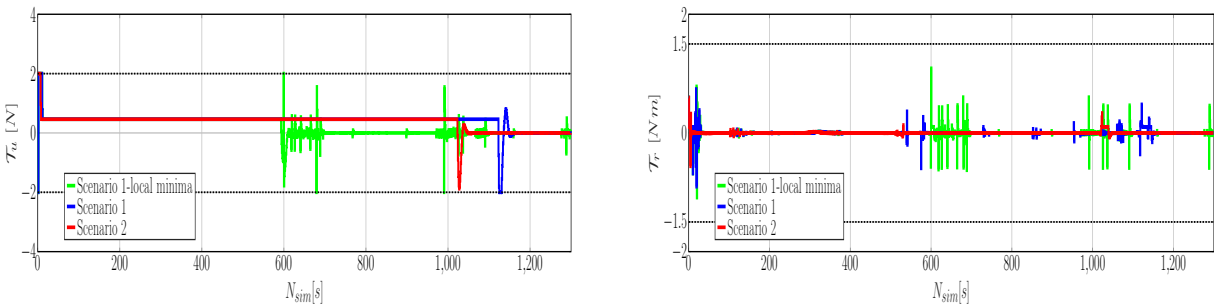
(b) Orkanger to Trondheim harbor.

Figure 3.5.1: Comparison of trajectories between *Scenarios 1 and 2*.

Table 3.5.1: Performance criteria of *Scenarios 1 and 2* in case the ship travel from Orkanger to Trondheim harbor.

Fig. 3.5.1b	*Scenario 1* (no local-minima)	*Scenario 2*
Prediction horizon (T_p) [s]	20	20
Sampling time [s]	1	1
Simulation number	1300	1300
Number of steps	1163	1081
Average of CPU [s]	38.61	36.27
Length of trajectory [m]	562.3	512

Fig. 3.5.1b presents a comparison between the agent's trajectory under Scenario 1 and Scenario 2 which go from Orkanger to Trondheim harbor. Similar to the previous case, in this case, the actual motion of ship under Scenario 1 is also trapped (green dash dotted line), the most straightforward solution is to reduce the strength or effect range of static repulsive potential fields in order to change their configuration. As a result, the ship can converge towards Trondheim harbor (blue dash dotted line). However, evaluation criteria such as smoothness (control input in Fig. 3.5.2, length of trajectory, number of steps, or computation time show that the proposed algorithm is superior to the classical approach. Table 3.5.1 gives some details of the two approaches. The proposed method has better results for all aspects compared with the classical approach.



(a) Surge thrust $\mathcal{T}_u$.

(b) Rudder deflection $\mathcal{T}_r$.

Figure 3.5.2: Control inputs under *Scenarios 1 and 2* of Fig. 3.5.1b (Orkanger to Trondheim harbor).

It is worth mentioning in Fig. 3.5.2 that the green line implies the local minima case of the ship in *Scenario 1* with a strong oscillation of control inputs. The blue line (*Scenario 1*-no local minima) even through it has fewer oscillations, it is still not as good as the red line which illustrates control inputs of *Scenario 2*.

3.5.2 Comparison of Scenario 2 (proposed APF) and 3 (MIP technique)

To simplify the MIP approach, we consider in *Scenario 3* the linear dynamical system representing an unmanned surface vehicle which was presented in (2.4.1) of Chapter 2.

As before, the agent will travel from Orkanger to Trondheim harbor, taking into account the static obstacles.

MPC with MIP can deal with the obstacle avoidance by explicitly considering the non-convex constraints via a mixed-integer representation as presented in Chapter 2. The CPLEX solver is used to solve the resulting quadratic mixed-integer optimization problem.

The parameters of both approaches are the same such as the weighting matrices, sampling time, and the constraints on inputs and states. See details in Section 2.4 of Chapter 2.

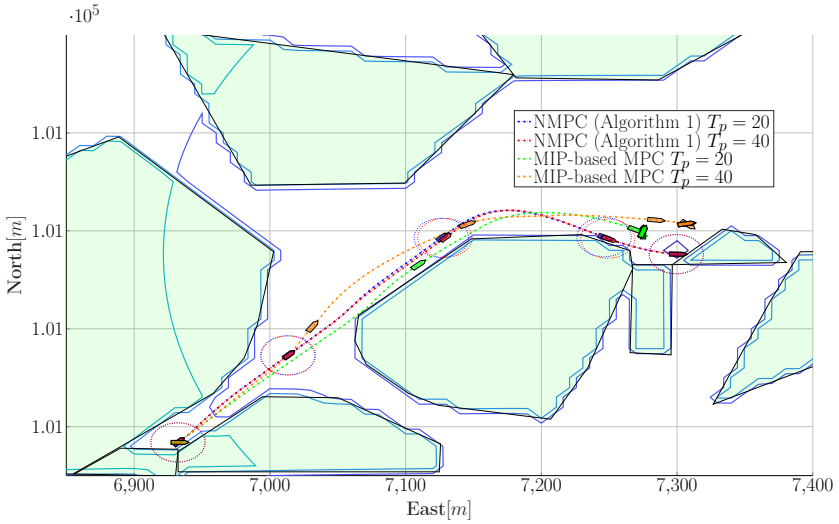


Figure 3.5.3: Comparison of trajectories between *Scenario 2 and 3* (Orkanger to Trondheim harbor).

Table 3.5.2: Performance criteria of *Scenario 2 and 3* in case the ship travel from Orkanger to Trondheim harbor.

Fig. 3.5.3	MIP-based MPC		NMPC (Algorithm 1)	
Prediction horizon (T_p)	20	40	20	40
Sampling time $[s]$	1	1	1	1
Simulation number	400	400	400	400
CPU time [s]	30.8	74	2.4	7.2
Length of trajectory $[m]$	471.5	477.4	477.6	477.2
Number of steps	infeasibility	infeasibility	141	140

Fig. 3.5.3 shows the two trajectories of MIP with various prediction horizons (T_p). Green and the orange dash dotted line representing MIP-based MPC with $T_p = 20$ and $T_p = 40$, respectively, show the danger of having infeasibilities in a complex workspace as Trondheim fjord. The sudden termination of the optimization (MIP is solved through a branch-and-bound algorithm which may get stuck) prevents the ship from reaching Trondheim harbor. Trajectories using NMPC with Algorithm 1 (blue and red dash dotted line) are not affected by the variation of prediction horizon and can converge to the target point, and their trajectories have a certain distance to the mainland. Data in Table 3.5.2 suggests that it is superior to MIP regarding some important criteria, local minima, solver time, and the number of steps in which the agent reaches the desired destination (Trondheim harbor).

3.5.3 Scenario 4: Consider the proposed algorithm under the COLREGS rules

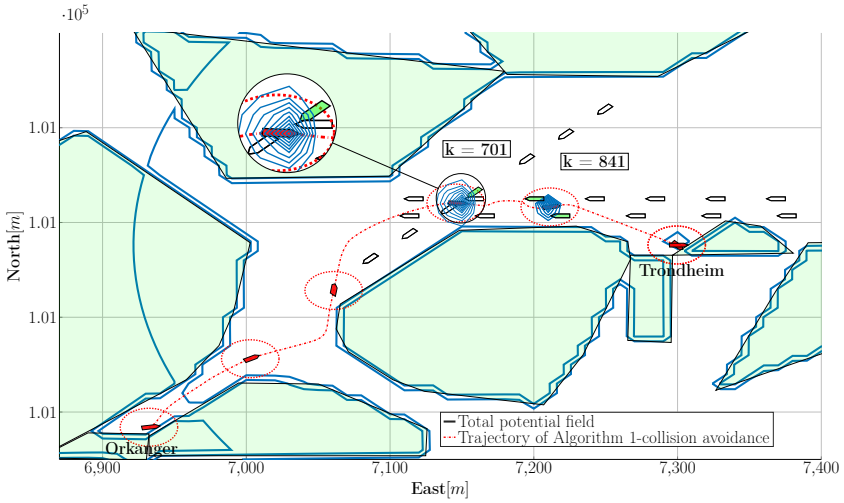


Figure 3.5.4: Ship's collision avoidance complies COLREGS rules 8 and 16 in *Scenario 3* (Orkanger to Trondheim harbor).

Fig. 3.5.4 shows the effectiveness of the algorithm proposed. The trajectory generated following *Scenario 3* can avoid both fixed obstacles and three moving obstacles (other ships). Notice that moving obstacles are drawn white since they do not belong to the ship's view range. When moving obstacles are inside the ship's view range, they are drawn in green, and the repulsive potential of the vessel is activated, as can be seen at time instances $k = 701$ and $k = 841$, to implement collision avoidance action.

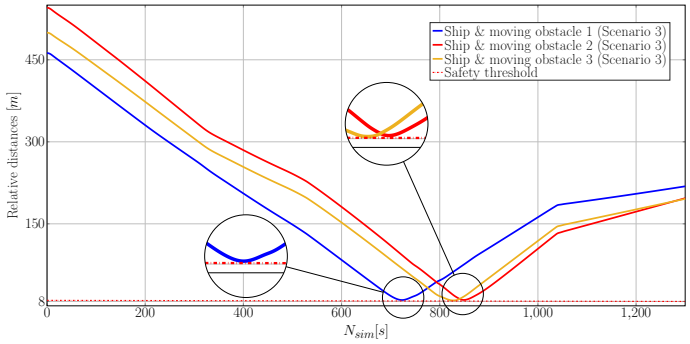
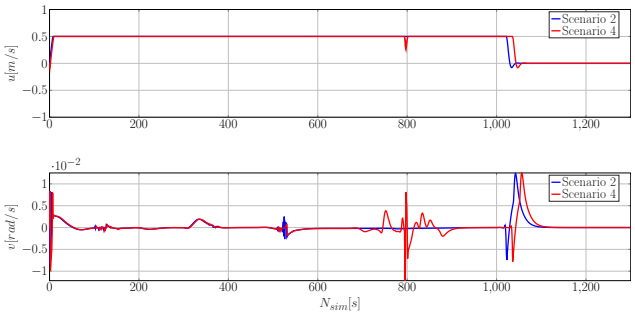


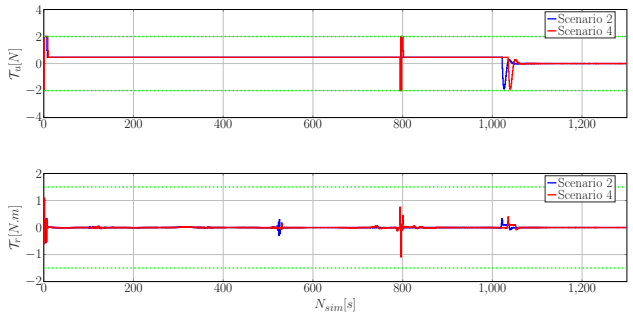
Figure 3.5.5: Relative distances between ship and moving obstacles in *Scenario 3* (Fig. 3.5.4).

Note that in Fig. 3.5.1a, Fig. 3.5.1b and Fig. 3.5.4, we also illustrate the repulsive and attractive potentials projected in 2D (the repulsive potentials appear around the forbidden region preventing ship's collision with the fixed and moving obstacles and circles around the destination harbor represent the attractive potential).

Fig. 3.5.5 shows the relative distances between the ship and the three moving obstacles in *Scenario 3*. The relative gaps between them depend on strength (λ) and effect range (γ) of the ship's repulsive potential at time instance $k = 701$ and $k = 841$ as illustrated in Fig. 3.5.4. In this case, we assume that their safe thresholds are not less than 8 [m], λ and γ are 1000 and 1, respectively.



(a) Comparison of surge and sway velocities in *Scenario 2 and 4*.



(b) Comparison of control inputs in *Scenario 2 and 4*.

Figure 3.5.6: Surge and sway velocities, and control inputs under *Scenario 2 and 4* of the trajectories presented in Fig. 3.5.1b and 3.5.4.

Observe in Fig. 3.5.6a, Fig. 3.5.7 and Fig. 3.5.6b, that in the range of 700 to 900 seconds, surge, sway, yaw rate and control input of the proposed algorithm are changed in order to avoid the three moving obstacles that enter the ship's view range. Consequently, by following rules 8 and 16 of COLREGS, the vessel has taken substantial actions to avoid a collision, and the change of course angle is significant enough for the approaching boat.

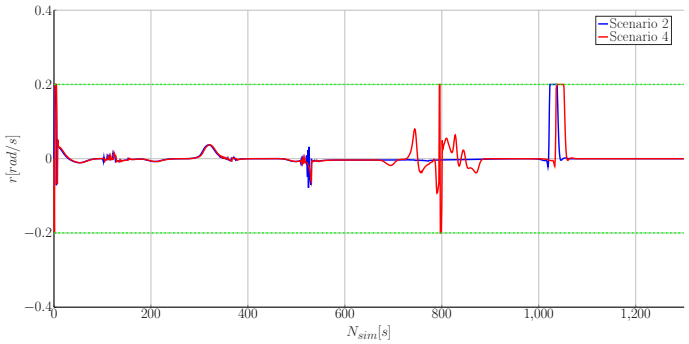


Figure 3.5.7: Yaw rate in *Scenario 2 and 4* (of the trajectories presented in Fig. 3.5.1b and 3.5.4).

Table 3.5.3: Performance criteria of *Scenario 2 and 4* in case the ship travel from Orkanger to Trondheim harbor complying with the COLREGS rules.

	Scenario 2 (Fig. 3.5.1b) (no local-minima)	*Scenario 4* (Fig. 3.5.4)
Prediction horizon (T_p) [s]	20	20
Sampling time [s]	1	1
Simulation number	1300	1300
Number of steps	1081	1112
CPU time [s]	36.27	40.69
Length of trajectory [m]	512	517.1

3.5.4 Scenario 5: Comparison of the proposed algorithm and another strategy

This scenario presents a comparison of the proposed method in this paper and the approach of [Breivik et al., 2017] in the context of COLREGS compliance, in particular, rule 8, 13, and 16. The strategy of [Breivik et al., 2017] is first to guarantee that the safe regions (disc-shaped) of the dynamic or static obstacles have to be larger or equal to their relative distances with respect to the agent which they are considering. A penalty term is then proposed for ensuring that any alteration of course and speed must be large enough to clear the approaching ships. It is worth noting that only ship 1 takes action to avoid colliding with ship 2 since ship 1 is overtaking ship 2 (i.e., ship 2 is being overtaken by ship 1), ship 2 must hold the course and speed until ship 1 is past and well clear.

Consider two ships as presented in (3.1.1), defined by a time-varying center position and its safe region (disc-shaped) $O_{mi}(t) = (p_{mi}(t), r_{mi}) \in \mathbb{R}^2 \times \mathbb{R}^+$, where $p_{mi} = [x_i \ y_i]^\top$ notes the position state and r_{mi} is its radius.

For guaranteeing collision avoidance among two ships, their relative distance between ship i and j is described as:

$$h_{mi} = r_{mi}^2 - \|p_i - p_j\|_2^2 \leq 0. \tag{3.5.1}$$

(3.5.1) means that, the overtaking ship i is not able to enter the safe region of the ship being overtaken j with safe radius $r_{mj} = 8$ [m].

The penalty term combines a square and an exponential term which is inspired by the normal distribution:

$$q(\zeta; a, b) = a\zeta^2 + \left((1 - exp(\tfrac{-\zeta^2}{b})\right), \tag{3.5.2}$$

where $a, b > 0$ control the shape of the function. Hence, the functions employed to penalize the yaw rate and and surge velocity can be described by the expressions:

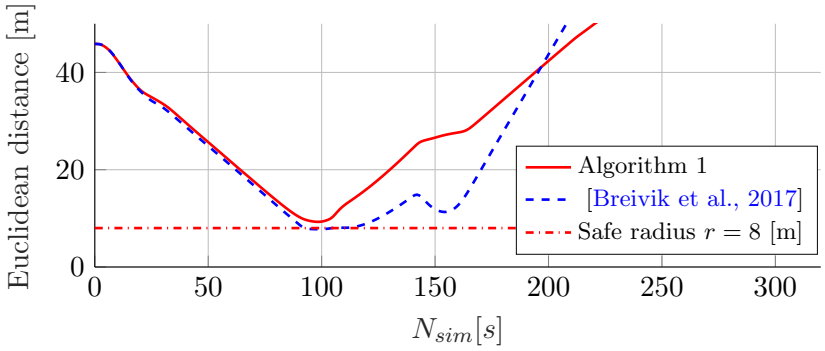
$$q_{r_i}(r_i; r_{imax}) = \frac{100}{q_{r_i}(r_{imax}; a_{r_i}, b_{r_i})} q_{r_i}(r_i; a_{r_i}, b_{r_i}) \tag{3.5.3a}$$

$$q_{u_i}(u_i; u_{imax}) = \frac{100}{q_{u_i}(u_{imax}; a_{u_i}, b_{u_i})} q_{u_i}(u_i; a_{u_i}, b_{u_i}), \tag{3.5.3b}$$

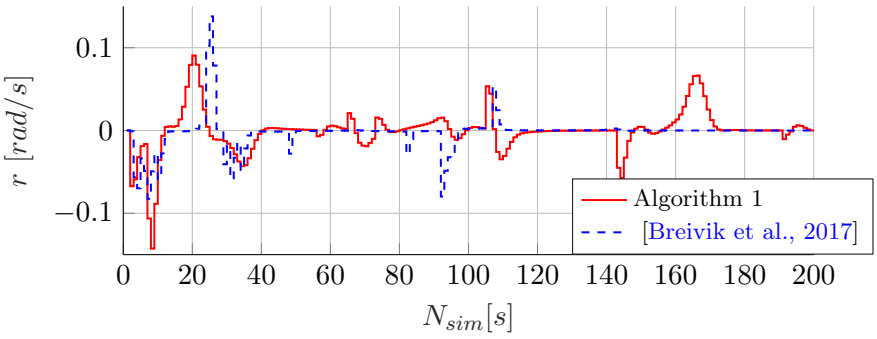
where the parameters a_{r_i} and b_{r_i}, a_{u_i} and b_{u_i} define the shape of the yaw rate $q_{r_i}(r_i)$ and surge velocity $q_{u_i}(u_i)$ penalty terms.

Therefore, (3.5.1) is considered as an explicit constraint and (3.5.3) is penalized in the cost function of the NMPC scheme to obtain the collision-free path of ships. For details, refer to [Breivik et al., 2017].

The method proposed by [Breivik et al., 2017] penalizes the slow alterations of yaw rate and surge velocity of ship 1, as can be seen in Fig. 3.5.8b. At time step $k = 82$, the yaw rate (drawn in dashed blue) is altered rapidly. The decrease in yaw rate implies that ship 1 turns starboard. The yaw rate in red color (Algorithm 1) changes gradually, and changes earlier than the one in dashed blue at time step $k = 78$ (an increase in the yaw rate means that ship 1 turns port). Also, the Euclidean distance between ship 1 and 2, which is implemented by Algorithm 1 is greater than the safe threshold whereas the relative distance between the two ships of Breivik's approach approaches the safe limit in a critical moment, as depicted in Fig. 3.5.8a.



(a) Euclidean distances.



(b) Yaw rates.

Figure 3.5.8: Comparison of Euclidean distances and yaw rates of two approaches.

In Fig. 3.5.9, we see the motion of two USVs while traveling from Orkanger to Trondheim harbor with COLREGS. The ship 1 (in blue) which is implemented by [Breivik et al., 2017] is trapped at time step $k = 150$ since it was cornered into the deadlock region where ship 1 both guarantees the safe distance with ship 2 and avoids the ground. Meanwhile, ship 1 (in red), for which we use our proposed potential field approach for moving obstacle, reaches the Trondheim harbor in a timely manner.

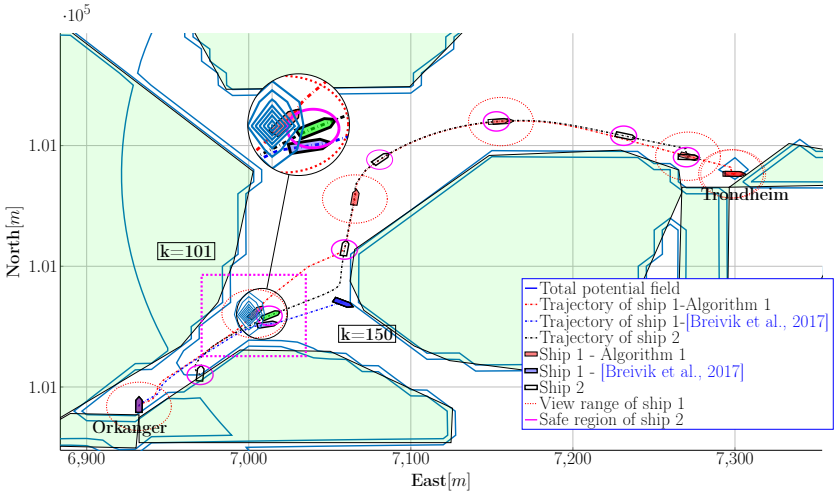


Figure 3.5.9: Motion planning by two approaches for two USVs while traveling from Orkanger to Trondheim harbor with COLREGS compliance - rules 8, 13 and 16.

3.6 Conclusions

In this chapter, we first recalled the potential field, as the total of the repulsive and attractive potentials. Differing from Chapter 2, the dynamic repulsive potentials are parametrized by the agents' current position. A method was proposed to deal with the local minima through the on-off barrier function, which is used to weight repulsive potentials for fixed and moving obstacles. This combination has made the magnitude of the total potential field change continuously, i.e., the null-potential field (if any) where the total of repulsive and attractive forces is zero will not survive. This combined function was added to the cost function of an NMPC scheme as a penalty term. Repulsive potential constructions were activated only in the view range of the agent. The focus was on a real application of motion planning and safe navigation for ships in the Trondheim fjord. Proof of concept, illustrations, numerical simulations, and comparison results over a real benchmark prove the efficiency of the proposed algorithm.

However, as observed in the illustrative examples, we are able to recognize that the trajectory generated by the proposed method, even if better than the classical potential field approach, it is still not realistic because the feasible path for agent was only generated by considering the initial and final point. Such a simple plan cannot guarantee the shortest collision-free path for MAS from the source to destination. Therefore, future works will investigate how the agents, particularly, vessels, can converge toward the desired destination by the shortest path. Also, we will consider other COLREGS rules for this benchmark within a distributed control framework for MAS. Of further interest is to ensure the necessary connectivity distance for information exchange among the agents in the presence of the external disturbances such that the robustness of the proposed algorithm is guaranteed.

Chapter 4

Distributed connectivity maintenance of MAS in the presence of environmental disturbances

The increased interest in autonomous robots and their use in various applications paves the way for emerging topics like motion planning and control. This plays a vital role in creating autonomous systems to execute complex tasks for navigation in challenging environments [Ge et al., 2018]. In particular, multiple USVs can be used to perform a specific mission with higher performance and lower costs than a single agent such as search and rescue, coverage tasks, and patrolling. Indeed, path following for multiple surface vessels has been both an interesting and a challenging control problem that has attracted attention from the control community in recent years. However, there are some critical issues which need to be considered and solved for achieving the best performance: generating collision-free paths with respect to static (e.g., islands, shoreline or ship anchoring) and moving obstacles (e.g., other ships) while simultaneously complying with the COLREGS rules [Commandant, 1999] for both manned and unmanned surface vehicles; guaranteeing the necessary connectivity distance for information exchange among the agents in the course of performing tasks and rejecting the external disturbances from the ocean which may significantly affect the nominal performance of the multiple USVs.

The non-convex constraints describing obstacle/collision avoidance have been added to the cost of the optimization-based control, as detailed in the previous chapters. This method, the so-called *on-off repulsive potential*, can deal with local minima when the APF approach is employed as an essential tool for motion planning of MAS in an NMPC framework. This proposal showed its efficiency as well as high applicability over a benchmark using real numerical data for USVs safe navigation. Nevertheless, the globally shortest path between the initial and target points have not yet been achieved since the proposed method is only concerned with the obstacle and collision avoidance. Depending on the operating space of the USV, two well-known approaches can be used in order to obtain the shortest route, such as cell decomposition and randomized sampling algorithms. *Cell decomposition* is the method in which the non-convex environment is partitioned into a set of simple cells. After that, the A* algorithm [Yang et al., 2015, Chao et al., 2017] is used to compute the cost of the current nodes to reach other nodes. These nodes are evaluated by applying heuristic information. Finally, the node with the smallest cost will be taken as the expansion node to achieve a collision-free path between the initial and the goal configuration of the vessel. *Randomized sampling algorithms* (i.e., graph-based methods), with two well-known algorithms, are PRM (Probabilistic Road Map) [Kneebone et al., 2006, Altinoz

et al., 2017] and RRT (Rapidly-Exploring Random Trees) [Chen et al., 2018c, Li et al., 2018]. Their basic concepts are introduced in Chapter 1, and are the two most interesting methods to obtain the shortest collision-free path for multiple USVs. In addition, COLREGS-compliance for safe maritime navigation should also be addressed. There are various approaches for this problem:

- Evolutionary algorithms [Szlapczynski, 2011] were employed to search the optimal paths set for the ships involved in the head-on situation;
- A Velocity Obstacles (VO) algorithm was proposed by [Kuwata et al., 2011] for motion planning compliant with the COLREGS rules. This approach is capable of maneuvering in any specified direction, but may not sufficiently capture the movements of underactuated systems;
- [Naeem et al., 2016] used an APF approach to obey the COLREGS rules. However, instead of considering the moving obstacles directly, the fixed virtual objects are arranged in places where the collision will occur;
- An MPC scheme was also presented in [Johansen et al., 2016], using binary variables to choose scenarios with multi-moving obstacles with erratic motion, [Breivik et al., 2017] penalized normal distribution in the cost function in the NMPC scheme, [Abdelaal et al., 2018] presented a method using slack variables to change the vessels' direction for specific situations. These approaches consider commonly disc-like approximations for the static or dynamic obstacles, which might be too conservative, w.r.t the conditions in a real environment. Moreover, usually, only one vessel is considered instead of multiple USVs when using these techniques.

Another challenging issue in motion planning for multi-agent systems is the connectivity maintenance. The purpose of this preservation is to ensure that the information exchange can be reliably done by forwarding message among agents while implementing and allocating tasks. Some applications can be found in a variety of practical tasks, such as search and rescue, surveillance, cooperative transportation by autonomous vehicles on sea or terrain, attitude alignment of clusters of satellites, air traffic management system, etc. Related to this problem, there are some recent works for always keeping agents within a connectivity distance. In general, we may classify them in two directions: i) imposing hard constraints based on the Euclidean distance among the agents, which must be less than their sensing radius as in [Filotheou et al., 2018]. This condition is handled in a decentralized NMPC framework. However, this approach is too strict and can lead to infeasibility in complex cases; ii) using the potential-field function in several ways to ensure that the agents operating in a specified range preserve the connectivity through information exchange, as presented in [Zavlanos and Pappas, 2007, Wang et al., 2012, Fang et al., 2017] but the researches are limited to single (or double)-integrator agents.

We underline that it is necessary to propose robust controllers for multi-agent systems to deal with uncertainties [Soloperto et al., 2019]. Tube-based MPC is considered as one of the most popular approaches for coping with perturbations [Mayne et al., 2005, Ke et al., 2018]. The general idea is to maintain the actual state within a safety region along the optimal state trajectory [Prodan et al., 2011]. Another approach is to design a disturbance observer to reject the disturbances leading to an improved system robustness [Yu et al., 2018a, Wu et al., 2019]. The authors in [Yang and Zheng, 2014] propose a nonlinear observer within an NMPC scheme

for a Static Var Compensator[1] system to reject arbitrary disturbances relative degree from its output channels (used for shunt compensation to maintain bus voltage magnitude). [Yu et al., 2018b] presents a compound disturbance-observer-based MPC scheme applied for a wheeled mobile robot which aims at compensating the slowly varying disturbances affecting the control inputs. A similar project for trajectory tracking of a small helicopter is considered in [Liu et al., 2011] were the disturbance observer provides the system with robustness against constant wind gusts. Furthermore, there are many works related to disturbance rejection for USVs using observer-based robust control. The authors in [Do, 2010, Yang et al., 2014, Wang et al., 2018] employ a nonlinear observer within a backstepping technique to design a trajectory tracking robust controller of the underactuated ship. However, the constraints coming from physical limitations are not considered. To the best of the author's knowledge, there are very few studies using disturbance observer-based MPC to ensure the robustness of a surface vehicle in case of uncertainties. [Liu et al., 2017, Abdelaal et al., 2018] present obstacle and collision avoidance in the presence of external disturbances while tracking a reference path, but they only take one ship into account which operates in a simple environment. For that reason, the practical application of safe navigation in a complex coastal environment for multi-surface vehicles under uncertainties remains an open problem.

Motivated by all the observations above and the results of Chapters 2 and 3, the present chapter introduces enhancements in the distributed motion planning for safe navigation of multiple surface vehicles in the presence of external disturbance in the Trondheim fjord complying with the COLREGS rules. Specifically, the contributions of this chapter are:

1. generates a LOS (Line-of-Sight) guidance system via a graph-based method, e.g., through the RRT* (optimal Rapidly-Exploring Random Tree) algorithm;

2. considers on-off barrier functions which guarantee the necessary connectivity distance for information exchange among the agents as well as activates the associated repulsive potential for static and dynamic obstacles;

3. considers a NDO (Nonlinear Disturbance Observer) to reject the disturbances from the ocean that may lead to undesirable performance for the ships;

4. integrate the above ingredients in a distributed NDO-NMPC - based algorithm with a threefold purpose: i) track the RRT*- based feasible path through LOS guidance system, ii) activate the constraints in the view range[2] of the agent for on-line collision avoidance complying with the COLREGs rules and iii) exchange information for connectivity maintenance;

5. validation of the proposed algorithm through simulations over a real benchmark for the safe navigation of ships in the Trondheim fjord.

This chapter is organized as follows: Section 4.1 presents the graph-based LOS guidance system. Section 4.2 presents the distributed motion planning algorithm for connectivity maintenance and COLREGS compliance in the presence of external disturbances. Section 4.3 shows the simulation results over a real benchmark. Section 4.4 draws the conclusions and presents the future work.

[1]The Static Var Compensator (SVC) is a device of the Flexible AC Transmission Systems (FACTS) family using power electronics to control power flow on power grids.

[2]As required by COLREGS, all ships shall maintain a proper radar lookout (has a view of up to several kilometers if there are no physical obstructions) to obtain early warning of risk of collision.

4.1 Path generation with LOS guidance based on randomized sampling algorithms

LOS guidance for path following is often used in the vessel control practice, [Pettersen and Lefeber, 2001]. In particular, the reference yaw angle can be obtained from the set of waypoints. The result is global asymptotic stabilization of the heading and cross-track error of the ship. Therefore, we can recognize that the set of waypoints has a crucial role concerning the establishment of the LOS guidance system. The collection of these points has to be determined from the a priori known environment (e.g., islands, shorelines or anchored ships representing fixed obstacles) so that a path generated for USVs from the waypoints is collision-free. Of course, we can choose any collision-free way from the starting to the target point but this does not guarantee the shortest path. Hence, randomized sampling algorithm is chosen as having a reasonable chance to obtain a near-optimal solution.

4.1.1 Graph-based methods

Table 4.1.1: Summary of advantages and disadvantages of sampling-based algorithms.

Algorithm	*Probabilistic Completeness*[3]	*Asymptotic Optimality*[4]	*Monotone Convergence*[5]
PRM	Yes	No	Yes
k-sPRM	Conditional	No	No
PRM*	Yes	Yes	No
RRT	Yes	No	Yes
RRT*	Yes	Yes	Yes
k-RRT*	Yes	Yes	Yes

Arguably, the most famous sampling-based algorithms until now include PRM and RRT. Although the idea of joining points sampled randomly from the state space is primary in both methods, there are still significant differences in the way that they construct a graph connecting these points [Karaman and Frazzoli, 2011]. The PRM algorithm and its variants (PRM*, k-sPRM) are multiple-query approaches that first construct a roadmap (i.e., graph), which stands for a rich set of feasible paths. Next, answering queries by computing the shortest path is obtained by using the Dijkstra algorithm to connect the initial state with a final state through the roadmap. Meanwhile, the RRT algorithm and its variants (RRT*, k-RRT*), is an approach built on ideas of optimal control theory to explore a tree with many branches (i.e., feasible paths). The collision-free paths rooted at the initial state grow incrementally and the algorithm returns a result as one of the collision-free paths reaches the goal region. RRT algorithm tries to find a feasible solution as quickly as possible and is terminated after seeing the first reasonable solution. The remaining computation time is employed to optimize the trajectory concerning a cost function.

[3]Probabilistic completeness means that if a path exists, the probability of finding the path is '1', as the number of iterations goes to infinity [Moon and Chung, 2014].

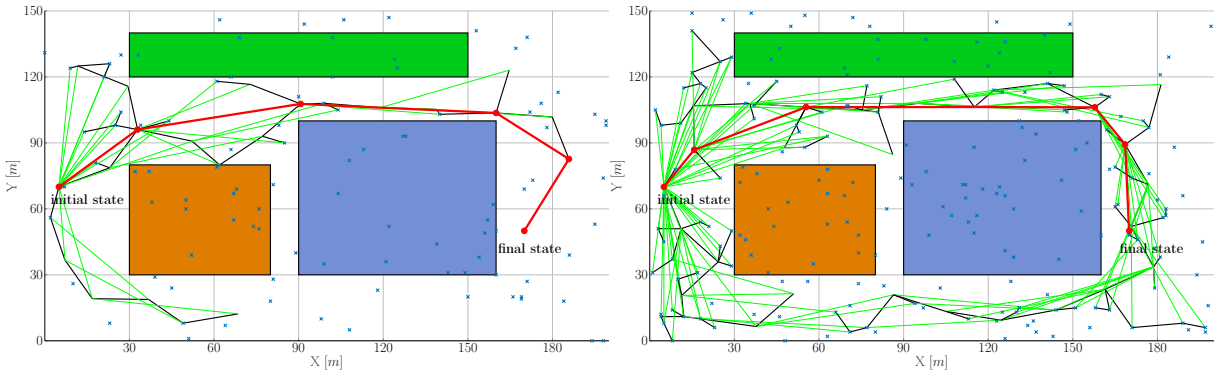
[4]The asymptotic optimality is defined as almost-sure convergence to optimal paths.

[5]The monotone convergence theorem states that if a sequence is increasing and bounded above by a supremum, then the sequence will converge to the supremum. Also, if a sequence is decreasing and is bounded below by an infimum, it will converge to the infimum.

Table. 4.1.1 presents the advantages and disadvantages of PRM and RRT as well as their variants in some criteria. RRT* and k-RRT* approaches are the most prominent among the sampling-based algorithms.

Usually for the control part for USVs is easier to have a priori defined a path which can be then be tracked on-line. In here, RRT*, with an asymptotic optimality property (i.e., almost-surely convergence toward an optimal solution) is applied for generating an off-line shortest collision-free path within the a priori known workspace (e.g., the initial state and the set of fixed obstacles). The general idea of the RRT* algorithm is to initialize a tree $\mathscr{T}$ from an initial vertex (x_{init}) and to bias growth towards unexplored regions of the state space randomly. In the process of exploring the tree, shortest paths are generated while verifying the collision-free condition involving the fixed obstacles after each iteration until reaching a new state close enough to the desired target. The feasible path is established based on the connection of shortest paths connecting among the waypoints. The RRT* algorithm is described in detail in Appendix B.

Example 4.1.1. Consider the operating space in the presence of three static obstacles a priori.



(a) The collision-free path generated from 100 randomized samples.

(b) The collision-free path generated from 200 randomized samples.

Figure 4.1.1: RRT* algorithm with the different randomized samples.

A feasible path without collision from the initial to final state generated by RRT* with the different randomized sample is illustrated in Fig. 4.1.1. It is worth noting that the determination of step size used in the forward simulation of the exploring-tree procedure is significant. If the obstacles are located far away and the system's dynamics are relatively simple (e.g., double integrator dynamics), then a large step size can be used. With a substantial number of ran-

Table 4.1.2: The comparison of RRT* with the different randomized samples.

RRT* algorithm	100 randomized samples (Fig. 4.1.1a)	200 randomized samples (Fig. 4.1.1b)
Time step [s]	20	20
CPU time [s]	11.79	25.92
Number of way-points	6	6
Length of trajectory [m]	236.08	225.87

domized samples the chances of obtaining a collision-free path generated from RRT* which is also shorter are increased. However, we must accept a trade-off between path shortness and the computation time as can be seen in Table. 4.1.2.

4.1.2 LOS guidance system

The LOS guidance principle embodies an intuitive understanding of the behavior of a ship and the action of a helmsman. An essential feature of the LOS guidance system is to be able to be applied with commercial autopilots by replacing the set of original course (i.e., heading) angle references[6] being fed to the controller by the desired course angles, which are yielded by the parameterization based on the error of the current position of ships and the group of the shortest collision-free paths generated from RRT*. Therefore, this guidance system can be computed in real-time and has an instant practical application.

In order to define the LOS guidance system, we first determine the LOS position, there are three ways to find it: i) the entire collision-free path for the ship to follow is composed of a set of way-points obtained, the LOS position is simply selected as the current way-point, denoted (x_p, y_p); ii) the path reference is also composed of a collection of waypoints, the LOS position is at the closest (w.r.t the next way-point in the sequence) intersection point obtained by intersecting the circle (with the radius expressed as a distance of two or three-times ship lengths) enclosing the ship's current position and the segment connecting the previous (x_{p-1}, y_{p-1}) and current waypoint, and iii) the feasible collision-free path is a parametrized geometrical path, this approach attracts attention because of the ability to guarantee global asymptotic stability (i.e., convergence toward the path) [Breivik, 2003, Pavlov et al., 2009, Oh and Sun, 2010], and is used to obtain the LOS guidance system in this manuscript.

The parametrization variable denoted $\alpha \in \mathbb{R}$ is a scalar. The path reference of the ships is a set of points, $p_d(\alpha) = [x_d(\alpha), y_d(\alpha)]^\top \in \mathbb{R}^2$, will depend on the dynamical model of the ship and is defined as follows:

$$x_d(\alpha) = x_{p-1} + \alpha\cos(\delta_p(\alpha)), \tag{4.1.1a}$$

$$y_d(\alpha) = y_{p-1} + \alpha\sin(\delta_p(\alpha)), \tag{4.1.1b}$$

where $\delta_p(\alpha)$ presents the orientation of the path and can be expressed as

$$\delta_p(\alpha) = atan2(\dot{y}_p(\alpha), \dot{x}_p(\alpha)). \tag{4.1.2}$$

The LOS position p_{LOS} is a point located along the straight line segment with a lookahead distance, $\Lambda > 0$ from $p_d(\alpha)$, connecting the previous way-point and current way-point. The position p_{LOS} moves along the line as the ship is approaching the path reference.

Therefore, a dynamic projection algorithm is used to calculate the value of α so that the discrepancies between the path reference and the current position of the ship are minimized in real-time. Usually, this distance is so-called along-track (x_e) and cross-track error (y_e) and is described as:

$$\begin{bmatrix} x_e \\ y_e \end{bmatrix} = \begin{bmatrix} \cos(\delta_p(\alpha)) & -\sin(\delta_p(\alpha)) \\ \sin(\delta_p(\alpha)) & \cos(\delta_p(\alpha)) \end{bmatrix}^\top \begin{bmatrix} x_i - x_d(\alpha) \\ y_i - y_d(\alpha) \end{bmatrix}, \tag{4.1.3}$$

where $(x_d(\alpha), y_d(\alpha))$ as presented in (4.1.1), $\delta_p(\alpha)$ as in (4.1.2), (x_i, y_i) is current position of the i^{th} ship.

It is worth mentioning that in (4.1.3), to simplify, along-track error (x_e) is considered zero.

[6] Corresponding to each feasible path found, there is a desired course angle as can be seen in Fig. 4.1.2.

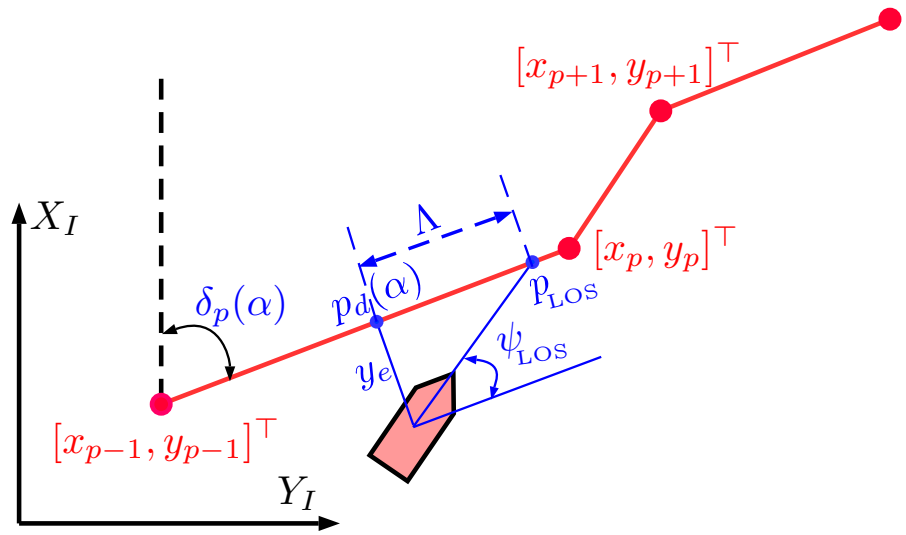


Figure 4.1.2: LOS guidance system for a feasible path.

The LOS angle, ψ_{LOS} (as can be seen in Fig. 4.1.2), is calculated based on the cross-track error y_e and the look-ahead distance Λ, expressed as:

$$\psi_{LOS} = -\frac{y_e}{\sqrt{y_e^2 + \Lambda^2}} \tag{4.1.4}$$

It is worth noting that in (4.1.4), arctan-function can also be used as described in [Qu et al., 2017], however, this approach will result in a very sophisticated controller when ψ_{LOS} is differentiated in the subsequent design steps[7].

The angle, ψ_{LOS}[8] is considered as a reference state of the heading angle ψ of surface vehicles to ensure convergence of the ships position to a feasible path generated by RRT*.

Fig. 4.1.2 depicts all the relevant parameters in (4.1.1), (4.1.2) and (4.1.3) in order to define the LOS guidance system, i.e., LOS angle as in (4.1.4).

4.2 Path planning for connectivity maintenance with COLREGS compliance with external disturbances

This section presents the motion planning algorithm for multi-surface vehicles connectivity maintenance. The motion planning strategy is illustrated via the block diagram in Fig. 4.2.1. From a map of the Trondheim fjord, the ground represented by static obstacles and the coordinates of the harbors can be acquired as seen in Fig. 3.1.3b. A collision-free path, and LOS guidance system can then be generated off-line from the starting harbor to the desired destination based on a set of waypoints yielded by the RRT* algorithm. NDO-based NMPC, a feedback control scheme in which an optimization problem is solved on-line at each time step, will ensure collision avoidance for dynamic obstacles and robustness in the presence of the external disturbances. If

[7] In case of $\psi_{LOS} = \frac{\pi}{2}$, the arctan-function goes to infinity.

[8] Note that, since locally Λ corresponds to the inverse proportional gain [Breivik and Fossen, 2008], the convergence to the path depends on the value of Λ. A low value means faster convergence than a larger one, but with a large overshoot.

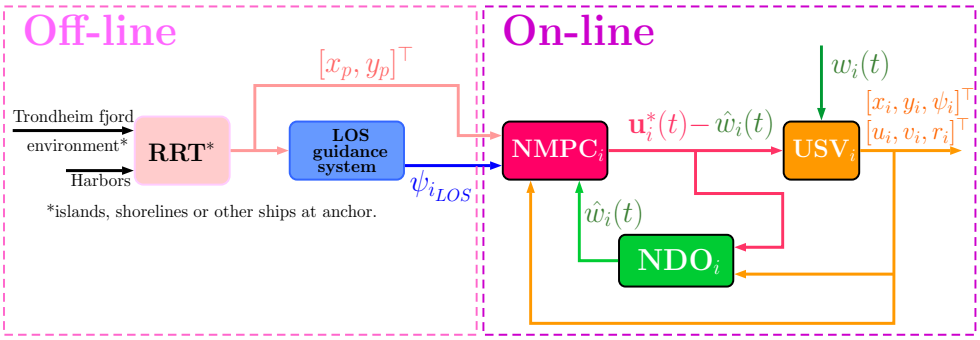


Figure 4.2.1: Path planning strategy for MAS.

there are no static or dynamic obstacles, the NDO-based NMPC will try to re-establish path following of the path obtained from RRT*.

4.2.1 Connectivity maintenance condition

We consider here that the information of the agents (trajectories, velocities, and the like) can only be exchanged with their neighbors who are inside the communication range.

Let us consider the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes representing all agents and $\mathcal{E} = \{(i,j) \in \mathcal{V} \times \mathcal{V} : i \leftrightarrow j \}$ is the set of edges of the graph. The adjacency matrix $\mathcal{A} = [a_{ij}]$ indicates if a pair of nodes are adjacent or not in the graph hence satisfying the property:

$$a_{ij} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.1}$$

Definition 4.2.1. *The neighboring set of agent $i \in \mathcal{V}$ is defined by $\mathcal{N}_i = \{j \in \mathcal{V} \setminus \{i\} : \Gamma_i^c\}$, where $\Gamma_i^c(p_i, r_i^c)$ is described as the communication range of agent i, with $r_i^c \in \mathbb{R}_+$, is the radius of the ball centered in $p_i \in \mathbb{R}^n$, the current position of agent i.*

Consequently, the condition of connectivity maintenance for information exchange among agent i and its neighbors ($\mathcal{N}_i$) is defined as follows:

$$\mathcal{N}_i = \{j \in \mathcal{V} \setminus \{i\} : \|\Delta p_{i,j}^d\| < \Gamma_i^c)\}. \tag{4.2.2}$$

Next, for any edge between two nodes of set $\mathcal{N}_i$, we define an on-off barrier function as in (3.2.2) whose value is near zero within the interval $[0, \min(\Gamma_i^c, \Gamma_j^c))$:

$$M_{i,j}(p_i, p_j) = \frac{L_{ij}}{1 + e^{-\beta_{ij}\left(\|\Delta p_{i,j}^d\| - \min(\Gamma_i^c, \Gamma_j^c)\right)}}. \tag{4.2.3}$$

Note that in this case, $\beta_{ij} > 0$ and L_{ij} has a high value such that the distance among the agents of $\mathcal{N}_i$ does not exceed the Γ_i because of penalizing.

Using the above construction we define a so called "*connectivity maintenance function*" of agent i with its neighbors $\mathcal{N}_i$:

$$\mathcal{M}_i(p_i, p_j) = \sum_{j \in \mathcal{N}_i} a_{ij} M_{i,j}(p_i, p_j). \tag{4.2.4}$$

4.2.2 NDO design for external time-variant disturbance

We first reconsider the ship model (3.1.1) as presented in Chapter 3 in the presence of the external time-variant disturbance:

$$\dot{\mathbf{x}}_i = f_i(\mathbf{x}_i(t), \mathbf{u}_i(t), w_i(t)) = \begin{cases} \dot{\eta}_i &= R_i(\psi_i)\nu_i, \\ M_i\dot{\nu}_i &= -C_i(\nu_i)\nu_i - D_i\nu_i + \mathbf{u}_i + w_i, \end{cases} \tag{4.2.5}$$

where the state vectors such as η_i, ν_i and the input vector, $\mathbf{u}_i$ are presented in detail in Chapter 3. The additive disturbance $w_i = \begin{bmatrix} w_{u_i} & w_{v_i} & w_{r_i} \end{bmatrix}^\top \in \mathbb{R}^3$ is not a pure Gaussian noise but rather the output of such a noise (after it passed through a nonlinear filter), is bounded by $\|w_i\| \leq \varrho_i$ and accounts for the ocean environment[9].

The unknown varying external disturbances (e.g., ocean currents) impact directly the control inputs of vessels leading to bad performance. Consequently, the nonlinear disturbance observer will provide an estimate of the disturbance, and its estimation is fed back to the NMPC solver. Since the control inputs of the ship model (4.2.5) are affected by the ocean currents accounting for vector w_i, a disturbance observer is presented for a general nonlinear dynamic. This disturbance observer will be used to estimate the uncertainties vector w_i with regard to the i-th agent.

A fundamental idea of disturbance observer in NMPC scheme is to estimate the unknown disturbance w_i by an estimate $\hat{w}_i$ such that $|\hat{w}_i - w_i| \to 0$. In fact, the acceleration, $\dot{\nu}_i$, is not available in many robotic manipulators, and it is also difficult to construct the acceleration signal from the velocity signal by differentiation due to measurement noise. Therefore, we first need to define the $q_i = M_i\nu_i$ as the auxiliary variable and its estimation $\hat{q}_i$. Consequently, the estimated error, q_{ie} is described as below:

$$q_{ie} = \hat{q}_i - q_i. \tag{4.2.6}$$

From the eq. (4.2.5), the time derivative of q_i can be rewritten as:

$$\dot{q}_i = -C_i(\nu_i)\nu_i - D_i(\nu_i)\nu_i + \mathbf{u}_i. \tag{4.2.7}$$

As the results of [Do, 2010, Chen et al., 2000], the time derivative of auxiliary variable's estimation, $\hat{q}_i$ is defined as follows:

$$\dot{\hat{q}}_i = L(-C_i(\nu_i)\nu_i - D_i(\nu_i)\nu_i + \mathbf{u}_i - h(q_{ie})), \tag{4.2.8}$$

where $L = diag\{c_1, c_2, c_3\} > 0$, are the gains of the NDO, and $h(q_{ie}) \in \mathbb{R}^3$ is a design vector to be determined.

Consequently, the time derivative of estimated error between auxiliary variable, q_i and its estimation, $\hat{q}_i$ described as:

$$\begin{aligned} \dot{q}_{ie} &= \dot{\hat{q}}_i - \dot{q}_i \\ &= (L - \mathbf{I_3})[-C_i(\nu_i)\nu_i - D_i(\nu_i)\nu_i + \mathbf{u}_i] - Lh(q_{ie}), \end{aligned} \tag{4.2.9}$$

where $\mathbf{I}_3$ is the identity matrix with dimension related to the state space vector of ν_i.

To ensure that (4.2.9) converges to zero, (4.2.9) has to satisfy two conditions:

1. $L = \mathbf{I}_3$,

[9] I.e., wind, wave, ocean current affect control input of the ship.

2. The designed function vector converges towards zero, i.e., $h(q_{ie}) \to 0$.

We propose the use of on-off barrier function with continuous nature in order to design $h(q_{ie})$ based on the estimated error, q_{ie} in (4.2.6). Hence, the design function, $h(q_{ie})$, is described as follows:

$$h(q_{ie}) = K_{1i}(q_{ie}) + K_{2i}|q_{ie}|\mathcal{J}(q_{ie}) + d_i\mathcal{J}_i(q_{ie}), \tag{4.2.10}$$

where $\mathcal{J}_i(q_{ie}) = \frac{e^{\beta_i^{NDO} M_i \nu_{ie}}}{\mathbf{1}+e^{\beta_i^{NDO} q_{ie}}} - \frac{e^{-\beta_i^{NDO} q_{ie}}}{\mathbf{1}+e^{-\beta_i^{NDO} q_{ie}}} \in \mathbb{R}^3$ is a vector whose dimension depends on vector $q_{ie} \in \mathbb{R}^3$; $K_{1i} = diag\{K_{1i}^1, K_{1i}^2, K_{1i}^3\}$, $K_{2i} = diag\{K_{2i}^1, K_{2i}^2, K_{2i}^3\}$ and $d_i = diag\{d_i^1, d_i^2, d_i^3\}$ positive gain matrices.

Remark 4.2.2. In [Zhu et al., 2017], the authors employed signum function $\left(\mathcal{J}_i(q_{ie}) = \frac{|q_{ie}|}{q_{ie}}\right)$ to design function $h(q_{ie})$. However, a discontinuity at zero is the main drawback which comes from the property of the signum function. Since design function $h(q_{ie})$ considers the estimation error of the real disturbance, a discontinuity at zero means that the convergence of this error toward zero will be indefinite.

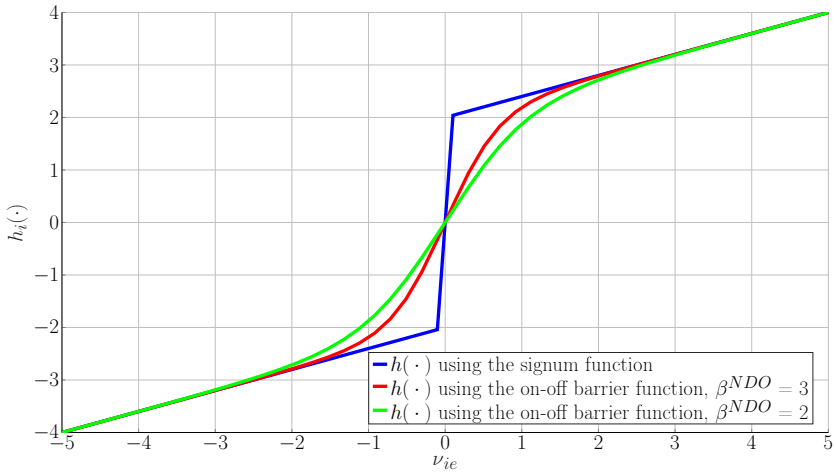


Figure 4.2.2: The comparison of designed function $h_i(\cdot)$ (4.2.10) between on-off barrier function (with the various value of $\beta^{NDO} > 0$) and signum function.

Fig. 4.2.2 compares the proposed function $h_i(\cdot)$ with two different values for β^{NDO} with the original signum function as in (4.2.10).

With $h_i(\cdot)$ as in (4.2.10), eq. (4.2.8) can be written as:

$$\begin{aligned}\dot{\hat{q}}_i = &- C_i(\nu_i)\nu_i - D_i(\nu_i)\nu_i + \mathbf{u}_i \\ &- K_{1i}(q_{ie}) - K_{2i}|q_{ie}|\mathcal{J}_i(q_{ie}) - d_i\mathcal{J}_i(q_{ie}).\end{aligned} \tag{4.2.11}$$

The estimated disturbance, $\hat{w}_i$ will be given by:

$$\begin{aligned}\hat{w}_i &= \dot{\hat{q}}_i + C_i(\nu_i)\nu_i + D_i(\nu_i)\nu_i - \mathbf{u}_i \\ &= -K_{1i}(q_{ie}) - K_{2i}|q_{ie}|\mathcal{J}(q_{ie}) - d_i\mathcal{J}(q_{ie}) = h(q_{ie}).\end{aligned} \tag{4.2.12}$$

The following proposition is introduced to highlight the convergence of estimated disturbances toward the real disturbances.

Proposition 4.2.3: Consider the additive disturbances in (4.2.5). Given the NDO as in Eq. (4.2.11)-(4.2.12), its estimate $\hat{w}$ will asymptotically track the input disturbances w. In other words, $\|\hat{w} - w\|_\infty \to 0$.

Proof. Consider the following Lyapunov function

$$V_i = \frac{1}{2}(q_{ie})^\top (q_{ie}). \tag{4.2.13}$$

The time derivative of the Lyapunov function is

$$\begin{aligned}
\dot{V}_i &= (q_{ie})^\top (\dot{q}_{ie}) = (q_{ie})^\top (\dot{\hat{q}}_i - M_i \dot{\nu}_i) \\
&= (q_{ie})^\top \left[-K_{1i}(q_{ie}) - K_{2i}|q_{ie}|\mathcal{J}_i(q_{ie}) - d_i \mathcal{J}_i(q_{ie}) \right] \\
&\leq -(q_{ie})^\top K_{1i}(q_{ie}) - (q_{ie})^\top K_{2i}|q_{ie}|\mathcal{J}_i(q_{ie}) \\
&\leq -2K_{1i_{min}} V_i - 2K_{2i_{min}} V_i \\
&\leq 0,
\end{aligned} \tag{4.2.14}$$

where $K_{1i_{min}} = min\{K_{1i}^1, K_{1i}^2, K_{1i}^3\}$ and $K_{2i_{min}} = min\{K_{2i}^1, K_{2i}^2, K_{2i}^3\}$.

The estimated error q_{ie} will converge to zero (i.e., $q_{ie} \to 0$) due to (4.2.14).

Combining (4.2.5) and (4.2.12), the disturbance error w_{ie} is rewritten as follows:

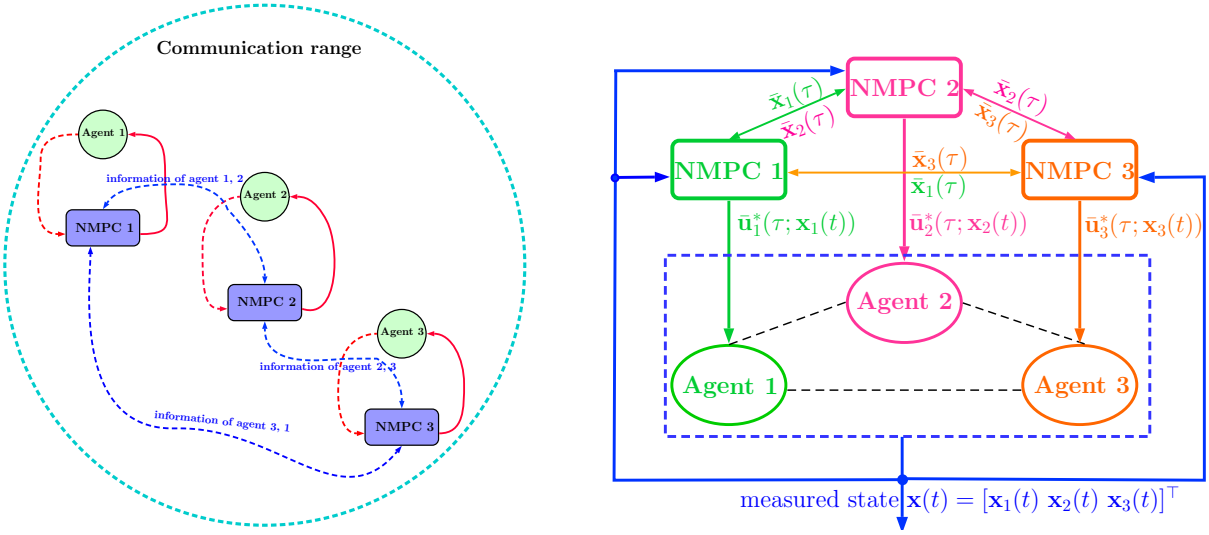
$$\begin{aligned}
w_{ie} &= \hat{w}_i - w_i \\
&= -K_{1i}(q_{ie}) - K_{2i}|q_{ie}|h_i(q_{ie}) - d_i h_i(q_{ie}) - M_i \nu_i - C_i \nu_i - D_i \nu_i + \mathbf{u}_i \\
&= \dot{\hat{q}}_i - \dot{q}_i = \dot{q}_{ie},
\end{aligned} \tag{4.2.15}$$

where $h_i(\nu_i)$ was defined in (4.2.10).

Since ν_{ie} is the velocity error (surge, sway and yaw rate), if ν_{ie} converges to zero, $\dot{\nu}_{ie}$ (acceleration's error) will converge to zero also. Hence, using the NDO as in (4.2.11)-(4.2.12), the observer $\hat{w}_i$ will converge towards the external disturbance w_i. □

4.2.3 NMPC - based distributed motion planning with disturbance compensation

The NMPC - based distributed approach of multi-agent systems has an operational mechanism based on information exchange between an agent and its neighbors to converge towards a standard solution. Each agent also has its controller, and their information will be transmitted among the local controllers within the communication range if the connectivity maintenance condition which is defined by (4.2.3) is guaranteed. Therefore, each one of them has some knowledge of the behavior of its neighbors. In other words, each controller receives the decision variables (trajectories, positions, velocities and the like) from the nearby neighbors through a specified communication topology to be aware of their behaviors and to make the right control decisions for the agent being controlled. The external disturbances (i.e., wind, waves and currents) acting on the agents (i.e., surface vehicles) are compensated by using the estimates obtained with the nonlinear observer.



(a) Communication topology of the distributed motion planning architecture.

(b) The communication between the distributed NMPC controllers.

Figure 4.2.3: Parallel distributed NMPC architecture.

Figure. 4.2.3a illustrates a communnication topology of distributed motion planning in the context of a group of the three agents if the connectivity maintenance among agents is guaranteed. The information is broadcasted and received by agents through the undirected graph in *parallel*.

Let us summarize in the following the necessary ingredients for formulating the motion planning with disturbance compensation algorithm:

1. to obtain a more flexible behavior, the look-ahead distance, Λ from (4.1.4) is time-varying and optimized to obtain accelerated convergence and negligible overshoot.
2. to satisfy collision avoidance with the ground and comply to rules 8 and 13 of the COLREGS with moving obstacles we add the repulsive potential constructions from (3.3.7), (3.3.10) in an NMPC optimization problem which will be activated only in the view range of the agents;
3. to comply with rules 14 and 15 of the COLREGS we add a slack variable in the cost to give priority to a negative rate of change of yaw moment, i.e., turning to starboard side as in [Mohamed et al., 2018].
4. to ensure the agents' connectivity we add in the optimization problem the connectivity function given in (4.2.4);
5. to reject the unknown disturbance from the current affecting the control system of the agents, the output of the NDO as in (4.2.11)-(4.2.12) is introduced as a compensation signal;
6. the implementation is done in a distributed fashion, i.e., the agents exchange information only with those inside their communication range.

The nominal system of i^{th} agent is defined from (4.2.5) by neglecting the disturbances, i.e., $w_i(t) \equiv 0$, and given by

$$\dot{\bar{x}}_i = f_i(\bar{x}_i(t), \bar{\mathbf{u}}_i(t), 0). \tag{4.2.11}$$

Problem 4.2.3 (Collison avoidance and connectivity maintenace for tracking DNMPC)**.** *The potential field approach, at this point, is the combination of the on-off repulsive potential fields representing static and dynamic obstacles (or the other agents)* (3.3.7) *and* (3.3.10) *and attractive potential function describing the goal configuration at a global energy minimum presented by quadratic function* (2.2.11)*. This is refomulated by finite prediction horizon strategies, as in Problem 2.2.8 in order to consider only the repulsive potential fields in the view ranges of MAS, thus leading to a better-behaved solution in terms of feasibility and collision avoidance.*

For each nominal system (4.2.11) we solve a finite horizon open-loop OCP (optimal control problem) at time t, using the measured state $x_i(t)$ over the prediction horizon T_p:

$$\min_{\bar{\mathbf{u}}_i(\cdot)} \quad E_i(\bar{\mathbf{x}}_i(t_k+T_p)) + \int_{t_k}^{t_k+T_p} \Big[\mathcal{L}_i^{\text{D-OORP}}(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{x}}_j(\tau), \bar{\mathbf{u}}_i(\tau))\Big] d\tau, \tag{4.2.12a}$$

subject to

$$\dot{\bar{\mathbf{x}}}_i(\tau) = f(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\tau), 0),\ \bar{\mathbf{x}}_i(t_k) = \mathbf{x}_i(t_k), \tag{4.2.12b}$$

$$\bar{\mathbf{x}}_i(\tau) \in \mathcal{X}_i; \bar{\mathbf{x}}_j(\tau) \in \mathcal{X}_j;\ \bar{\mathbf{u}}_i(\tau)) \in \mathcal{U}_i,\ \forall \tau \in [t_k, t_k+T_p], \tag{4.2.12c}$$

$$\Lambda_{\min} \leq \Lambda(\tau) \leq \Lambda_{\max}, \tag{4.2.12d}$$

$$\epsilon_i(\tau) > 0, \tag{4.2.12e}$$

$$\dot{\mathcal{T}}_{r_i}(\tau) - \epsilon_i(\tau) < 0, \tag{4.2.12f}$$

where $\mathcal{L}_i^{\text{D-OORP}}(\cdot)$ is the cost per stage of distributed motion planning using the on-off repulsive potential for collision avoidance and connectivity-preserving approach.

$$\mathcal{L}_i^{\text{D-OORP}}(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\cdot)) = \mathcal{L}_i^{\text{OORP}}(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\cdot)) + \|\dot{\bar{\mathbf{u}}}_i(\tau)\|^2_{\Delta\mathbf{R}_i} + \mathcal{M}_i(\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{x}}_j(\tau)) + \|\epsilon_i(t)\|^2_{\mathbf{S}_i}, \tag{4.2.13}$$

with $\mathcal{L}_i^{\text{OORP}}(\cdot)$ defined as in (3.4.3).

The terminal cost is defined as:

$$E_i(\bar{\mathbf{x}}_i(t+T_p)) = \|\bar{\mathbf{x}}_i(t+T_p) - \mathbf{x}_{i,ref}\|^2_{\mathbf{P}_i}. \tag{4.2.14}$$

In (4.2.12b), $f_i(\cdot,\cdot)$ is presented in (4.2.5), $\bar{\mathbf{x}}_i(\tau), \bar{\mathbf{u}}_i(\tau)$ are the predicted states and inputs while $\bar{\mathbf{u}}_i(\cdot)$ represents the predicted input trajectory along the prediction horizon T_p. In the cost per stage (4.2.13), $\dot{\bar{\mathbf{u}}}_i(\tau)$ is the predicted input variations, $\mathcal{M}_i$ penalises the connectivity maintenance between agent i and its neighbors as defined in (4.2.4) and the last term ϵ_i is the slack variable. $\mathbf{Q}_i, \mathbf{R}_i, \mathbf{P}_i$ and $\mathbf{S}_i$ are (semi)-positive definite weighting matrices of appropriate dimensions. Look-ahead constraint, Λ is presented in (4.2.12d). Constraint (4.2.12f) is tightened by the positive slack variable, ϵ_i (4.2.12f) in order to prioritize the negative rate of change of the yaw moment. This forces the vessel to turn to starboard in any situation, hence complying with the rules 13, 14 and 15 of COLREGS (see Appendix A). Rule 8 is also obeyed through the use of the repulsive potential in (3.3.10).

At each sampling instance, each agent solves its OCP (4.2.12a) and yields the minimizing control sequence for the nominal system (4.2.11) over the interval $[t, t+T_p]$. Then, only the first sample

of the control input will be combined with the estimated disturbance in a closed-loop to yield a compound control as follows:

$$\mathbf{u}_i(t) = \mathbf{u}_i^*(t) - \hat{w}_i(t). \tag{4.2.15}$$

Thus, the agents compensate external disturbances in real-time and obtain a predicted collision-free trajectory $\bar{\mathbf{x}}_i(\tau)$ through the use of repulsive potential fields constructed for fixed and moving obstacles. The information (predicted trajectory) of agent i will be transmitted to its neighbor $j \in \mathcal{N}_i$ whenever the connectivity between them holds. Then the agent j, with the information updated from agent i, will solve its own OCP (4.2.12a). This procedure will continue until agent $i \in \mathcal{V}$ solves its own problem (4.2.12a) and is repeated at next sampling instance.

Note that the repulsive potentials of static obstacles (3.1.9) and the moving obstacle/other ships (3.1.6) are taken into account over the prediction horizon for collision avoidance, and is called repeatedly, for each time instant τ. However, they are disabled due to combining with on-off barrier function as in (3.3.6) and (3.3.9) (it has a "0" value if nothing is in the ship's view range). The repulsive potential will be activated via on-off barrier function (it has a "1" value if the moving obstacle/other ships or the fixed obstacles are in the ship's view range). This approach has shown effectiveness when the potential field changes constantly via on-off barrier function thus allowing the vessel to avoid getting trapped inside a local minima as presented in Chapter 3.

Note that the distributed NMPC scheme employed is a parallel architecture with the aim of improving the closed-loop performance for USVs (compared with the sequential architecture) [Liu et al., 2010]. More specifically, the separate NMPC controllers are evaluated based on a new measured state (as can be seen $x(t)$ in Fig. 4.2.3b) and broadcast their information (i.e., predicted states, for example, $\bar{x}_1(\tau)$, $\bar{x}_2(\tau)$ and $\bar{x}_3(\tau)$ in Fig. 4.2.3b) to their neigbors if they stay interior of the communication range in *parallel* at the same sampling time. Fig. 4.2.3b illustrates the communication topology of the distributed control framework for three agents.

Several remarks are as follows:

Remark 4.2.4. The nominal system (4.2.11) admits only one solution with one input.

Remark 4.2.5. Only the nominal system (4.2.11) is utilized in OCP.

Remark 4.2.6. The initial state of NDO, $\hat{q}_i(\tau)$ (4.2.11), has to be provided.

Remark 4.2.7. Regarding the rate of change of yaw moment, $\dot{\mathcal{T}}_{r_i}$, there are two possible cases: i) $\dot{\mathcal{T}}_{r_i} > 0$ (i.e., ship will turn to port side), ii) $\dot{\mathcal{T}}_{r_i} < 0$, (i.e., ship will turn to starboard).
The slack variable weight should be carefully chosen so that $\dot{\mathcal{T}}_{r_i}$ can only be a very small positive number but still satisfy constraint (4.2.12f) while the slack variable, ϵ_i, is always a small positive number as in (4.2.12e). These restrictions combine to make case ($\dot{\mathcal{T}}_{r_i} > 0$) improbable. In other words, the case ($\dot{\mathcal{T}}_{r_i} < 0$) is strongly encouraged.

Remark 4.2.8. The prediction horizon should be chosen large enough to ensure cover of the agent's view range or at least assure that the agent has time enough to avoid an imminent collision. Conversely, the prediction horizon cannot be chosen arbitrarily large due to numerical and computational issues.

The details are presented in Algorithm 2. The procedure of information exchange among the agents in Algorithm 2 is based on [Müller et al., 2012] and Algorithm 1 as presented in Chapter 3. Collision-free motion planning with the rejection of external disturbances through NDO is proposed by the authors and described in the following:

Algorithm 2 Information exchange procedure and collision avoidance of the NMPC-based distributed motion planning with the rejection of external disturbances through NDO.

Require: Consider the set of agents $\mathcal{V}$ with the nominal model (i.e., w_i is zero in (4.2.11)), the set of repulsive potentials of the forbidden cells (3.1.9) and of the agents (3.3.4), the on-off barrier function as in (3.3.6) and (3.3.9), the safe distance $D_s^{i,\ell}$, the vew and communication ranges of agents $i, j \in \mathcal{V}$, set of waypoint obtained through RRT*.

1: $\tau \in [t, t + T_p]$;
2: Initialize the NDO as in (4.2.11)-(4.2.12);
3: Each controller updates the measured state ($\mathbf{x}(t_k)$) and shared information;
4: **for** $i = 1 : N_a$ **do**
5: Agent i solves OCP (4.2.12a) and obtains the predicted trajectory $\bar{x}_i(\tau)$;
6: **if** the connectivity conditions between agent i and the others are satisfied **then**
7: NMPC i broadcasts its entire information (i.e., predicted state $\bar{x}_i(\tau)$) to its neighbors ($\mathcal{N}_i$). $\mathcal{N}_i$ receives the predicted state $\bar{x}_i(\tau)$ of NMPC i and evaluates their predicted input trajectories based on the measured state $\mathbf{x}(t_k)$ and received predicted state (i.e., $\bar{x}_i(\tau)$);
8: **if** $\|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| > \min(\Gamma_i, \Gamma_j)$ **then**
9: inactivate static and dynamic repulsive potential fields for the ℓ^{th} fixed and and the j^{th} moving obstacle (or other agent) in (4.2.13);
10: **end if**
11: **if** $\|\Delta p_{i,\ell}^s\| > D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| \leq \min(\Gamma_i, \Gamma_j)$ **then**
12: inactivate the ℓ^{th} static RP of the fixed obstacles and activate the dynamic RP of the j^{th} agent (or moving obstacle) in (4.2.13);
13: **end if**
14: **if** $\|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| > \min(\Gamma_i, \Gamma_j)$ **then**
15: activate the ℓ^{th} static RP of fixed obstacle whose value stays within the range $(0, 1]\Phi_\ell^f(\cdot)$[10]and inactivate the dynamic RP of the j^{th} agent (or moving obstacle) in (4.2.13);
16: **end if**
17: **if** $\|\Delta p_{i,\ell}^s\| \leq D_s^{i,\ell}$ and $\|\Delta p_{i,j}^d\| \leq \min(\Gamma_i, \Gamma_j)$ **then**
18: activate the ℓ^{th} static RP of fixed obstacle whose value stays within the range $(0, 1]\Phi_\ell^f(\cdot)$ and activate the dynamic RP of j^{th} agent (or moving obstacle) whose value stays within the range $(0, 1]\Upsilon_j(\cdot)$[11]in (4.2.13);
19: **end if**
20: **end if**
21: Apply only the first sample of the compound control sequence (i.e., $\bar{\mathbf{u}}_i^*(\tau; \mathbf{x}_i(t_k))$) as in (4.2.15) to both the nominal system and NDO over the interval $[t, t + T_p]$;
22: **end for**
23: Continue to the next sampling instance;
24: RETURN step 1;

4.3 Simulation results for collision avoidance and connectivity maintenance

Hereinafter we use the Cybership II model for ship dynamics. This characterizes a real ship at a scale of 1:70. Since we aim to test our algorithm on a realistic benchmark (using real movement

[10] $\Phi_\ell^f(\cdot)$ is repulsive potential of the ℓ^{th} static obstacle as defined in (2.2.12).
[11] $\Upsilon_j(\cdot)$ is repulsive potential of the j^{th} agent (or moving obstacle) as defined in (3.1.6).

data, as given by the AIS), we scale all the information (distances, velocities) proportionally to the scale mentioned earlier (e.g., the distance between harbors used in simulation is 70 times less than the real one).

Consider a set of $N_a = 3$ underactuated ships with the dynamical model (4.2.5) with the length overall (LOA) being 1.255 [m], and the matrices[12] R, M, D and B as presented in the simulation section in Chapter 3. These vessels are simulated to navigate in the Trondheim fjord (Norway). Fig. 3.1.3b illustrates the operating region in which the ships need to navigate among different harbors while simultaneously maintaining connectivity and avoiding the shore or small islands and other vessels. The Automatic Identification Systems (AIS) is used to provide real numerical data related to the position and velocity of ships, as well as time of navigation between harbors.

Furthermore, in order to ensure that the ships can broadcast and be aware of each other's complete information, we have three neighboring sets. The neighbors of ship 1 is defined $\mathcal{N}_1 = \{2, 3\}$ (i.e., neighbors of ship 1 are ship 2 and 3) and $\mathcal{N}_3 = \{1, 2\}$ (i.e., neighbor of ship 3 are ship 1 and 2).

The three agents in the group need to maintain the connectivity among them while traveling from Orkanger to Trondheim harbor (for situation 1) and Stjordal harbor (for situation 2) and avoid fixed obstacles and a mobile obstacle whose dynamical model is described in (4.2.5).

The number of cells considered from partitioning the map as in Fig. 3.1.3b is $N_{\text{cell}} = 22$.

The view-range of the ships used in (3.3.5), (3.3.8) is $\Gamma_i = 35$ [m]. The scaling coefficient is $\varepsilon_\ell \in [1.2, 1.8]$.

In order to observe easily, the communication range, Γ_i^c for information exchange among ships used in (4.2.3), is assumed 70 [m].

The steepness of the repulsive potential defined in (3.3.6) and (3.3.9) is given by $\beta_i = 1$. The steepness of connectivity maintenance function (4.2.4), β_{ij} and curve's maximum value, L_{ij} used as in (4.2.3) are $\beta_{12} = \beta_{13} = \beta_{23} = 0.002$ and $L_{12} = L_{13} = L_{23} = 1000$. Other parameters of the NMPC optimization problem in (4.2.13) are chosen as follows: the weighting matrices $\mathbf{Q}_i = 0.1\mathbf{I}_6$, $\mathbf{R}_i = 0.1\mathbf{I}_2$, $\mathbf{P}_i = [0.5\mathbf{I}_2 \quad \mathbf{0}_2 \quad \mathbf{0}_2; \mathbf{0}_2 \quad \mathbf{I}_2 \quad \mathbf{0}_2; \mathbf{0}_2 \quad \mathbf{0}_2 \quad \mathbf{I}_2]$, $\mathbf{S}_i = 10^{-6}$. We consider a conservatively chosen prediction horizon $T_p = 6s$ in order to asses less than ideal conditions (decisions have to be made when the other ships/obstacles are already very close) and the sampling time T_e is $2s$.

The design parameters of the NDO in (4.2.11) and (4.2.12) are chosen as follows: $\lambda_i = 1$ and $\beta_i^{NDO} = 1$, $K_{1i} = diag\{0.2, \quad 0, \quad 0.2\}$, $K_{2i} = diag\{0.025, \quad 0, \quad 0.025\} = d_i$.

We consider the surge velocity of agent 1 to be the highest ($u_1 \in [-0.2, 0.6]$ [m/s]); the surge velocities of agent 2 and 3 equal ($u_2 = u_3 \in [-0.2, 0.4]$ [m/s]) and less than the surge velocity of agent 1. The yaw angles' constraint is $[-\pi, \pi]$. We consider constraints on the actuation force $\mathcal{T}_{u_1} \in [-2, 2]$ [N], $\mathcal{T}_{u_2} = \mathcal{T}_{u_3} \in [-2, 2]$ [N] and on the yaw moment $\mathcal{T}_{r_1} = \mathcal{T}_{r_2} = \mathcal{T}_{r_3} \in [-0.2, 0.2]$ [Nm]. The look-ahead distance is taken in the interval $\Lambda \in [2 \cdot \text{LOA}, 10 \cdot \text{LOA}]$.

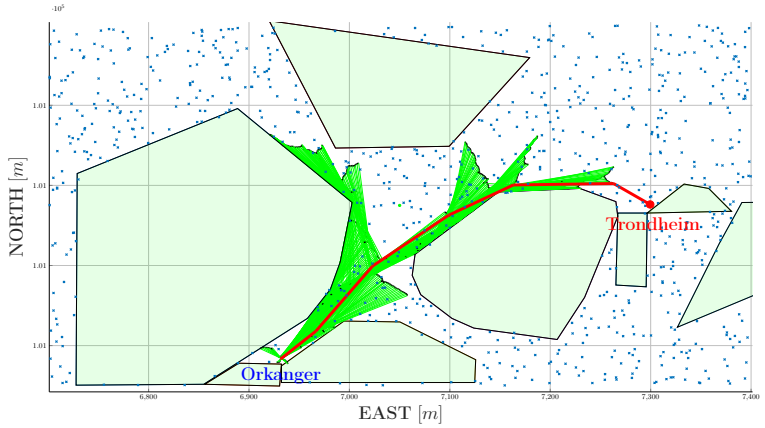
The simulations are done using IPOPT solver and CASADI [Andersson et al., 2018] toolkit in Matlab R2016a on a computer with the following configuration: Intel Core i7-4790CPU, 3.60GHz, 8GB RAM.

[12]For simplicity, the Coriolis matrix is neglected.

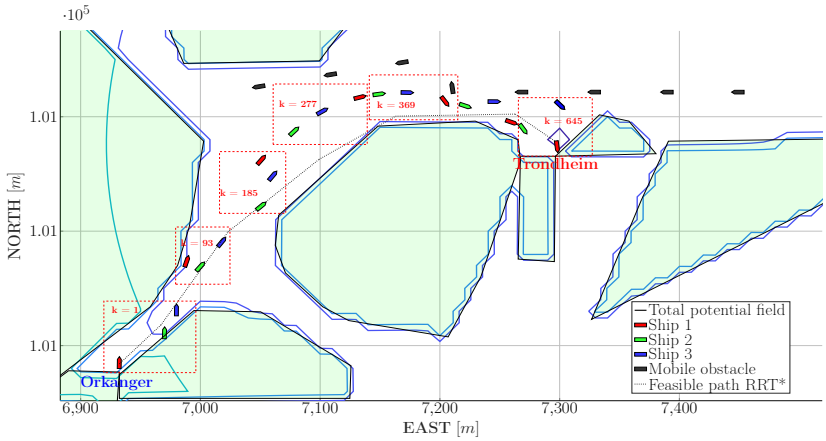
4.3.1 Scenario 1: Connectivity maintenance of MAS

Scenario 1 evaluates the validity of motion planning for connectivity maintenance in a group of three vessels in the course of going to the desired harbors, while maintaining collision avoidance, complying with the COLREG rules, avoiding dynamic obstacles under LOS guidance in the absence of disturbances.

Situation 1: Multi-USVs depart from Orkanger to Trondheim harbor



(a) Feasible collision-free path from Orkanger to Trondheim harbor.

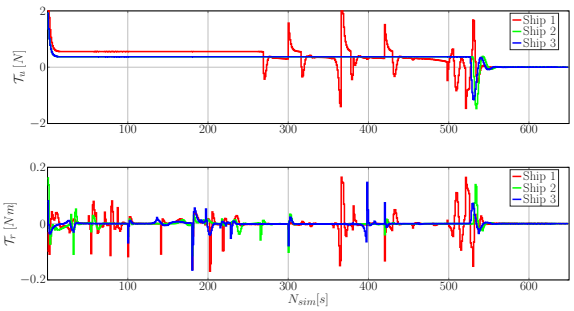


(b) Connectivity maintenance of the 3 USVs in situation 1 with COLREGS compliance - rules 13 and 14.

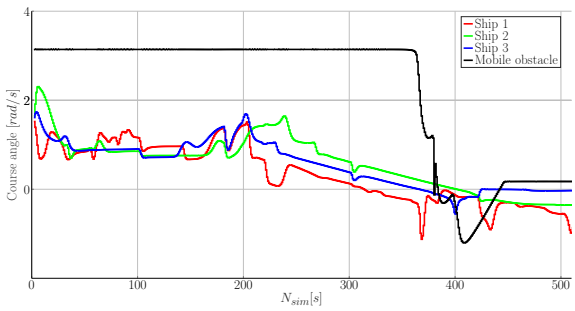
Figure 4.3.1: Motion planning of the three agents in situation 1.

Fig. 4.3.1a shows the off-line collision-free path generated using the RRT* algorithm after 1500 iterations.

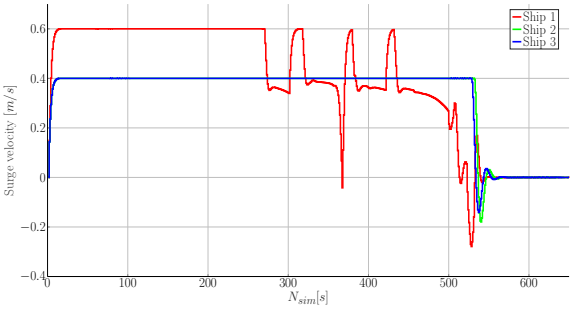
Fig. 4.3.1b depicts the actual motion of the three vessels traveling from Orkanger to Trondheim harbor. In the same figure, we show the group of agents at 7 different time instances as they are changing their formation configuration, for instance due to the moving obstacle motion illustrated in black. Fig. 4.3.2d shows that the connectivity condition is maintained.



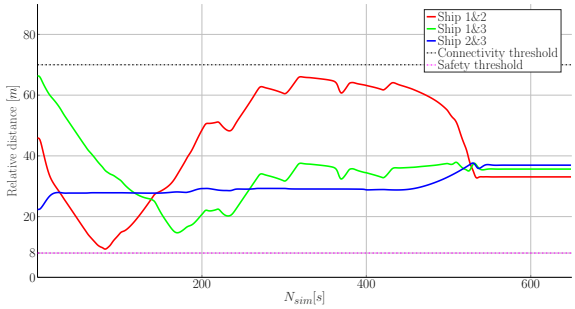
(a) Control inputs of the three vessels.



(b) Course angles of the three vessels and obstacle ship.



(c) Surge velocities of the three vessels.



(d) Relative distances of the three agents.

Figure 4.3.2: Control inputs, course angles, surge velocities and relative distances of the three agents in situation 1.

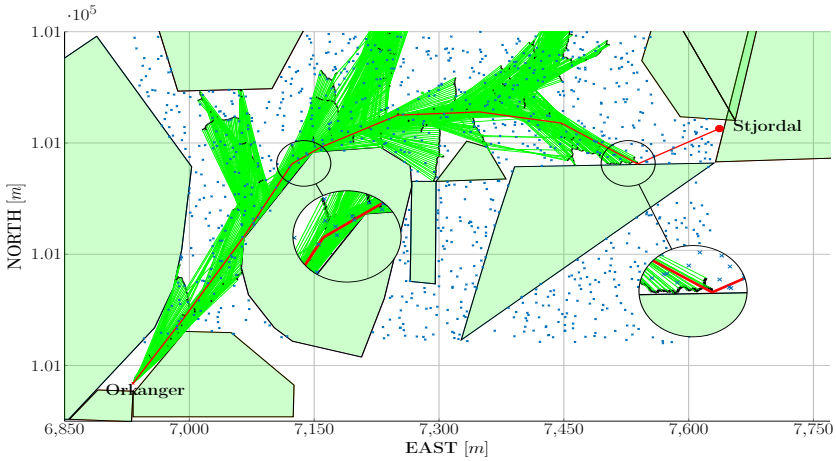
At time instance $k = 93$, the ship 1 (in red) is overtaking on the port side of the second and third ship, hence satisfying the COLREGS rule 13 - the overtaking situation. This can also be observed in Fig. 4.3.2b and 4.3.2c where ship 2 (in green) and 3 (in blue) are being overtaken, not changing their course and speed until step $k = 135$. In time interval $k \in [365, 400]$ the COLREGS rule 14 concerning head-on collisions between ships and/or mobile obstacles, becomes active. The first ship in the group (in red) and the mobile obstacle start changing their course angles to turn to their starboard (right side) at time step $k = 365$ (Fig. 4.3.2b), then the course angle of the third ship of the group (in blue) has a significant increase at time step $k = 400$ (see also Fig. 4.3.2b). Note that if the course angle of the ship is increasing, then the vessel is steered to its starboard and vice-versa.

Fig. 4.3.2a depicts the control inputs corresponding to surge thrust and rudder deflection of the three vessels.

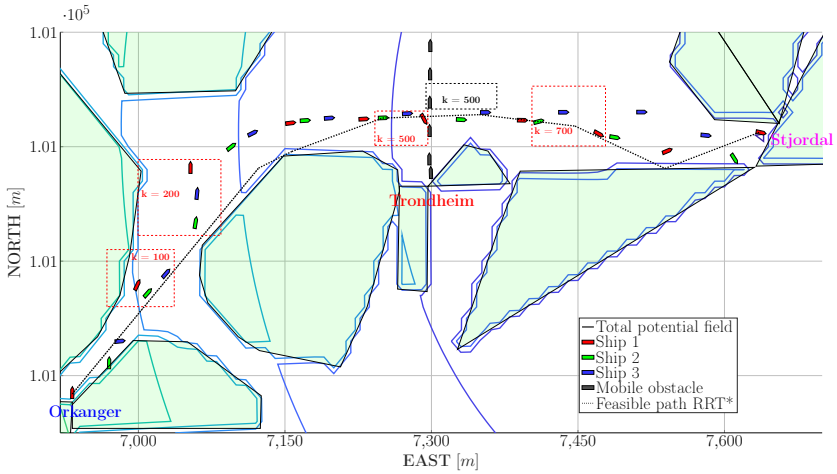
Situation 2: Multi-USVs depart from Orkanger to Stjordal harbor

Similarly with situation 1, a collision-free path is generated from Orkanger to Stjordal harbor via RRT* after 1500 iterations as in Fig. 4.3.3a.

Fig. 4.3.3b illustrates the actual motion of the three vessels traveling from Orkanger to Stjordal harbor. We show the group of agents at 10 different time instances changing their formation configuration due to the moving obstacle illustrated in black. Note that starting with time step $k = 200$, the three ships steer away from the given path due to the repulsive potential of the forbidden cell for ensuring a safe distance between agents and the shore-line. However, after some steps, in particular, at $k = 416$, the agents re-approach the given path and continue



(a) Feasible collision-free path Orkanger to Stjordal harbor.

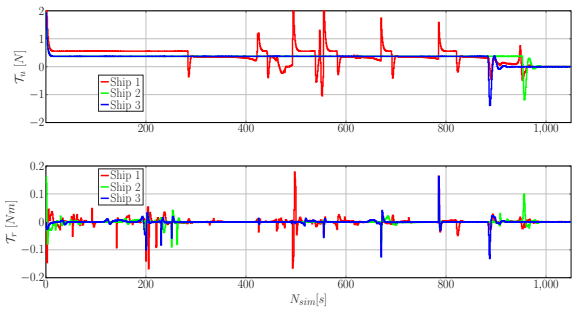


(b) Connectivity maintenance of the three ships group in situation 2 with COLREGS compliance - rules 13 and 15.

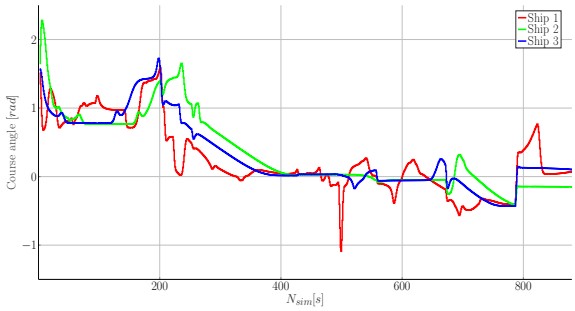
Figure 4.3.3: Motion planning of the three agents in situation 2.

tracking it. A crossing situation (Rule 15 of COLREGS) between the group of three ships (which start from Orkanger harbor) and the mobile obstacle (which starts from Trondheim harbor) is encountered. Since the mobile obstacle crosses from the starboard of the three ships group, the mobile obstacle can maintain its direction, while the three ships must turn starboard for collision avoidance at time step $k = 500$. The change of the ships' course angles can also be observed in Fig. 4.3.4b, the first (the red line) and third (the blue line) ship's course angles start increasing at time steps $k = 500$ and $k = 515$, respectively for collision avoidance on starboard with the mobile obstacle. The course angle of the second (the green line) ship is not changing since the mobile obstacle has passed by the time they enter this area. Fig. 4.3.4a and Fig. 4.3.4c present the control input and the surge velocities of the three ships respectively.

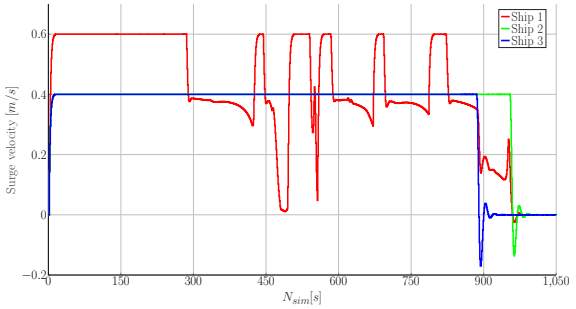
Collision avoidance and connectivity-preserving among agents are also validated since the relative distances are greater than the safety threshold and smaller than the connectivity threshold, as can be seen in Fig. 4.3.4d.



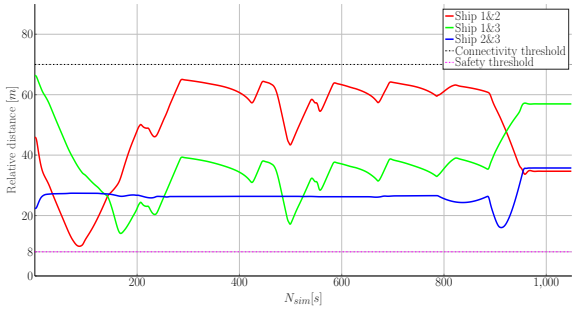
(a) Control inputs of the three vessels.



(b) Course angles of the three vessels and obstacle ship.



(c) Surge velocities of the three vessels.



(d) Relative distances of the three vessels.

Figure 4.3.4: Control inputs, course angles, surge velocities and relative distances of the three agents in situation 2.

4.3.2 Scenario 2: The environmental disturbances affect the performance of MAS

Assume that the disturbances originated by the ocean environment are time-varying and are described as follows:

$$\begin{cases} w_{u_i} = 0.96sin(0.02t) + 0.84sin(0.03t), \\ w_{v_i} = 0, \\ w_{r_i} = -0.16sin(0.09t + \frac{\pi}{3}) - 0.02sin(0.01t). \end{cases} \tag{4.3.1}$$

Then the external disturbances are bounded by $\begin{bmatrix} \|w_{u_i}\| & |w_{r_i}| \end{bmatrix}^\top \leq \begin{bmatrix} 1.72 & 0.18 \end{bmatrix}^\top$.

Situation 1: Multi-USVs depart from Orkanger to Trondheim harbor

In this scenario, the NMPC-based optimized control is combined with NDO (eq. (4.2.11) and (4.2.12)) in the presence of the time varying disturbance as in eq. (4.3.1). The effectiveness of the combination between NMPC and NDO is shown in Fig. 4.3.5 where the trajectories of three vessels are similar with those shown in situation 1 of scenario 1. The connectivity maintenance is also guaranteed as observed in Fig. 4.3.6d. Although the course angle of the three ships is oscillating more than in the case of no disturbance (in Fig. 4.3.2b), there is also an increased trend for the ships and the mobile obstacle to steer to their starboard side. For example, at the time instance $k = 368$, the course angle of ship 1 (red line) and mobile obstacle (black line) increase, which also can be observed in Fig. 4.3.6b. As a consequence, the COLREG rules 13 and 14 are maintained as in situation 1 of Scenario 1.

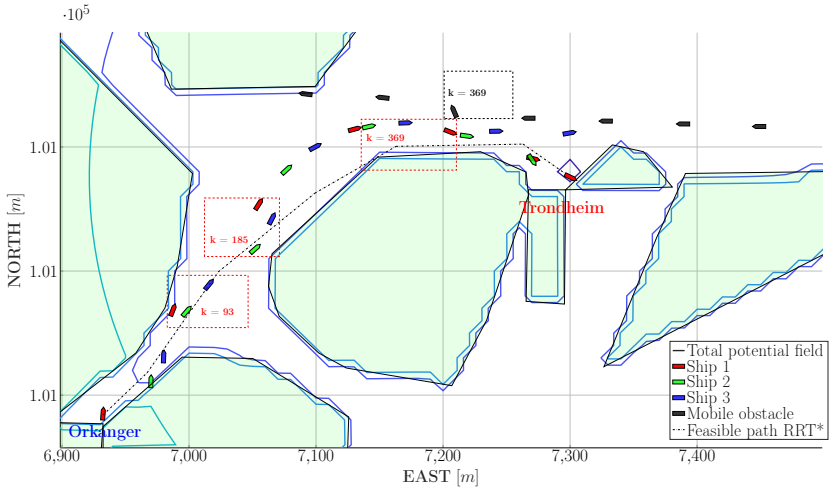
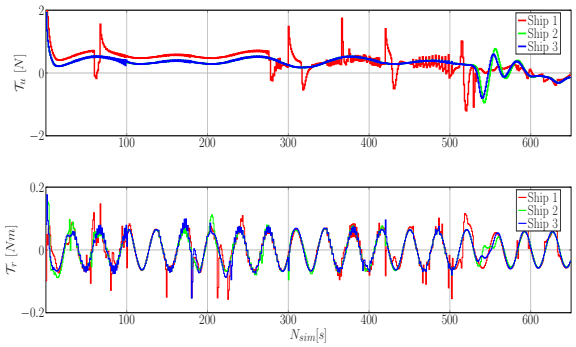
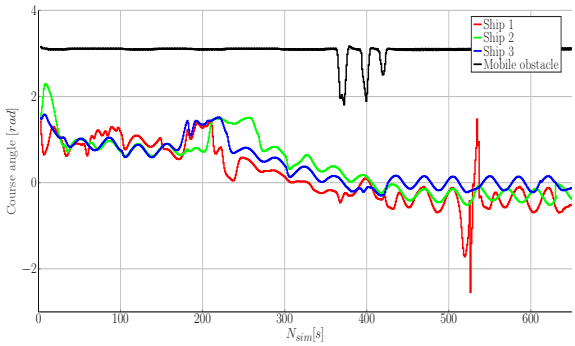


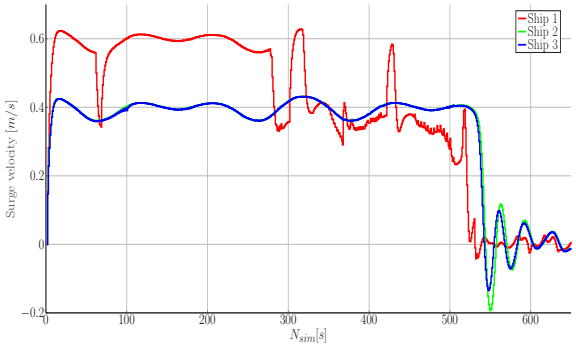
Figure 4.3.5: Connectivity maintenance of the three ships group for situation 1 in the presence of disturbances and NDO.



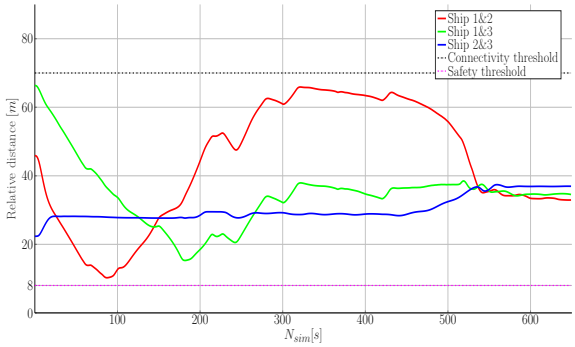
(a) Control inputs of the three vessels.



(b) Course angles of the three vessels and obstacle ship.



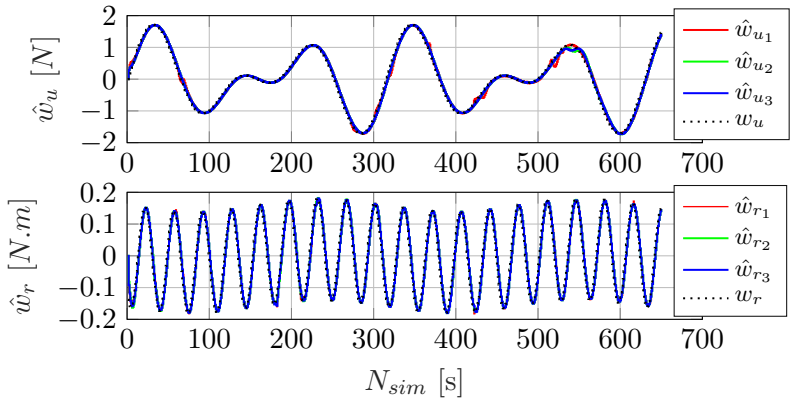
(c) Surge velocities of the three vessels.



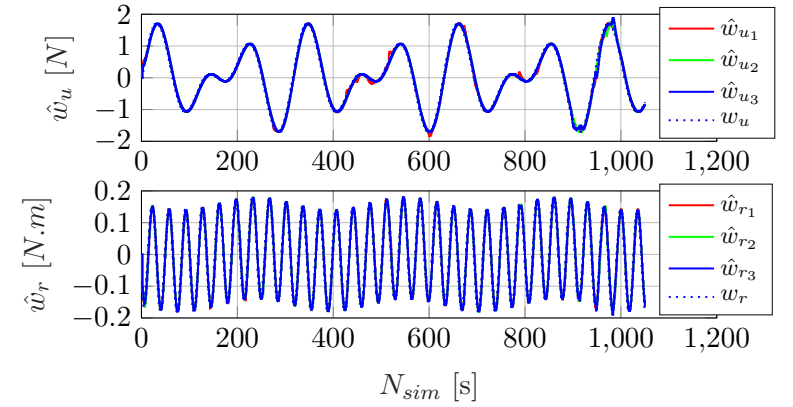
(d) Relative distances of the three vessels.

Figure 4.3.6: Control inputs, course angles, surge velocities and relative distances of the three agents in situation 1 in the presence of disturbances and NDO.

Fig. 4.3.6a and Fig. 4.3.6c illustrate the control inputs and surge velocites of the three vessels.



(a) Situation 1 of Scenario 2.



(b) Situation 2 of Scenario 2.

Figure 4.3.7: The estimated disturbances of the situation 1 and 2.

Since the nonlinear disturbances of each ship can be estimated and compensated for, the estimated perturbations ($\hat{w}$) of three ships converge towards the real disturbance (w), as depicted in Fig. 4.3.7a. The effectiveness of the use of on-off barrier function for the designed function (eq. (4.2.10)) (for example, the NDO of the 2^{nd} vessel) can be seen in Fig. 4.3.8. The errors between estimated and actual disturbances when applying the on-off barrier function (the red one) have had shown a slight improvement.

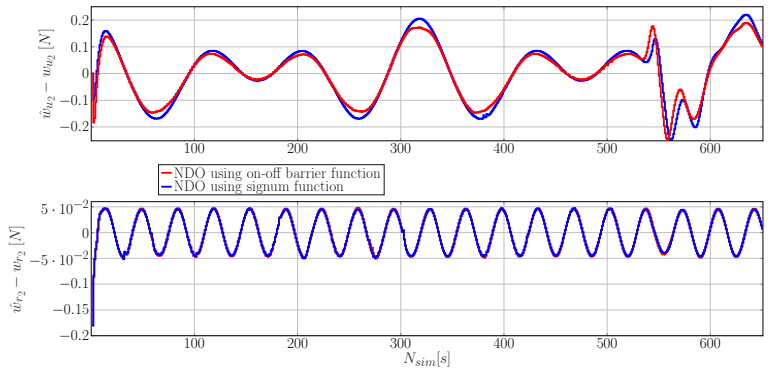
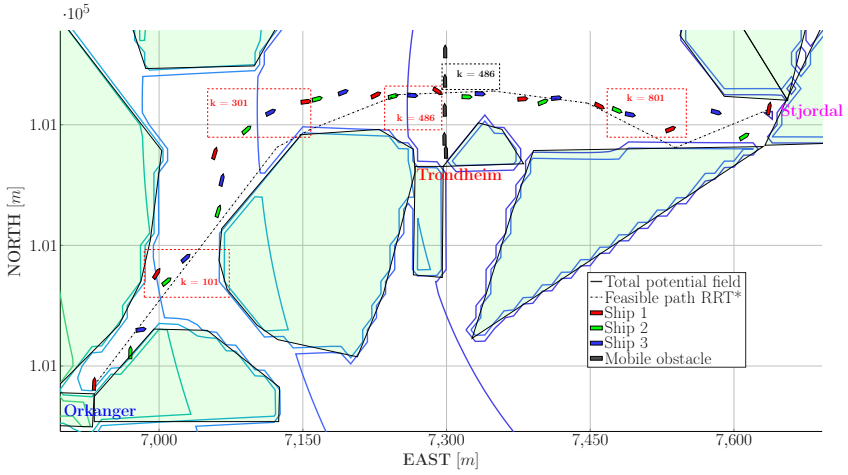
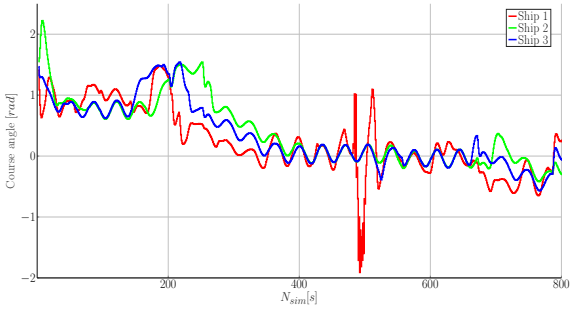


Figure 4.3.8: Comparison of NDO using signum function and on-off barrier function.

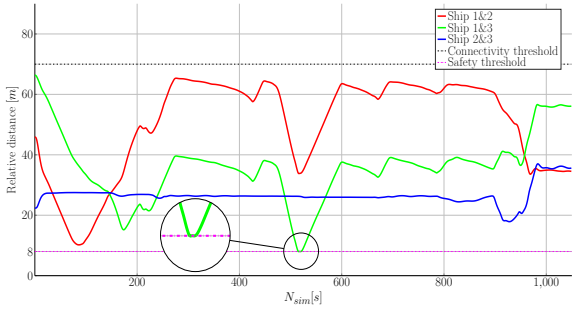
Situation 2: Multi-USVs depart from Orkanger to Stjordal harbor



(a) Connectivity maintenance of the three vessels group.



(b) Course angles of the three vessels.



(c) Relative distances of the three vessels.

Figure 4.3.9: Simulation results of the three vessels in situation 2 in the presence of disturbances and NDO.

In this situation, the connectivity maintenance and collision avoidance among agents in the group are maintained in simulation as illustrated in Fig. 4.3.9c.

Rule 13 of COLREG - overtaking is obeyed as illustrated in Fig. 4.3.9a (at time instance $k = 101$, ship 1 in red, overtakes on the left side of the two ships). Moreover, rule 15 - crossing, is also complied with when the obstacle ship keeps its direction and ship 1 turns starboard for collision avoidance as can be seen ($k = 486$) in Fig. 4.3.9a. This can also be seen in Fig. 4.3.9b, the course angle of ship 1 (red) which increases at $k = 486$ while course angles of ship 1 (green) and 3 (blue) are not changing since the mobile obstacle has passed by the time they enter this area.

Similar to situation 1 in this scenario, good results are obtained because NDO provides an estimate that rapidly converges towards the real disturbances depicted in Fig. 4.3.7b. This allows the ships' trajectories in the presence of external disturbances (Fig. 4.3.9a) to behave like the trajectories in the absence of disturbances (Fig. 4.3.3b).

Note that in Fig. 4.3.1b, 4.3.3b, 4.3.5 and 4.3.9a, we also illustrate the repulsive and attractive potentials projected in 2D (the repulsive potentials appear around the forbidden region preventing the ship's collison with the fixed and moving obstacles and the attractive potential is represented by circles around the destination harbor).

Computation time of the optimization problem, trajectory length and number of simulations to arrive at the destinations are delineated in Table 4.3.1.

Table 4.3.1: Performance criteria for the motion planning algorithm of situation 1 and 2 for Scenario 2 and 3.

Situation 1	Scenario 1	Scenario 2
Prediction horizon [s]	6	6
Number of simulations	650	650
CPU time [s]/step	0.0039	0.0039
Length of trajectory [m]	505.2	503.4
Situation 2	Scenario 1	Scenario 2
Prediction horizon [s]	6	6
Number of simulations	1050	1050
CPU time [s]/step	0.0103	0.0103
Length of trajectory [m]	847.6	858.5

4.4 Conclusions

In this chapter, we first presented the collision-free path generation, which is based on the randomized sampling algorithm, namely RRT*. This algorithm assures the necessary conditions for sampling-based motion planners towards probabilistic completeness, asymptotic optimality, and monotone convergence. This path is generated off-line and based on a set of the waypoints yielded by the RRT* algorithm. It is further parameterized by the associated heading angle which is to be used for the LOS guidance system employed by the USVs for path tracking.

An NDO-based distributed NMPC scheme has been developed for the motion planning problem which ensures connectivity maintenance and collision avoidance for multi-surface vehicles. The classical non-convex constraints from this problem were penalized in the cost function through appropriate potential field constructions. NDO can estimate the external disturbance and compensate for their impact via feedback control. Simulation results prove the excellent performances of the algorithm as well as giving insights for real implementation.

More details on LOS guidance system used for multi-vehicle surfaces can be found in [Breivik and Fossen, 2004, Børhaug et al., 2010, Hinostroza et al., 2018, Jiang et al., 2019, Wang et al., 2019c]. NDO formulations can be found in [Fu et al., 2018, Gao and Guo, 2019, He et al., 2019b]. Some applications of tracking MPC for autonomous surface vehicles are introduced in [Yan and Wang, 2012, Guerreiro et al., 2014, Sun et al., 2018a, Hagen et al., 2018, Persson and Wahlberg, 2019, Chen et al., 2019].

Future work will concentrate on the recursive feasibility of NMPC, the stability of the closed-loop system as well as experimental results.

Chapter 5

Conclusions and future developments

5.1 Conclusions

The present manuscript proposed an NMPC (Nonlinear Model Predictive Control) algorithm for motion planning of multi-agent systems in an environment with fixed and moving obstacles. The focus was on a real application of safe navigation for ships in the Trondheim fjord. The classical non-convex constraints resulted from this problem were penalized in the cost function through appropriate potential field constructions. To reduce complexity and computation times we have considered on-off barrier functions as weights for the repulsive components of the potential field (such that only agents and obstacles in the view range of the current agent contribute into its potential field). Then, an NDO (Nonlinear Disturbance Observer)-based distributed NMPC scheme has been developed for motion planning problem with connectivity maintenance and collision avoidance for multi-surface vehicles. NDO was used to estimate the external disturbance and compensate for their impact via feedback control. Simulation results have shown the excellent performances of the algorithm as well as given insights towards real implementations.

We have to emphasize that the collision avoidance problem for multi-agent systems is a large and challenging domain. We have thus concentrated our contribution on a specific thematic related to the safe navigation of ships. The topic is complex enough as it covers different challenging issues appearing in the control of the multi-agent systems such as having underactuated dynamics and ensuring collision avoidance and connectivity maintenance, all of these in presence of external disturbances and in a cluttered environment.

Throughout the manuscript we have considered various tools. Chiefly among these, we consider polytopic sets to characterize obstacles, agents' safety regions and ground features (islands or mainland). We have also considered barrier functions and proposed an implementation based on the logistic regression (shown to behave better than, e.g., the signum function). These tools allowed us to write the various limitations (either as hard constraints for the MIP (Mixed-Integer Programming) approach or as penalty terms into the cost for the APF (Artificial Potential Field) approach) into a manageable form which was further integrated into a MPC (Model Predictive Control) framework. While in the beginning of the thesis we show both MIP and APF approaches, we have done so only to highlight the shortcomings of the former (in terms of computation time and feasibility) w.r.t. to the later.

Despite the fact that the potential field approach in MPC scheme showed some efficiencies in

obstacle avoidance, the local minima usually emerges as the main shortcoming which may affect the motion planning procedure. Therefore, we proposed a combined methodology between *on-off barrier functions* (is a technique borrowed from machine learning algorithms used in the field of statistics) and APF constructions to make sure that the magnitude of the total potential fields (repulsive and attractive potential field) is always variable in the view range of the agents (needed in order to reduce the probability of having null potential field which would lead to undesirable behavior for the agent). The proposed method was applied over a benchmark using real numerical data for the safe navigation of ships in the Trondheim fjord complying to COLREGS rules. The simulations have shown the effectiveness of the proposed method with respect to other classical methods.

Finally, we have developed a distributed control framework to deal with the motion planning problem for multiple surface vehicles in the presence of varying environmental disturbances. The goal was to steer the vehicles from an initial to a final destination while ensuring connectivity and avoiding collision with fixed and moving obstacles. Some ingredients like LOS (Line-of-Sight) guidance and RRT* (optimal Rapidly-exploring Random Tree) algorithms were first employed for generating a collision-free path considering static surroundings. The main contribution was in the employment of potential field constructions within a distributed NMPC framework for safe navigation in a dynamic coastal environment. We discussed the proposed algorithm through simulations with different scenarios and comparisons carried out over a benchmark for motion planning in the Trondheim fjord, Norway.

Since the combination between *on-off barrier function* and APF approach in NMPC scheme is a new approach for distributed motion planning of MAS topic, there are still many open questions that will be detailed in the last section.

5.2 Future developments

First of all, we consider the most relevant topic to be the one of time-delay systems[1] in the distributed motion planning framework of MAS. Delays in communication and in reaching a decision may strongly influence the collision avoidance guarantees. Not in the least, we may mention that the particular choice for expressing the repulsive component, the sum function, is not always well-posed. That is, it may happen that the sum function's derivatives have discontinuities which negatively influence the resulting cost function. Hence, alternatives to the sum function will be considered.

Secondly, although the *on-off repulsive potentials* are the main contribution in this manuscript, improvements can be carried out in order to better the results. Instead of assuming a fixed slope parameter β, we may obtain it through an estimation/fitting procedure which maximizes the appearance of the obstacles in the agent's view range, i.e., a collision risk may be alerted earlier. The *Newton-Raphson* [Bakari et al., 2016, Al-Daffaie and Khan, 2017] and *Gradient Descent* [Kim et al., 2018, Manogaran and Lopez, 2018] method as well as its variants such as *Batch Gradient Descent* and *Stochastic Gradient Descent* can be employed to estimate this parameter.

Thirdly, the additive disturbances need to be fully considered despite the fact that we took the impact of the environmental disturbances into account with respect to the control inputs (i.e.,

[1]Delays occur due to finite velocities of signal propagation or processing delays leading to memory effects and, in general, infinite-dimensional systems [Otto et al., 2019].

ocean currents affect the performance of the surge thrust and rudder deflection). That is, the effect of ocean currents defined in the inertial frame [Caharija et al., 2018] has not yet been taken into account. Furthermore, the impact of multiplicative disturbances should also be discussed. These factors cause delays in the communication channel, or degradation of the information exchanged among the agents. One of the most popular approaches for dealing with the issues in MPC framework is to employ the min−max MPC, which uses the maximum overall realizations of the disturbance sequence in the objective function [Gesser et al., 2018, Liu et al., 2018a, Hu and Ding, 2019]. Another interesting method is tube-based MPC [Villanueva et al., 2017, Sun et al., 2018b, Yadbantung and Bumroongsri, 2019] which guarantees that the actual dynamics track the nominal dynamic and are guaranteed to stay in a tube around it.

Finally, one of the most challenging perspectives is to guarantee asymptotic stability of the potential field-based MPC scheme by using the predicted input trajectory along the prediction horizon. Without doubt, this is not a trivial problem to deal with since MAS have to leave their predicted paths due to repulsive forces coming from the forbidden areas. Some well-known approaches from the literature are promising. On the one hand, stability can be ensured for finite-horizon problems by suitably choosing a weighting matrix for terminal cost and an attractive terminal region [Chen and Allgöwer, 1998, Nguyen et al., 2019]. On the other hand, closed-loop stability can also be achieved for relatively long horizons without the need to use a terminal cost or a terminal constraint [Grüne, 2012, Grüne and Pannek, 2017]. Stability is guaranteed by tuning the weighting matrices of the cost function.

Last but not least, while our results have been tested only in simulation, they consider realistic ship models, ocean disturbances and a benchmark inspired from real data (shore shape, ship movements, etc.). Thus, we are interested to continue the work and apply our results in real settings and/or more complex scenarios.

Appendix A

COLREGS

- Rule 8 - Action to avoid collision: avoidance action must be applied timely, before other vessel approaches. Any alterations of course and/or speed must be large enough to clear the approaching vessels.
- Rule 13 - Overtaking: The overtaking vessel can pass on either side and must keep out of the way of the vessel being overtaken. The vessel being overtaken must hold the course and speed until other vessel is past and well clear.
- Rule 14 - Head on situation: when two power-driven vessels are meeting on nearly reciprocal courses so as to involve risk of collision, then alter course to starboard so that each pass on the port side of each other.
- Rule 15 - Crossing situation: when two power-driven vessels are crossing so as to involve risk of collision, the vessel which has the other on her own starboard side shall keep out of the way.
- Rule 16 - Actions by give-way vessel: take early and substantial action to keep well clear.

Appendix B

Brief overview of RRT*

Algorithm 3 The Exploring-tree procedure.

Input: Nodes $K \in \mathbb{N}$, $\mathrm{x}_{\mathrm{init}} \in \mathcal{F}$, $\mathbb{X}_{goal} \subset \mathcal{F}$.
Output: tree $\mathscr{T}$.

$\mathcal{V} \leftarrow \mathscr{T}$.Node; $\mathcal{E} \leftarrow \mathscr{T}$.Edge;
$(\mathrm{x}_{\mathrm{near}}) \leftarrow \mathrm{Nearest}(\mathscr{T}, \mathrm{x}_{\mathrm{rand}})$;
$(\mathrm{x}_{\mathrm{new}}) \leftarrow \mathrm{Steer}(\mathrm{x}_{\mathrm{near}}, \mathrm{x}_{\mathrm{rand}}, \Delta t)$;
if Collision-free-path$(\mathrm{x}_{\mathrm{near}}, \mathrm{x}_{\mathrm{new}})$ **then**
  $\mathscr{T} \leftarrow \mathscr{T} \bigcup \{\mathrm{x}_{\mathrm{new}}\}$;
  $\mathrm{x}_{\mathrm{min}} \leftarrow \mathrm{x}_{\mathrm{near}}$; $\mathrm{C}_{\mathrm{min}} \leftarrow \mathrm{cost}(\mathrm{x}_{\mathrm{near}})$;
  $\mathrm{cost}(\mathrm{x}_{\mathrm{new}}) \leftarrow \mathrm{cost}(\mathrm{x}_{\mathrm{near}}) + \mathrm{dist}(\{\mathrm{x}_{\mathrm{near}}, \mathrm{x}_{\mathrm{new}}\})$;
  $x_{nearest} \leftarrow$ Check-nearest-neighbors$(\mathrm{x}_{\mathrm{rand}}, \mathcal{R})$;
  for all $\mathrm{x}_{\mathrm{nearest}} \in \mathbb{X}_{nearest}$ **do**
    if Collision-free-path$(\mathrm{x}_{\mathrm{nearest}}, x_{new})$ and $\mathrm{cost}(\mathrm{x}_{\mathrm{nearest}}) + \mathrm{dist}(\{\mathrm{x}_{\mathrm{nearest}}, \mathrm{x}_{\mathrm{new}}\}) < \mathrm{C}_{\mathrm{min}}$ **then**
      $\mathrm{x}_{\mathrm{min}} \leftarrow \mathrm{x}_{\mathrm{nearest}}$;
      $\mathrm{C}_{\mathrm{min}} \leftarrow \mathrm{cost}(\mathrm{x}_{\mathrm{nearest}}) + \mathrm{dist}(\{\mathrm{x}_{\mathrm{nearest}}, \mathrm{x}_{\mathrm{new}}\})$;
    end
  end
  $\mathcal{E} \leftarrow \mathcal{E} \bigcup \{(\mathrm{C}_{\mathrm{min}}, \mathrm{x}_{\mathrm{new}})\}$;
  for all $\mathrm{x}_{\mathrm{nearest}} \in \mathbb{X}_{nearest} \setminus \{\mathrm{x}_{\mathrm{min}}\}$ **do**
    if Collision-free-path$(x_{\mathrm{nearest}}, \mathrm{x}_{\mathrm{new}})$ and $\mathrm{cost}(\mathrm{x}_{\mathrm{nearest}}) > \mathrm{cost}(\mathrm{x}_{\mathrm{new}}) + \mathrm{dist}(\{\mathrm{x}_{\mathrm{nearest}}, \mathrm{x}_{\mathrm{new}}\})$; **then**
      $\mathrm{x}_{\mathrm{parent}} \leftarrow \mathrm{Parent}(\mathrm{x}_{\mathrm{nearest}})$;
      $\mathcal{E} \leftarrow \mathcal{E} \setminus \{(\mathrm{x}_{\mathrm{parent}}, \mathrm{x}_{\mathrm{nearest}})\}$;
      $\mathcal{E} \leftarrow \mathcal{E} \bigcup \{(\mathrm{x}_{\mathrm{nearest}}, \mathrm{x}_{\mathrm{new}})\}$;
    end
  end
end
Return $\mathscr{T} = (\mathcal{V}, \mathcal{E})$.

Appendix C

Solving local minima

Assume that the operating space of the i^{th} agent is the total potential field in classical APF approach, which is presented as (2.2.10):

$$S^c(\mathbf{x}_i) = \sum_{\ell=1}^{N_{\text{obs}}} \frac{c_{1\ell}}{(c_{2\ell} + \gamma_\ell(\mathbf{x}_i))^2} + \|\mathbf{x}_i - \mathbf{x}_{i,ref}\|^2_{\mathbf{Q}_i}, \tag{C.0.1}$$

The simplest control action would be a descent gradient method were the input is equated with the steepest gradient:

$$F_i^c(\mathbf{x}_i) = -\nabla S^c(\mathbf{x}_i) = -\sum_{\ell=1}^{N_{\text{obs}}} \left[\frac{c_{1\ell}}{(c_{2\ell}+\gamma_\ell(\mathbf{x}_i))^3}\right] \nabla \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i) - 2\mathbf{Q}_i(\mathbf{x}_i - \mathbf{x}_{i,ref})\nabla\mathbf{x}_i, \tag{C.0.2}$$

Local minima will happen when there is a bad choice of the parameters of repulsive and attractive potential making these potential components cancel each other. This means that (C.0.2) is equal to zero, i.e.,

$$-\sum_{\ell=1}^{N_{\text{obs}}} \left[\frac{c_{1\ell}}{(c_{2\ell}+\gamma_\ell(\mathbf{x}_i))^3}\right] \nabla \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i) = 2\mathbf{Q}_i(\mathbf{x}_i - \mathbf{x}_{i,ref})\nabla\mathbf{x}_i, \tag{C.0.3}$$

On the other hand, the proposed method gives us a total potential[1] described as follows:

$$S^p(\mathbf{x}_i) = \sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta p^s_{i,\ell}\|, \beta_{i\ell}) \frac{c_{1\ell}}{(c_{2\ell} + \gamma_\ell(\mathbf{x}_i))^2} + \|\mathbf{x}_i - \mathbf{x}_{i,ref}\|^2_{\mathbf{Q}_i}, \tag{C.0.4}$$

Similar with the classical approach, one has also:

$$\begin{aligned} F_i^p(\mathbf{x}_i) = -\nabla S^p(\mathbf{x}_i) = -\Bigg[&-\sum_{\ell=1}^{N_{\text{obs}}} \nabla\overline{F}_{i,\ell}(\|\Delta p^s_{i,\ell}\|, \beta_{i\ell}) \frac{c_{1\ell}}{(c_{2\ell} + \gamma_\ell(\mathbf{x}_i))^2} \\ &+ \sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta p^s_{i,\ell}\|, \beta_{i\ell}) \frac{c_{1\ell}}{(c_{2\ell} + \gamma_\ell(\mathbf{x}_i))^3} \nabla \sum_{\ell=1}^{N_{\text{obs}}} \gamma_\ell(\mathbf{x}_i) - 2\mathbf{Q}_i(\mathbf{x}_i - \mathbf{x}_{i,ref})\nabla\mathbf{x}_i \Bigg] \end{aligned} \tag{C.0.5}$$

(C.0.5) will become (C.0.2) (i.e., local minima) if and only if $\sum_{\ell=1}^{N_{\text{obs}}} \overline{F}_{i,\ell}(\|\Delta p^s_{i,\ell}\|, \beta_{i\ell}) = 1$ and $\sum_{\ell=1}^{N_{\text{obs}}} \nabla\overline{F}_{i,\ell}(\|\Delta p^s_{i,\ell}\|, \beta_{i\ell}) = 0$, i.e., entire total static repulsive potential must be in the interior of

[1] It is worth noting that all the parameters of total the potential fields are similar.

the view range of i^{th} agent (i.e., all static repulsive must be activated). This cannot happen since the view range of the agent is significantly smaller than the size of the workspace under consideration. Therefore, (C.0.5) always differs from (C.0.2), in other words, the proposed method can better deal with local minima in comparison to the classical approach.

Bibliography

[Aarts et al., 2005] Aarts, E., Korst, J., and Michiels, W. (2005). Simulated annealing. In *Search methodologies*, pages 187–210. Springer. 4

[Abdelaal et al., 2018] Abdelaal, M., Fränzle, M., and Hahn, A. (2018). Nonlinear model predictive control for trajectory tracking and collision avoidance of underactuated vessels with disturbances. *Ocean Engineering*, 160:168–180. 77, 78

[Abichandani et al., 2015] Abichandani, P., Torabi, S., Basu, S., and Benson, H. (2015). Mixed integer nonlinear programming framework for fixed path coordination of multiple underwater vehicles under acoustic communication constraints. *IEEE Journal of Oceanic Engineering*, 40(4):864–873. 46

[Aghababa et al., 2012] Aghababa, M. P., Amrollahi, M. H., and Borjkhani, M. (2012). Application of ga, pso, and aco algorithms to path planning of autonomous underwater vehicles. *Journal of Marine Science and Application*, 11(3):378–386. 45

[Aguilar et al., 2016] Aguilar, J. V., Langarita, P., Rodellar, J., Linares, L., and Horvath, K. (2016). Predictive control of irrigation canals–robust design and real-time implementation. *Water resources management*, 30(11):3829–3843. 5

[Al-Daffaie and Khan, 2017] Al-Daffaie, K. and Khan, S. (2017). Logistic regression for circular data. In *AIP Conference Proceedings*, volume 1842, page 030022. AIP Publishing. 102

[Allgöwer and Zheng, 2012] Allgöwer, F. and Zheng, A. (2012). *Nonlinear model predictive control*, volume 26. Birkhäuser. 5, 28

[Altinoz et al., 2017] Altinoz, O. T., Yanar, T. A., Ozguven, C., Demirdogen, G., and Yilmaz, A. E. (2017). Improved non-probabilistic roadmap method for determination of shortest nautical navigation path. In *2017 4th International Conference on Electrical and Electronic Engineering (ICEEE)*, pages 261–266. IEEE. 76

[Ames et al., 2019] Ames, A. D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). Control barrier functions: Theory and applications. *arXiv preprint arXiv:1903.11199*. 15

[Ames et al., 2014] Ames, A. D., Grizzle, J. W., and Tabuada, P. (2014). Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278. IEEE. 31, 32, 33

[Ames et al., 2017] Ames, A. D., Xu, X., Grizzle, J. W., and Tabuada, P. (2017). Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876. 43

[Andersson et al., 2019] Andersson, J. A., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (2019). Casadi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36. 9, 31

[Andersson et al., 2018] Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (In Press, 2018). CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*. 91

[Appapogu, 2019] Appapogu, R. D. (2019). *Autonomous navigation in GPS denied environments using MPC and LQR with potential field based obstacle avoidance*. PhD thesis, Colorado School of Mines. Arthur Lakes Library. 16

[Arzamendia et al., 2019] Arzamendia, M., Gregor, D., Reina, D. G., and Toral, S. L. (2019). An evolutionary approach to constrained path planning of an autonomous surface vehicle for maximizing the covered area of ypacarai lake. *Soft Computing*, 23(5):1723–1734. 45

[Baek et al., 2018] Baek, D., Hwang, M., Kim, H., and Kwon, D.-S. (2018). Path planning for automation of surgery robot based on probabilistic roadmap and reinforcement learning. In *2018 15th International Conference on Ubiquitous Robots (UR)*, pages 342–347. IEEE. 4

[Baillard et al., 2018] Baillard, V., Goy, A., Vasselin, N., and Maniu, C. S. (2018). Potential field based optimization of a prey-predator multi-agent system. In *9th Vienna Conference on Mathematical Modelling*. ARGESIM Publisher Vienna. 5

[Bakari et al., 2016] Bakari, H., Adegoke, T., and Yahya, A. (2016). Application of newton rhapson method to non-linear models. *International Journal of Mathematics and Statistics Studies*, 4(4):21–31. 102

[Blanchini, 1999] Blanchini, F. (1999). Set invariance in control. *Automatica*, 35(11):1747–1767. 15

[Bock et al., 2007] Bock, H. G., Diehl, M., Kühl, P., Kostina, E., Schiöder, J. P., and Wirsching, L. (2007). Numerical methods for efficient and fast nonlinear model predictive control. In *Assessment and future directions of nonlinear model predictive control*, pages 163–179. Springer. 5

[Bock and Plitt, 1984] Bock, H. G. and Plitt, K.-J. (1984). A multiple shooting algorithm for direct solution of optimal control problems. *IFAC Proceedings Volumes*, 17(2):1603–1608. 28

[Bonthu et al., 2019] Bonthu, R. K., Aguilera, R. P., Pham, H., Phung, M. D., and Ha, Q. P. (2019). Energy cost optimization in microgrids using model predictive control and mixed integer linear programming. In *International Conference on Industrial Technology*. 6

[Børhaug et al., 2010] Børhaug, E., Pavlov, A., Panteley, E., and Pettersen, K. Y. (2010). Straight line path following for formations of underactuated marine surface vessels. *IEEE Transactions on Control Systems Technology*, 19(3):493–506. 99

[Borrmann et al., 2015] Borrmann, U., Wang, L., Ames, A. D., and Egerstedt, M. (2015). Control barrier certificates for safe swarm behavior. *IFAC-PapersOnLine*, 48(27):68–73. 34

[Boyd and Vandenberghe, 2004] Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press. 17, 32, 55

[Breivik, 2003] Breivik, M. (2003). Nonlinear maneuvering control of underactuated ships. *MS thesis*. 48, 81

[Breivik et al., 2017] Breivik, M. et al. (2017). Mpc-based mid-level collision avoidance for asvs using nonlinear programming. In *Control Technology and Applications (CCTA), 2017 IEEE Conference on*, pages 766–772. IEEE. 46, 67, 72, 73, 77

[Breivik and Fossen, 2004] Breivik, M. and Fossen, T. I. (2004). Path following for marine surface vessels. In *Oceans' 04 MTS/IEEE Techno-Ocean'04 (IEEE Cat. No. 04CH37600)*, volume 4, pages 2282–2289. IEEE. 99

[Breivik and Fossen, 2008] Breivik, M. and Fossen, T. I. (2008). Guidance laws for planar motion control. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 570–577. IEEE. 82

[Cafieri and Rey, 2017] Cafieri, S. and Rey, D. (2017). Maximizing the number of conflict-free aircraft using mixed-integer nonlinear programming. *Computers & Operations Research*, 80:147–158. 46

[Caharija et al., 2018] Caharija, W., Grøtli, E. I., and Pettersen, K. Y. (2018). Semiglobal exponential stability of a counter-current and co-current guidance scheme. *IFAC-PapersOnLine*, 51(29):274–280. 103

[Cairano et al., 2007] Cairano, S. D., Bemporad, A., Kolmanovsky, I. V., and Hrovat, D. (2007). Model predictive control of magnetically actuated mass spring dampers for automotive applications. *International Journal of Control*, 80(11):1701–1716. 5

[Camacho and Bordons, 2004] Camacho, E. F. and Bordons, C. (2004). Model predictive control. advanced textbooks in control and signal processing. *Springer-Verlag, London*. 18

[Camponogara et al., 2002] Camponogara, E., Jia, D., Krogh, B. H., and Talukdar, S. (2002). Distributed model predictive control. *IEEE control systems magazine*, 22(1):44–52. 9

[Cao et al., 2013] Cao, Y., Yu, W., Ren, W., and Chen, G. (2013). An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9(1):427–438. 7

[Cetin and Yilmaz, 2016] Cetin, O. and Yilmaz, G. (2016). Real-time autonomous uav formation flight with collision and obstacle avoidance in unknown environment. *Journal of Intelligent & Robotic Systems*, 84(1-4):415–433. 5

[Chao et al., 2017] Chao, W., Feng, M., Qing, W., and Shuwu, W. (2017). A situation awareness approach for usv based on artificial potential fields. In *2017 4th International Conference on Transportation Information and Safety (ICTIS)*, pages 232–235. IEEE. 76

[Chen et al., 2013] Chen, C., Wang, J., Heo, Y., and Kishore, S. (2013). Mpc-based appliance scheduling for residential building energy management controller. *IEEE Transactions on Smart Grid*, 4(3):1401–1410. 28

[Chen and Ye, 2012] Chen, C.-Y. and Ye, F. (2012). Particle swarm optimization algorithm and its application to clustering analysis. In *2012 Proceedings of 17th Conference on Electrical Power Distribution*, pages 789–794. IEEE. 4

[Chen and Allgöwer, 1998] Chen, H. and Allgöwer, F. (1998). A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217. 29, 62, 103

[Chen et al., 2018a] Chen, L., Hopman, H., and Negenborn, R. R. (2018a). Distributed model predictive control for vessel train formations of cooperative multi-vessel systems. *Transportation Research Part C: Emerging Technologies*, 92:101–118. 9

[Chen et al., 2019] Chen, L., Huang, Y., Zheng, H., Hopman, H., and Negenborn, R. (2019). Cooperative multi-vessel systems in urban waterway networks. *IEEE Transactions on Intelligent Transportation Systems*. 99

[Chen et al., 2018b] Chen, L., Negenborn, R. R., and Hopman, H. (2018b). Intersection crossing of cooperative multi-vessel systems. *IFAC-PapersOnLine*, 51(9):379–385. 9

[Chen et al., 2000] Chen, W.-H., Ballance, D. J., Gawthrop, P. J., and O'Reilly, J. (2000). A nonlinear disturbance observer for robotic manipulators. *IEEE Transactions on industrial Electronics*, 47(4):932–938. 84

[Chen et al., 2018c] Chen, X., Liu, Y., Hong, X., Wei, X., and Huang, Y. (2018c). Unmanned ship path planning based on rrt. In *International Conference on Intelligent Computing*, pages 102–110. Springer. 77

[Chen et al., 2016] Chen, Y.-b., Luo, G.-c., Mei, Y.-s., Yu, J.-q., and Su, X.-l. (2016). Uav path planning using artificial potential field method updated by optimal control theory. *International Journal of Systems Science*, 47(6):1407–1420. 16

[Chen and Wang, 2005] Chen, Y. Q. and Wang, Z. (2005). Formation control: a review and a new consideration. In *2005 IEEE/RSJ International conference on intelligent robots and systems*, pages 3181–3186. IEEE. 3

[Chiang and Tapia, 2018] Chiang, H.-T. L. and Tapia, L. (2018). Colreg-rrt: an rrt-based colregs-compliant motion planner for surface vehicle navigation. *IEEE Robotics and Automation Letters*, 3(3):2024–2031. 46

[Christofides et al., 2013] Christofides, P. D., Scattolini, R., de la Pena, D. M., and Liu, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering*, 51:21–41. 8

[Clark, 2005] Clark, C. M. (2005). Probabilistic road map sampling strategies for multi-robot motion planning. *Robotics and Autonomous Systems*, 53(3-4):244–264. 4

[Commandant, 1999] Commandant, U. (1999). International regulations for prevention of collisions at sea, 1972 (72 colregs). *US Department of Transportation, US Coast Guard, COMMANDANT INSTRUCTION M*, 16672. 45, 76

[Cui et al., 2015] Cui, R., Li, Y., and Yan, W. (2015). Mutual information-based multi-auv path planning for scalar field sampling using multidimensional rrt. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(7):993–1004. 46

[Dalpe and Thein, 2017] Dalpe, A. and Thein, M.-W. (2017). Obstacle avoidance strategies for autonomous surface vehicles. In *OCEANS 2017-Anchorage*, pages 1–8. IEEE. 5

[Davis, 1987] Davis, L. (1987). Genetic algorithms and simulated annealing. 4

[Deits and Tedrake, 2015] Deits, R. and Tedrake, R. (2015). Efficient mixed-integer planning for uavs in cluttered environments. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 42–49. IEEE. 6

[Deo, 2017] Deo, N. (2017). *Graph theory with applications to engineering and computer science*. Courier Dover Publications. 8

[Di Cairano et al., 2010] Di Cairano, S., Tseng, H., Bernardini, D., and Bemporad, A. (2010). Steering vehicle control by switched model predictive control. *IFAC Proceedings Volumes*, 43(7):1–6. 5

[Di Cairano et al., 2008] Di Cairano, S., Yanakiev, D., Bemporad, A., Kolmanovsky, I. V., and Hrovat, D. (2008). An mpc design flow for automotive control and applications to idle speed regulation. In *2008 47th IEEE Conference on Decision and Control*, pages 5686–5691. IEEE. 5

[Diehl et al., 2009] Diehl, M., Ferreau, H. J., and Haverbeke, N. (2009). Efficient numerical methods for nonlinear mpc and moving horizon estimation. In *Nonlinear model predictive control*, pages 391–417. Springer. 5

[Ding et al., 2017] Ding, L., Han, Q.-L., Ge, X., and Zhang, X.-M. (2017). An overview of recent advances in event-triggered consensus of multiagent systems. *IEEE transactions on cybernetics*, 48(4):1110–1123. 8

[Do, 2010] Do, K. D. (2010). Practical control of underactuated ships. *Ocean Engineering*, 37(13):1111–1119. 78, 84

[Dombrovskii and Obyedko, 2015] Dombrovskii, V. and Obyedko, T. (2015). Model predictive control for constrained systems with serially correlated stochastic parameters and portfolio optimization. *Automatica*, 54:325–331. 28

[Doria et al., 2013] Doria, N. S. F., Freire, E. O., and Basilio, J. C. (2013). An algorithm inspired by the deterministic annealing approach to avoid local minima in artificial potential fields. In *2013 16th International Conference on Advanced Robotics (ICAR)*, pages 1–6. IEEE. 46

[Dorigo and Birattari, 2010] Dorigo, M. and Birattari, M. (2010). *Ant colony optimization*. Springer. 4

[Dunbar and Murray, 2006] Dunbar, W. B. and Murray, R. M. (2006). Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558. 9

[Eberhart and Kennedy, 1995] Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE. 4

[Eriksen et al., 2019] Eriksen, B.-O. H., Breivik, M., Wilthil, E. F., Flåten, A. L., and Brekke, E. F. (2019). The branching-course model predictive control algorithm for maritime collision avoidance. *Journal of Field Robotics*, 36(7):1222–1249. 48

[Fang et al., 2017] Fang, H., Wei, Y., Chen, J., and Xin, B. (2017). Flocking of second-order multiagent systems with connectivity preservation based on algebraic connectivity estimation. *IEEE transactions on cybernetics*, 47(4):1067–1077. 77

[Ferguson and Stentz, 2006] Ferguson, D. and Stentz, A. (2006). Anytime rrts. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5369–5375. IEEE. 5

[Ferreau et al., 2014] Ferreau, H. J., Kirches, C., Potschka, A., Bock, H. G., and Diehl, M. (2014). qpoases: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363. 31

[Filotheou et al., 2018] Filotheou, A., Nikou, A., and Dimarogonas, D. V. (2018). Decentralized control of uncertain multi-agent systems with connectivity maintenance and collision avoidance. In *2018 European Control Conference (ECC)*, pages 8–13. IEEE. 77

[Findeisen et al., 2007] Findeisen, R., Allgöwer, F., and Biegler, L. T. (2007). *Assessment and future directions of nonlinear model predictive control*, volume 358. Springer. 28

[Fontana et al., 2017] Fontana, N., Giugni, M., Glielmo, L., Marini, G., and Zollo, R. (2017). Real-time control of pressure for leakage reduction in water distribution network: Field experiments. *Journal of Water Resources Planning and Management*, 144(3):04017096. 3

[Fossen, 2002] Fossen, T. I. (2002). Marine control system-guidance, navigation and control of ships, rigs and underwater vehicles. *Marine Cybemetics*. 48

[Fossen, 2011] Fossen, T. I. (2011). *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons. 48

[Fu et al., 2018] Fu, B., Wang, Q., and He, W. (2018). Nonlinear disturbance observer-based control for a class of port-controlled hamiltonian disturbed systems. *IEEE Access*, 6:50299–50305. 99

[Gao and Guo, 2019] Gao, Z. and Guo, G. (2019). Command filtered path tracking control of saturated asvs based on time-varying disturbance observer. *Asian Journal of Control*. 99

[Garcia et al., 1989] Garcia, C. E., Prett, D. M., and Morari, M. (1989). Model predictive control: theory and practicea survey. *Automatica*, 25(3):335–348. 5

[Garey and Johnson, 2002] Garey, M. R. and Johnson, D. S. (2002). *Computers and intractability*, volume 29. wh freeman New York. 6

[Ge and Han, 2017] Ge, X. and Han, Q.-L. (2017). Distributed formation control of networked multi-agent systems using a dynamic event-triggered communication mechanism. *IEEE Transactions on Industrial Electronics*, 64(10):8118–8127. 8

[Ge et al., 2018] Ge, X., Han, Q.-L., Ding, D., Zhang, X.-M., and Ning, B. (2018). A survey on recent advances in distributed sampled-data cooperative control of multi-agent systems. *Neurocomputing*, 275:1684–1701. 76

[Gesser et al., 2018] Gesser, R. S., Lima, D. M., and Normey-Rico, J. E. (2018). Robust model predictive control: Implementation issues with comparative analysis. *IFAC-PapersOnLine*, 51(25):478–483. 103

[Godsil and Royle, 2013] Godsil, C. and Royle, G. F. (2013). *Algebraic graph theory*, volume 207. Springer Science & Business Media. 8

[Goerzen et al., 2010] Goerzen, C., Kong, Z., and Mettler, B. (2010). A survey of motion planning algorithms from the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems*, 57(1-4):65. 7

[Grüne, 2012] Grüne, L. (2012). Nmpc without terminal constraints. *IFAC Proceedings Volumes*, 45(17):1–13. 103

[Grüne and Pannek, 2017] Grüne, L. and Pannek, J. (2017). Nonlinear model predictive control. In *Nonlinear Model Predictive Control*, pages 45–69. Springer. 103

[Guerreiro et al., 2014] Guerreiro, B. J., Silvestre, C., Cunha, R., and Pascoal, A. (2014). Trajectory tracking nonlinear model predictive control for autonomous surface craft. *IEEE Transactions on Control Systems Technology*, 22(6):2160–2175. 99

[Guerrero and Lozano, 2012] Guerrero, J. A. and Lozano, R. (2012). *Flight formation control*. Wiley Online Library. 3

[Guo et al., 2019] Guo, H., Cao, D., Chen, H., Sun, Z., and Hu, Y. (2019). Model predictive path following control for autonomous cars considering a measurable disturbance: Implementation, testing, and verification. *Mechanical Systems and Signal Processing*, 118:41–60. 2

[Hagen et al., 2018] Hagen, I. B., Kufoalor, D. K. M., Brekke, E. F., and Johansen, T. A. (2018). Mpc-based collision avoidance strategy for existing marine vessel guidance systems. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7618–7623. IEEE. 99

[Han et al., 2017] Han, Y., Zhang, K., Li, H., Coelho, E. A. A., and Guerrero, J. M. (2017). Mas-based distributed coordinated control and optimization in microgrid and microgrid clusters: A comprehensive overview. *IEEE Transactions on Power Electronics*, 33(8):6488–6508. 8

[He et al., 2019a] He, S., Wang, M., Dai, S.-L., and Luo, F. (2019a). Leader–follower formation control of usvs with prescribed performance and collision avoidance. *IEEE Transactions on Industrial Informatics*, 15(1):572–581. 3

[He et al., 2019b] He, X., Kou, G., Calaf, M., and Leang, K. K. (2019b). In-ground-effect modeling and nonlinear-disturbance observer for multirotor unmanned aerial vehicle control. *Journal of Dynamic Systems, Measurement, and Control*, 141(7):071013. 99

[Hicks and Ray, 1971] Hicks, G. and Ray, W. (1971). Approximation methods for optimal control synthesis. *The Canadian Journal of Chemical Engineering*, 49(4):522–528. 28

[Hinostroza et al., 2018] Hinostroza, M., Guedes Soares, C., and Xu, H. (2018). Motion planning, guidance and control system for autonomous surface vessel. In *ASME 2018 37th International Conference on Ocean, Offshore and Arctic Engineering*. American Society of Mechanical Engineers Digital Collection. 99

[Hosmer and Lemesbow, 1980] Hosmer, D. W. and Lemesbow, S. (1980). Goodness of fit tests for the multiple logistic regression model. *Communications in statistics-Theory and Methods*, 9(10):1043–1069. 10

[Hou et al., 2018] Hou, X., Sun, Y., Lu, J., Zhang, X., Koh, L. H., Su, M., and Guerrero, J. M. (2018). Distributed hierarchical control of ac microgrid operating in grid-connected, islanded and their transition modes. *Ieee Access*, 6:77388–77401. 7

[Houska et al., 2011] Houska, B., Ferreau, H. J., and Diehl, M. (2011). Acado toolkitan open-source framework for automatic control and dynamic optimization. *Optimal Control Applications and Methods*, 32(3):298–312. 9, 31

[Hsu et al., 1998] Hsu, D., Kavraki, L. E., Latombe, J.-C., Motwani, R., Sorkin, S., et al. (1998). On finding narrow passages with probabilistic roadmap planners. In *Robotics: The Algorithmic Perspective: 1998 Workshop on the Algorithmic Foundations of Robotics*, pages 141–154. 4

[Hu and Ding, 2019] Hu, J. and Ding, B. (2019). An efficient offline implementation for output feedback min-max mpc. *International Journal of Robust and Nonlinear Control*, 29(2):492–506. 103

[Hu et al., 2017] Hu, L., Naeem, W., Rajabally, E., Watson, G., Mills, T., Bhuiyan, Z., and Salter, I. (2017). Colregs-compliant path planning for autonomous surface vehicles: A multi-objective optimization approach. *IFAC-PapersOnLine*, 50(1):13662–13667. 45

[Hu and Yang, 2004] Hu, Y. and Yang, S. X. (2004). A knowledge based genetic algorithm for path planning of a mobile robot. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, volume 5, pages 4350–4355. IEEE. 4

[Huang et al., 2012] Huang, G.-B., Zhou, H., Ding, X., and Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529. 47, 52

[Huang et al., 2018] Huang, Z., Chu, D., Wu, C., and He, Y. (2018). Path planning and cooperative control for automated vehicle platoon using hybrid automata. *IEEE Transactions on Intelligent Transportation Systems*, 20(3):959–974. 9

[IBM, 1999] IBM (1999). Ilog cplex users guide. https://www.ibm.com/products/ilog-cplex-optimization-studio. 9

[Ioan et al., 2019a] Ioan, D., Olaru, S., Prodan, I., Stoican, F., and Niculescu, S.-I. (2019a). From obstacle-based space partitioning to corridors and path planning. a convex lifting approach. *IEEE Control Systems Letters*, 4(1):79–84. 6

[Ioan et al., 2019b] Ioan, D., Prodan, I., Stoican, F., Olaru, S., and Niculescu, S.-I. (2019b). Complexity bounds for obstacle avoidance within a zonotopic framework. 6

[Jeroslow, 1973] Jeroslow, R. C. (1973). There cannot be any algorithm for integer programming with quadratic constraints. *Operations Research*, 21(1):221–224. 31

[Jiang et al., 2019] Jiang, Y., Peng, Z., Wang, D., and Chen, C. P. (2019). Line-of-sight target enclosing of an underactuated autonomous surface vehicle with experiment results. *IEEE Transactions on Industrial Informatics*. 99

[Johansen, 2011] Johansen, T. A. (2011). Introduction to nonlinear model predictive control and moving horizon estimation. *Selected topics on constrained and nonlinear control*, 1:1–53. 29

[Johansen et al., 2016] Johansen, T. A., Perez, T., and Cristofaro, A. (2016). Ship collision avoidance and colregs compliance using simulation-based control behavior selection with predictive hazard assessment. *IEEE transactions on intelligent transportation systems*, 17(12):3407–3422. 46, 77

[Jünger et al., 2009] Jünger, M., Liebling, T. M., Naddef, D., Nemhauser, G. L., Pulleyblank, W. R., Reinelt, G., Rinaldi, G., and Wolsey, L. A. (2009). *50 Years of integer programming 1958-2008: From the early years to the state-of-the-art*. Springer Science & Business Media. 16

[Karaman and Frazzoli, 2011] Karaman, S. and Frazzoli, E. (2011). Sampling-based algorithms for optimal motion planning. *The international journal of robotics research*, 30(7):846–894. 79

[Karami and Hasanzadeh, 2015] Karami, A. H. and Hasanzadeh, M. (2015). An adaptive genetic algorithm for robot motion planning in 2d complex environments. *Computers & Electrical Engineering*, 43:317–329. 4

[Katriniok et al., 2019] Katriniok, A., Sopasakis, P., Schuurmans, M., and Patrinos, P. (2019). Nonlinear model predictive control for distributed motion planning in road intersections using panoc. *arXiv preprint arXiv:1903.12091*. 9

[Kavraki et al., 1994] Kavraki, L., Svestka, P., and Overmars, M. H. (1994). *Probabilistic roadmaps for path planning in high-dimensional configuration spaces*, volume 1994. Unknown Publisher. 4

[Ke et al., 2018] Ke, F., Li, Z., and Yang, C. (2018). Robust tube-based predictive control for visual servoing of constrained differential-drive mobile robots. *IEEE Transactions on Industrial Electronics*, 65(4):3437–3446. 77

[Khalil and Grizzle, 2002] Khalil, H. K. and Grizzle, J. W. (2002). *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ. 33

[Khatib, 1986] Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous robot vehicles*, pages 396–404. Springer. 5, 21

[Kim et al., 2017] Kim, D., Kim, H., and Huh, K. (2017). Local trajectory planning and control for autonomous vehicles using the adaptive potential field. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 987–993. IEEE. 16

[Kim et al., 2018] Kim, M., Song, Y., Wang, S., Xia, Y., and Jiang, X. (2018). Secure logistic regression based on homomorphic encryption: Design and evaluation. *JMIR medical informatics*, 6(2):e19. 102

[Kneebone et al., 2006] Kneebone, M., Barnden, J., and Wyatt, J. (2006). Probabilistic roadmaps for unknown kinodynamic constraints. 76

[Koditschek, 1992] Koditschek, D. E. (1992). Task encoding: Toward a scientific paradigm for robot planning and control. *Robotics and autonomous systems*, 9(1-2):5–39. 5

[Koren and Borenstein, 1991] Koren, Y. and Borenstein, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 1398–1404. IEEE. 16

[Kothari and Postlethwaite, 2013] Kothari, M. and Postlethwaite, I. (2013). A probabilistically robust path planning algorithm for uavs using rapidly-exploring random trees. *Journal of Intelligent & Robotic Systems*, 71(2):231–253. 5

[Kraft, 1985] Kraft, D. (1985). On converting optimal control problems into nonlinear programming problems. In *Computational mathematical programming*, pages 261–280. Springer. 28

[Krejsa and Věchet, 2005] Krejsa, J. and Věchet, S. (2005). Rapidly exploring random trees used for mobile robots path planning. *Engineering Mechanics*, 12(4):231–237. 5

[Kurzhanski and Varaiya, 2000] Kurzhanski, A. B. and Varaiya, P. (2000). Ellipsoidal techniques for reachability analysis. In *International Workshop on Hybrid Systems: Computation and Control*, pages 202–214. Springer. 15

[Kuwata and How, 2010] Kuwata, Y. and How, J. P. (2010). Cooperative distributed robust trajectory optimization using receding horizon milp. *IEEE Transactions on Control Systems Technology*, 19(2):423–431. 9

[Kuwata et al., 2011] Kuwata, Y., Wolf, M. T., Zarzhitsky, D., and Huntsberger, T. L. (2011). Safe maritime navigation with colregs using velocity obstacles. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4728–4734. IEEE. 77

[Kwon and Han, 2006] Kwon, W. H. and Han, S. H. (2006). *Receding horizon control: model predictive control for state models*. Springer Science & Business Media. 5

[Kyurkchiev and Markov, 2016] Kyurkchiev, N. and Markov, S. (2016). Approximation of the cut function by some generic logistic functions and applications. *Advances in Applied Sciences*, 1(2):24–29. 52

[Larson et al., 2007] Larson, J., Bruch, M., Halterman, R., Rogers, J., and Webster, R. (2007). Advances in autonomous obstacle avoidance for unmanned surface vehicles. Technical report, SPACE AND NAVAL WARFARE SYSTEMS CENTER SAN DIEGO CA. 46

[Latombe, 2012] Latombe, J.-C. (2012). *Robot motion planning*, volume 124. Springer Science & Business Media. 16

[LaValle, 1998] LaValle, S. M. (1998). Rapidly-exploring random trees: A new tool for path planning. 4

[Lee and Park, 2003] Lee, M. C. and Park, M. G. (2003). Artificial potential field based path planning for mobile robots using a virtual obstacle concept. In *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, volume 2, pages 735–740. IEEE. 46

[Leineweber et al., 2003] Leineweber, D. B., Bauer, I., Bock, H. G., and Schlöder, J. P. (2003). An efficient multiple shooting based reduced sqp strategy for large-scale dynamic process optimization. part 1: theoretical aspects. *Computers & Chemical Engineering*, 27(2):157–166. 28

[Leitão, 2009] Leitão, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*, 22(7):979–991. 7

[Leonard and Fiorelli, 2001] Leonard, N. E. and Fiorelli, E. (2001). Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, volume 3, pages 2968–2973. IEEE. 5

[Li et al., 2013] Li, G., Tamura, Y., Yamashita, A., and Asama, H. (2013). Effective improved artificial potential field-based regression search method for autonomous mobile robot path planning. *International Journal of Mechatronics and Automation*, 3(3):141–170. 46

[Li et al., 2012a] Li, G., Yamashita, A., Asama, H., and Tamura, Y. (2012a). An efficient improved artificial potential field based regression search method for robot path planning. In *2012 IEEE International Conference on Mechatronics and Automation*, pages 1227–1232. IEEE. 46

[Li et al., 2012b] Li, Q., Wang, L.-J., Chen, B., Zhou, Z., and Yin, Y.-X. (2012b). An improved artificial potential field method with parameters optimization based on genetic algorithms. *Journal of University of science and technology BEIJING*, 34(2):202–206. 46

[Li et al., 2017] Li, W., Yang, C., Jiang, Y., Liu, X., and Su, C.-Y. (2017). Motion planning for omnidirectional wheeled mobile robot by potential field method. *Journal of Advanced Transportation*, 2017. 16

[Li et al., 2018] Li, Y., Cui, R., Li, Z., and Xu, D. (2018). Neural network approximation based near-optimal motion planning with kinodynamic constraints using rrt. *IEEE Transactions on Industrial Electronics*, 65(11):8718–8729. 77

[Lim et al., 2018] Lim, W., Lee, S., Sunwoo, M., and Jo, K. (2018). Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method. *IEEE Transactions on Intelligent Transportation Systems*, 19(2):613–626. 2

[Limón et al., 2008] Limón, D., Alvarado, I., Alamo, T., and Camacho, E. F. (2008). Mpc for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9):2382–2387. 28

[Linderoth and Savelsbergh, 1999] Linderoth, J. T. and Savelsbergh, M. W. (1999). A computational study of search strategies for mixed integer programming. *INFORMS Journal on Computing*, 11(2):173–187. 6

[Liu et al., 2011] Liu, C., Chen, W.-H., and Andrews, J. (2011). Trajectory tracking of small helicopters using explicit nonlinear mpc and dobc. *IFAC Proceedings Volumes*, 44(1):1498–1503. 78

[Liu et al., 2018a] Liu, C., Li, H., Gao, J., and Xu, D. (2018a). Robust self-triggered min–max model predictive control for discrete-time nonlinear systems. *Automatica*, 89:333–339. 103

[Liu et al., 2015] Liu, C., Zheng, H., Negenborn, R. R., Chu, X., and Wang, L. (2015). Trajectory tracking control for underactuated surface vessels based on nonlinear model predictive control. In *International Conference on Computational Logistics*, pages 166–180. Springer. 46

[Liu et al., 2006] Liu, G., Mu, J., Rees, D., and Chai, S. (2006). Design and stability analysis of networked control systems with random communication time delay using the modified mpc. *International Journal of Control*, 79(4):288–297. 5

[Liu et al., 2010] Liu, J., Chen, X., Muñoz de la Peña, D., and Christofides, P. D. (2010). Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal*, 56(8):2137–2149. 8, 89

[Liu et al., 2018b] Liu, Y., Huang, P., Zhang, F., and Zhao, Y. (2018b). Distributed formation control using artificial potentials and neural network for constrained multiagent systems. *IEEE Transactions on Control Systems Technology*. 9

[Liu et al., 2017] Liu, Z., Geng, C., and Zhang, J. (2017). Model predictive controller design with disturbance observer for path following of unmanned surface vessel. In *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1827–1832. IEEE. 78

[Löfberg, 2004] Löfberg, J. (2004). Yalmip: A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference*, volume 3. Taipei, Taiwan. 9, 31

[Luis and Schoellig, 2019] Luis, C. E. and Schoellig, A. P. (2019). Trajectory generation for multiagent point-to-point transitions via distributed model predictive control. *IEEE Robotics and Automation Letters*, 4(2):375–382. 9

[Luo et al., 2012] Luo, G.-c., Yu, J.-q., Zhang, S.-y., and Zhang, W. (2012). Artificial potential field based receding horizon control for path planning. In *2012 24th Chinese Control and Decision Conference (CCDC)*, pages 3665–3669. IEEE. 46

[Maciejowski, 2002] Maciejowski, J. M. (2002). *Predictive control: with constraints*. Pearson education. 5

[Maghenem et al., 2018] Maghenem, M., Loría, A., and Panteley, E. (2018). Formation-tracking control of autonomous vehicles under relaxed persistency of excitation conditions. *IEEE Transactions on Control Systems Technology*, 26(5):1860–1865. 3

[Magni et al., 2001] Magni, L., De Nicolao, G., and Scattolini, R. (2001). Output feedback and tracking of nonlinear systems with model predictive control. *Automatica*, 37(10):1601–1607. 28

[Manogaran and Lopez, 2018] Manogaran, G. and Lopez, D. (2018). Health data analytics using scalable logistic regression with stochastic gradient descent. *International Journal of Advanced Intelligence Paradigms*, 10(1-2):118–132. 102

[Mayne et al., 2000] Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814. 5

[Mayne et al., 2005] Mayne, D. Q., Seron, M. M., and Raković, S. (2005). Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224. 5, 77

[McNinch et al., 2008] McNinch, L. C., Muske, K. R., and Ashrafiuon, H. (2008). Model-based predictive control of an unmanned surface vessel. In *Proceedings of the 11th IASTED International Conference on Intelligent Systems and Control*, pages 385–390. 46

[Mei et al., 2015] Mei, J., Ren, W., and Chen, J. (2015). Distributed consensus of second-order multi-agent systems with heterogeneous unknown inertias and control gains under a directed graph. *IEEE Transactions on Automatic Control*, 61(8):2019–2034. 8

[Mendes Filho et al., 2017] Mendes Filho, J. M., Lucet, E., and Filliat, D. (2017). Real-time distributed receding horizon motion planning and control for mobile multi-robot dynamic systems. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 657–663. IEEE. 9

[Mfumu et al., 2018] Mfumu, J.-C. T., Mercier, A., Occello, M., and Verdier, C. (2018). A multiagent-based model for epidemic disease monitoring in dr congo. In *International Joint Conference on Biomedical Engineering Systems and Technologies*, pages 326–347. Springer. 2

[Mo and Lin, 2018] Mo, L. and Lin, P. (2018). Distributed consensus of second-order multiagent systems with nonconvex input constraints. *International Journal of Robust and Nonlinear Control*, 28(11):3657–3664. 9

[Mohamed et al., 2018] Mohamed, A., Martin, F., and Axel, H. (2018). Nonlinear model predictive control for trajectory tracking and collision avoidance of underactuated vessels with disturbances. *Ocean Engineering*, 160(12):168–180. 87

[Molinari et al., 2017] Molinari, F., Anh, N. N., and Del Re, L. (2017). Efficient mixed integer programming for autonomous overtaking. In *2017 American Control Conference (ACC)*, pages 2303–2308. IEEE. 6

[Moon and Chung, 2014] Moon, C.-b. and Chung, W. (2014). Kinodynamic planner dual-tree rrt (dt-rrt) for two-wheeled mobile robots using the rapidly exploring random tree. *IEEE Transactions on industrial electronics*, 62(2):1080–1090. 79

[Moon and Chung, 2015] Moon, C.-b. and Chung, W. (2015). Kinodynamic planner dual-tree rrt (dt-rrt) for two-wheeled mobile robots using the rapidly exploring random tree. *IEEE Transactions on industrial electronics*, 62(2):1080–1090. 5

[Morstyn et al., 2015] Morstyn, T., Hredzak, B., and Agelidis, V. G. (2015). Distributed cooperative control of microgrid storage. *IEEE transactions on power systems*, 30(5):2780–2789. 3

[Morstyn et al., 2018] Morstyn, T., Hredzak, B., and Agelidis, V. G. (2018). Control strategies for microgrids with distributed energy storage systems: An overview. *IEEE Transactions on Smart Grid*, 9(4):3652–3666. 8

[Mortezaei et al., 2018] Mortezaei, A., Simões, M. G., Savaghebi, M., Guerrero, J. M., and Al-Durra, A. (2018). Cooperative control of multi-master–slave islanded microgrid with power quality enhancement based on conservative power theory. *IEEE Transactions on Smart Grid*, 9(4):2964–2975. 3

[Motzkin, 1953] Motzkin, T. S. (1953). The double description method, in contributions to the theory of games ii. *Annals of Mathematics Study*, 28. 15

[Müller et al., 2012] Müller, M. A., Reble, M., and Allgöwer, F. (2012). Cooperative control of dynamically decoupled systems via distributed model predictive control. *International Journal of Robust and Nonlinear Control*, 22(12):1376–1397. 89

[Murray et al., 2018] Murray, A., Engelmann, A., Hagenmeyer, V., and Faulwasser, T. (2018). Hierarchical distributed mixed-integer optimization for reactive power dispatch. *IFAC-PapersOnLine*, 51(28):368–373. 9

[Nadkarni, 2016] Nadkarni, P. (2016). *Clinical Research Computing: A Practitioner's Handbook*. Academic Press. 47

[Naeem et al., 2016] Naeem, W., Henrique, S. C., and Hu, L. (2016). A reactive colregs-compliant navigation strategy for autonomous maritime navigation. *IFAC-PapersOnLine*, 49(23):207–213. 77

[Nägeli et al., 2017] Nägeli, T., Alonso-Mora, J., Domahidi, A., Rus, D., and Hilliges, O. (2017). Real-time motion planning for aerial videography with dynamic obstacle avoidance and viewpoint optimization. *IEEE Robotics and Automation Letters*, 2(3):1696–1703. 5

[Nayl et al., 2015] Nayl, T., Nikolakopoulos, G., and Gustafsson, T. (2015). Effect of kinematic parameters on mpc based on-line motion planning for an articulated vehicle. *Robotics and Autonomous Systems*, 70:16–24. 5

[Nazzal, 2017] Nazzal, A. H. (2017). Improved artificial potential field based ant colony optimization for path planning. *International Journal of Computer Science and Mobile Computing*, 6(7):114–120. 46

[Negenborn et al., 2010] Negenborn, R. R., Hellendoorn, H., and Lukszo, Z. (2010). *Intelligent infrastructures*, volume 2. Springer. 8

[Nguyen, 2016] Nguyen, M. T. (2016). *Commande prédictive sous contraintes de sécurité pour des systèmes dynamiques Multi-Agents*. PhD thesis, Université Paris-Saclay. 3

[Nguyen and Maniu, 2016] Nguyen, M. T. and Maniu, C. S. (2016). Voronoi based decentralized coverage problem: From optimal control to model predictive control. In *2016 24th Mediterranean Conference on Control and Automation (MED)*, pages 1307–1312. IEEE. 5

[Nguyen et al., 2019] Nguyen, N. T., Prodan, I., and Lefèvre, L. (2019). On the use of a computed-torque control law for the terminal region of an nmpc scheme. In *2019 19th American Control Conference (ACC)*, pages 1008–1013. IEEE. 103

[Nguyen and Sreenath, 2016] Nguyen, Q. and Sreenath, K. (2016). Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. In *2016 American Control Conference (ACC)*, pages 322–328. IEEE. 43

[Niu et al., 2019] Niu, H., Savvaris, A., Tsourdos, A., and Ji, Z. (2019). Voronoi-visibility roadmap-based path planning algorithm for unmanned surface vehicles. *The Journal of Navigation*, pages 1–25. 45

[Oh and Sun, 2010] Oh, S.-R. and Sun, J. (2010). Path following of underactuated marine surface vessels using line-of-sight based model predictive control. *Ocean Engineering*, 37(2-3):289–295. 46, 81

[Otto et al., 2019] Otto, A., Just, W., and Radons, G. (2019). Nonlinear dynamics of delay systems: an overview. 102

[Pack et al., 2009] Pack, D. J., DeLima, P., Toussaint, G. J., and York, G. (2009). Cooperative control of uavs for localization of intermittently emitting mobile targets. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(4):959–970. 3

[Paolillo et al., 2018] Paolillo, A., Gergondet, P., Cherubini, A., Vendittelli, M., and Kheddar, A. (2018). Autonomous car driving by a humanoid robot. *Journal of Field Robotics*, 35(2):169–186. 2

[Parisio et al., 2014] Parisio, A., Rikos, E., and Glielmo, L. (2014). A model predictive control approach to microgrid operation optimization. *IEEE Transactions on Control Systems Technology*, 22(5):1813–1827. 5

[Park et al., 2001] Park, M. G., Jeon, J. H., and Lee, M. C. (2001). Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing. In *ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570)*, volume 3, pages 1530–1535. IEEE. 46

[Pavlov et al., 2009] Pavlov, A., Nordahl, H., and Breivik, M. (2009). Mpc-based optimal path following for underactuated vessels. *IFAC Proceedings Volumes*, 42(18):340–345. 81

[Pawlowicz, 2019] Pawlowicz, R. (2019). M-map: A mapping package for matlab, version 1.4k, [computer software]. https://www.eoas.ubc.ca/~rich/map.html. 50

[Pedersen et al., 2017] Pedersen, E. B., Herbertsson, H. R., Niemann, H., Poulsen, N. K., and Falk, A. K. (2017). Model predictive control of sewer networks. In *Journal of Physics: Conference Series*, volume 783, page 012026. IOP Publishing. 5

[Persson and Wahlberg, 2019] Persson, L. and Wahlberg, B. (2019). Model predictive control for autonomous ship landing in a search and rescue scenario. In *AIAA Scitech 2019 Forum*, page 1169. 99

[Pettersen and Lefeber, 2001] Pettersen, K. Y. and Lefeber, E. (2001). Way-point tracking control of ships. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, volume 1, pages 940–945. IEEE. 79

[Pham et al., 2017] Pham, T. H., Prodan, I., Genon-Catalot, D., and Lefevre, L. (2017). Power balancing in a dc microgrid elevator system through constrained optimization. *IFAC-PapersOnLine*, 50(1):19–24. 5

[Prajna et al., 2007] Prajna, S., Jadbabaie, A., and Pappas, G. J. (2007). A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428. 32

[Prodan, 2012] Prodan, I. (2012). *Commande sous contraintes de systemes dynamiques multi-agents*. PhD thesis, Supélec. 3, 35

[Prodan et al., 2011] Prodan, I., Olaru, S., Stoica, C., and Niculescu, S.-I. (2011). Predictive control for tight group formation of multi-agent systems. *IFAC Proceedings Volumes*, 44(1):138–143. 6, 77

[Prodan et al., 2013] Prodan, I., Olaru, S., Stoica, C., and Niculescu, S.-I. (2013). Predictive control for trajectory tracking and decentralized navigation of multi-agent formations. *International Journal of Applied Mathematics and Computer Science*, 23(1):91–102. 5, 9, 16

[Prodan et al., 2015] Prodan, I., Stoican, F., Olaru, S., and Niculescu, S.-I. (2015). *Mixed-integer representations in control design: Mathematical foundations and applications*. Springer. 6, 16

[Prodan et al., 2014] Prodan, I., Stoican, F., Olaru, S., Stoica, C., and Niculescu, S.-I. (2014). Mixed-integer programming techniques in distributed mpc problems. In *Distributed Model Predictive Control Made Easy*, pages 275–291. Springer. 9

[Qin and Badgwell, 2003] Qin, S. J. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764. 5

[Qu et al., 2017] Qu, Y., Xu, H., Yu, W., Feng, H., and Han, X. (2017). Inverse optimal control for speed-varying path following of marine vessels with actuator dynamics. *Journal of marine science and application*, 16(2):225–236. 82

[Radmanesh and Kumar, 2016] Radmanesh, M. and Kumar, M. (2016). Flight formation of uavs in presence of moving obstacles using fast-dynamic mixed integer linear programming. *Aerospace Science and Technology*, 50:149–160. 6

[Rafael, 2004] Rafael (2004). Protector Unmanned Surface Vehicle. https://www.rafael.co.il/worlds/naval/. [Online; accessed March-2017]. 2

[Ragi and Mittelmann, 2017] Ragi, S. and Mittelmann, H. D. (2017). Mixed-integer nonlinear programming formulation of a uav path optimization problem. In *2017 American Control Conference (ACC)*, pages 406–411. IEEE. 46

[Rasekhipour et al., 2017] Rasekhipour, Y., Khajepour, A., Chen, S.-K., and Litkouhi, B. (2017). A potential field-based model predictive path-planning controller for autonomous road vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1255–1267. 5, 7

[Rawlings and Mayne, 2009] Rawlings, J. B. and Mayne, D. Q. (2009). *Model predictive control: Theory and design*. Nob Hill Pub. Madison, Wisconsin. 5

[Ren et al., 2018] Ren, Y., Zheng, L., Yang, W., and Li, Y. (2018). Potential field–based hierarchical adaptive cruise control for semi-autonomous electric vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, page 0954407018797571. 9

[Richards and How, 2005] Richards, A. and How, J. (2005). Mixed-integer programming for control. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 2676–2683. IEEE. 6

[Roberge et al., 2013] Roberge, V., Tarbouchi, M., and Labonté, G. (2013). Comparison of parallel genetic algorithm and particle swarm optimization for real-time uav path planning. *IEEE Transactions on Industrial Informatics*, 9(1):132–141. 4

[Robusto, 1957] Robusto, C. C. (1957). The cosine-haversine formula. *The American Mathematical Monthly*, 64(1):38–40. 51

[Santos and Egerstedt, 2018] Santos, M. and Egerstedt, M. (2018). Coverage control for multi-robot teams with heterogeneous sensing capabilities using limited communications. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5313–5319. IEEE. 9

[Saunders et al., 2005] Saunders, J., Call, B., Curtis, A., Beard, R., and McLain, T. (2005). Static and dynamic obstacle avoidance in miniature air vehicles. In *Infotech@ Aerospace*, page 6950. 5

[Scattolini, 2009] Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control–a review. *Journal of process control*, 19(5):723–731. 7

[Scokaert and Mayne, 1998] Scokaert, P. O. and Mayne, D. (1998). Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic control*, 43(8):1136–1142. 5

[Sfeir et al., 2011] Sfeir, J., Saad, M., and Saliah-Hassane, H. (2011). An improved artificial potential field approach to real-time mobile robot path planning in an unknown environment. In *2011 IEEE international symposium on robotic and sensors environments (ROSE)*, pages 208–213. IEEE. 46

[Shibata et al., 2018] Shibata, K., Shibata, N., Nonaka, K., and Sekiguchi, K. (2018). Model predictive obstacle avoidance control for vehicles with automatic velocity suppression using artificial potential field. *IFAC-PapersOnLine*, 51(20):313–318. 7, 16

[Skjetne et al., 2004] Skjetne, R., Smogeli, Ø. N., and Fossen, T. I. (2004). A nonlinear ship manoeuvering model: Identification and adaptive control with experiments for a model ship. 48

[Soloperto et al., 2019] Soloperto, R., Köhler, J., Müller, M. A., and Allgöwer, F. (2019). Collision avoidance for uncertain nonlinear systems with moving obstacles using robust model predictive control. In *2019 European Control Conference (ECC)*, pages 811–817. IEEE. 77

[Song and Fan, 2018] Song, C. and Fan, Y. (2018). Coverage control for mobile sensor networks with limited communication ranges on a circle. *Automatica*, 92:155–161. 3

[Stastny et al., 2015] Stastny, T. J., Garcia, G. A., and Keshmiri, S. S. (2015). Collision and obstacle avoidance in unmanned aerial systems using morphing potential field navigation and nonlinear model predictive control. *Journal of Dynamic Systems, Measurement, and Control*, 137(1):014503. 5

[Stella et al., 2017] Stella, L., Themelis, A., Sopasakis, P., and Patrinos, P. (2017). A simple and efficient algorithm for nonlinear model predictive control. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 1939–1944. IEEE. 9

[Stoican et al., 2011] Stoican, F., Prodan, I., and Olaru, S. (2011). On the hyperplanes arrangements in mixed-integer techniques. In *Proceedings of the 2011 American Control Conference*, pages 1898–1903. IEEE. 16

[Stork et al., 2006] Stork, E., Duda, R., Hart, P., and Stork, D. (2006). Pattern classification. *New York [ua]: Academic Internet Publishers*. 53

[Sugihara and Smith, 1997] Sugihara, K. and Smith, J. (1997). Genetic algorithms for adaptive motion planning of an autonomous mobile robot. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*, pages 138–143. IEEE. 4

[Sun et al., 2018a] Sun, X., Wang, G., Fan, Y., Mu, D., and Qiu, B. (2018a). Collision avoidance using finite control set model predictive control for unmanned surface vehicle. *Applied Sciences*, 8(6):926. 46, 99

[Sun et al., 2018b] Sun, Z., Dai, L., Liu, K., Xia, Y., and Johansson, K. H. (2018b). Robust mpc for tracking constrained unicycle robots with additive disturbances. *Automatica*, 90:172–184. 103

[Szlapczynski, 2011] Szlapczynski, R. (2011). Evolutionary sets of safe ship trajectories: a new approach to collision avoidance. *The Journal of Navigation*, 64(1):169–181. 77

[Taha, 2011] Taha, H. A. (2011). *Operations research: an introduction*, volume 790. Pearson/Prentice Hall. 27

[Tan et al., 2005] Tan, C. S., Sutton, R., and Chudley, J. (2005). Quasi-random, manoeuvre-based motion planning algorithm for autonomous underwater vehicles. *IFAC Proceedings Volumes*, 38(1):103–108. 5

[Teatro et al., 2014] Teatro, T. A., Eklund, J. M., and Milman, R. (2014). Nonlinear model predictive control for omnidirectional robot motion planning and tracking with avoidance of moving obstacles. *Canadian Journal of Electrical and Computer Engineering*, 37(3):151–156. 7

[Tee et al., 2009] Tee, K. P., Ge, S. S., and Tay, E. H. (2009). Barrier lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 45(4):918–927. 32

[Tran et al., 2018] Tran, N., Prodan, I., Grøtli, E., and Lefèvre, L. (2018). Potential-field constructions in an mpc framework: application for safe navigation in a variable coastal environment. *IFAC-PapersOnLine*, 51(20):307–312. 10

[Tran et al., 2019] Tran, N., Prodan, I., Grøtli, E., and Lefèvre, L. (2019). Distributed nonlinear optimization-based control for multi-agent systems navigation in a coastal environment. In *2019 European Control Conference (ECC)*. IEEE. 10

[Tran et al., 2017] Tran, N. Q. H., Prodan, I., and Lefèvre, L. (2017). Nonlinear optimization for multi-agent motion planning in a multi-obstacle environment. In *2017 21st International Conference on System Theory, Control and Computing (ICSTCC)*, pages 488–493. IEEE. 10

[Tuncer and Yildirim, 2012] Tuncer, A. and Yildirim, M. (2012). Dynamic path planning of mobile robots with improved genetic algorithm. *Computers & Electrical Engineering*, 38(6):1564–1572. 4

[Tzes et al., 2018] Tzes, M., Papatheodorou, S., and Tzes, A. (2018). Visual area coverage by heterogeneous aerial agents under imprecise localization. *IEEE control systems letters*, 2(4):623–628. 5

[Venkat et al., 2005] Venkat, A. N., Rawlings, J. B., and Wright, S. J. (2005). Stability and optimality of distributed model predictive control. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 6680–6685. IEEE. 9

[Villanueva et al., 2017] Villanueva, M. E., Quirynen, R., Diehl, M., Chachuat, B., and Houska, B. (2017). Robust mpc via min–max differential inequalities. *Automatica*, 77:311–321. 103

[Wächter and Biegler, 2006] Wächter, A. and Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57. 9, 31

[Wang et al., 2019a] Wang, H., Huang, Y., Khajepour, A., Rasekhipour, Y., Zhang, Y., and Cao, D. (2019a). Crash mitigation in motion planning for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*. 7, 16

[Wang et al., 2017a] Wang, L., Ames, A. D., and Egerstedt, M. (2017a). Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674. 15, 31

[Wang et al., 2018] Wang, N., Sun, Z., Yin, J., Su, S.-F., and Sharma, S. (2018). Finite-time observer based guidance and control of underactuated surface vehicles with unknown sideslip angles and disturbances. *IEEE Access*, 6:14059–14070. 78

[Wang et al., 2012] Wang, Q., Fang, H., Chen, J., Mao, Y., and Dou, L. (2012). Flocking with obstacle avoidance and connectivity maintenance in multi-agent systems. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 4009–4014. IEEE. 77

[Wang et al., 2017b] Wang, T., Yan, X., Wang, Y., and Wu, Q. (2017b). A distributed model predictive control using virtual field force for multi-ship collision avoidance under colregs. In *2017 4th International Conference on Transportation Information and Safety (ICTIS)*, pages 296–305. IEEE. 9

[Wang et al., 2015] Wang, X., White, L., Chen, X., Chen, K., Li, X., and Wang, H. (2015). On the model design of integrated intelligent big data analytics systems. *Industrial Management & Data Systems*. 2

[Wang et al., 2019b] Wang, Y., He, L., and Huang, C. (2019b). Adaptive time-varying formation tracking control of unmanned aerial vehicles with quantized input. *ISA transactions*, 85:76–83. 3

[Wang et al., 2019c] Wang, Y., Tong, H., and Wang, C. (2019c). High-gain observer-based line-of-sight guidance for adaptive neural path following control of underactuated marine surface vessels. *IEEE Access*, 7:26088–26101. 99

[Webb and Van Den Berg, 2013] Webb, D. J. and Van Den Berg, J. (2013). Kinodynamic rrt*: Asymptotically optimal motion planning for robots with linear dynamics. In *2013 IEEE International Conference on Robotics and Automation*, pages 5054–5061. IEEE. 5

[Wieland and Allgöwer, 2007] Wieland, P. and Allgöwer, F. (2007). Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467. 31

[Woods and La, 2017] Woods, A. C. and La, H. M. (2017). A novel potential field controller for use on aerial robots. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(4):665–676. 5

[Wooldridge, 2009] Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons. 2

[Wu et al., 2013] Wu, B., Wen, Y., Huang, Y., and Zhu, M. (2013). Research of unmanned surface vessel (usv) path-planning algorithm based on arcgis. *Proceedings of ICTIS 2013*, pages 2125–2134. 45

[Wu et al., 2019] Wu, M., Gao, F., Yu, P., She, J., and Cao, W. (2019). Improve disturbance-rejection performance for an equivalent-input-disturbance-based control system by incorporating a proportional-integral observer. *IEEE Transactions on Industrial Electronics*. 77

[Wu et al., 2018] Wu, Z.-G., Xu, Y., Pan, Y.-J., Su, H., and Tang, Y. (2018). Event-triggered control for consensus problem in multi-agent systems with quantized relative state measurements and external disturbance. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(7):2232–2242. 8

[Xiong et al., 2019] Xiong, C., Chen, D., Lu, D., Zeng, Z., and Lian, L. (2019). Path planning of multiple autonomous marine vehicles for adaptive sampling using voronoi-based ant colony optimization. *Robotics and Autonomous Systems*, 115:90–103. 45

[Xue et al., 2013] Xue, B., Zhang, M., and Browne, W. N. (2013). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE transactions on cybernetics*, 43(6):1656–1671. 4

[Yadbantung and Bumroongsri, 2019] Yadbantung, R. and Bumroongsri, P. (2019). Tube-based robust output feedback mpc for constrained ltv systems with applications in chemical processes. *European Journal of Control*, 47:11–19. 103

[Yan and Wang, 2012] Yan, Z. and Wang, J. (2012). Model predictive control for tracking of underactuated vessels based on recurrent neural networks. *IEEE Journal of Oceanic Engineering*, 37(4):717–726. 99

[Yang et al., 2018] Yang, C., Duan, M., Lin, P., Ren, W., and Gui, W. (2018). Distributed containment control of continuous-time multiagent systems with nonconvex control input constraints. *IEEE Transactions on Industrial Electronics*, 66(10):7927–7934. 9

[Yang and Zheng, 2014] Yang, J. and Zheng, W. X. (2014). Offset-free nonlinear mpc for mismatched disturbance attenuation with application to a static var compensator. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 61(1):49–53. 77

[Yang et al., 2015] Yang, J.-M., Tseng, C.-M., and Tseng, P. (2015). Path planning on satellite images for unmanned surface vehicles. *International Journal of Naval Architecture and Ocean Engineering*, 7(1):87–99. 76

[Yang and Sukkarieh, 2012] Yang, K. and Sukkarieh, S. (2012). Model predictive unified planning and control of rotary-wing unmanned aerial vehicle. In *2012 12th International Conference on Control, Automation and Systems*, pages 1974–1979. IEEE. 16

[Yang et al., 2016] Yang, X., Yang, W., Zhang, H., Chang, H., Chen, C.-Y., and Zhang, S. (2016). A new method for robot path planning based artificial potential field. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pages 1294–1299. IEEE. 46

[Yang et al., 2014] Yang, Y., Du, J., Liu, H., Guo, C., and Abraham, A. (2014). A trajectory tracking robust controller of surface vessels with disturbance uncertainties. *IEEE Transactions on Control Systems Technology*, 22(4):1511–1518. 78

[Yu et al., 2018a] Yu, P., Wu, M., She, J., Liu, K.-Z., and Nakanishi, Y. (2018a). Robust tracking and disturbance rejection for linear uncertain system with unknown state delay and disturbance. *IEEE/ASME Transactions on Mechatronics*, 23(3):1445–1455. 77

[Yu et al., 2018b] Yu, S., Guo, Y., Meng, L., Qu, T., and Chen, H. (2018b). Mpc for path following problems of wheeled mobile robots. *IFAC-PapersOnLine*, 51(20):247–252. 78

[Zanelli et al., 2017] Zanelli, A., Domahidi, A., Jerez, J., and Morari, M. (2017). Forces nlp: an efficient implementation of interior-point methods for multistage nonlinear nonconvex programs. *International Journal of Control*, pages 1–17. 9

[Zavlanos and Pappas, 2007] Zavlanos, M. M. and Pappas, G. J. (2007). Potential fields for maintaining connectivity of mobile networks. *IEEE Transactions on robotics*, 23(4):812–816. 77

[Zhong and Cassandras, 2008] Zhong, M. and Cassandras, C. G. (2008). Distributed coverage control in sensor network environments with polygonal obstacles. *IFAC Proceedings Volumes*, 41(2):4162–4167. 3

[Zhu et al., 2017] Zhu, Q., Ma, J., Liu, Z., and Liu, K. (2017). Containment control of autonomous surface vehicles: A nonlinear disturbance observer–based dynamic surface control design. *Advances in Mechanical Engineering*, 9(10):1687814017727384. 85

[Zhu et al., 2006] Zhu, Q., Yan, Y., and Xing, Z. (2006). Robot path planning based on artificial potential field approach with simulated annealing. In *Sixth International Conference on Intelligent Systems Design and Applications*, volume 2, pages 622–627. IEEE. 46

Abstract

This thesis proposes optimization-based control solutions for the motion planning of multi-agent dynamical systems operating in a variable environment (with static/mobile obstacles and time-varying environmental disturbances).

Collision-free paths are planned for the agents through the combined use of set theory (particularly, bounded convex sets), non(-linear) Model Predictive Control (MPC), Potential Field (PF) and graph-based methods. The contributions build on the proposal of repulsive potential field constructions together with on-off barrier functions which describe and, respectively, activate/deactivate the collision-free conditions introduced in a distributed NMPC framework. These constructions are further used for connectivity maintenance conditions among the group of agents while ensuring the tracking of the a priori generated path. Furthermore, a nonlinear disturbance observer is integrated within the control scheme for environmental disturbance rejection.

Finally, the results are validated in simulation through comparisons with mixed-integer approaches and over a benchmark for the safe navigation of Unmanned Surface Vehicles (USVs) in the Trondheim fjord, Norway, using real numerical data.

Résumé

Cette thèse propose des solutions de commande basées sur la planification optimale de trajectoires pour des systèmes dynamiques multi-agents fonctionnant dans un environnement variable (avec obstacles statiques ou mobiles et des perturbations variables dans le temps).

Cette planification de trajectoires repose sur l'utilisation combinée de la théorie des ensembles (en particulier des ensembles convexes bornés), de la commande prédictive non-linéaire (NMPC), du calcul de champs de potentiel et des méthodes basées sur des graphes. Elle se base sur la construction de champs de potentiel répulsifs associés à des fonctions de barrière marche-arrêt (on-off barrier functions) qui décrivent et activent ou désactivent les trajectoires libres (sans collision) calculées au préalable par une commande de type NMPC distribuée. Ces constructions sont ensuite utilisées pour maintenir la connectivité dans le groupe d'agents, tout en assurant le suivi du chemin pré-généré. En outre, un observateur pour l'estimation de perturbations non linéaires est intégré dans le schéma de commande afin de les rejeter.

Les résultats théoriques obtenus sont validés en simulation, par des comparaisons avec des approches utilisant la programmation mixte en nombres entiers, à l'aide de données numériques réelles provenant d'une plateforme de navigation sécurisée pour les véhicules de surface non habités dans le fjord de Trondheim (Norvège).