MPC-BASED MOBILE ROBOTS MOTION PLANNING AND CONTROL IN UNCERTAIN DYNAMIC ENVIRONMENT

by

MORTEZA FARROKHSIAR

B.Sc., Amirkabir University of Technology (Tehran Polytechnic), 2007 M.A.Sc., The University of British Columbia, 2009

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Okanagan)

April 2015

© Morteza Farrokhsiar, 2015

Abstract

The main focus of this thesis is on the motion planning and control of mobile robots in dynamic unstructured environments in which the primary challenge is to formulate and deal with uncertainty. This thesis contributes to the motion planning problem in three distinct yet related aspects that can together present a model predictive approach to enhance autonomy of mobile robots in dynamic unknown environments.

The first contribution of this thesis is to introduce a robust yet probing control algorithm. The proposed algorithm is based on the output-feedback tube-based model predictive control (MPC). The performance of the algorithm has been enhanced using the partially-closed loop strategy. The tube-based approach requires uncertainties to be modeled in the set-theoretic framework, whereas the partially closed-loop strategy is modeled in the probabilistic framework. A key component of the algorithm is related to proposing the relationship between these two different paradigms. The proposed framework utilizes the uncertainty fusion in the probabilistic framework and collision avoidance in the set-theoretic framework. The efficiency of the proposed algorithm is verified using thorough numerical simulations and experiments.

The second contribution of this thesis is in regards with linearization of stochastic nonlinear systems. A statistical linearization method, unscented transform, is proposed to replace the analytical linearization method in MPC. The advantage and disadvantage of such replacement has been examined through extensive numerical simulations. The numerical simulation indicates that statistical linearization has two important advantages. First, the proposed approach is derivative free that is it can be applied to complex systems for which no analytical model exists. Second, it is more accurate so that it enhances performance of the planning algorithm. However, the tradeoff is that the analytical linearization is computationally less expensive.

The third contribution of this thesis is related to the formulation of the robust tube-based MPC scheme for incremental smoothing and mapping known as active iSAM problem in the literature. In addition to utilizing a robust MPC scheme, the active iSAM utilizes the optimization-based method, iSAM, to solve the simultaneous localization and mapping SLAM problem. Extensive numerical simulations have been conducted to verify the performance of the algorithm.

Preface

This thesis is produced based on my research work, and in collaboration with other researchers in Advanced Control and Intelligent Systems (ACIS) Laboratory of the University of British Columbia. In particular, Chapter 2, is based on the research carried out in collaboration with Graham Pavlik and under the supervision of Dr. Homayoun Najjaran [1]. I am thankful to Graham Pavlik for his help while building the testbed and conducting the experiment.

The idea of unscented transform based model predictive control, presented in Chapter 3, evolved over the course of several years based on discussions with Dr. Najjaran, as well as anonymous feedback received from peer reviews. The initial results were subsequently published [2-4]. Chapter 3 of this thesis is based on the final results, which are additionally published [5].

Also, I would like to thank Monica Cella for the editorial review of this thesis.

Table of Contents

Abstractii
Prefaceiv
Table of Contentsv
Table of Tablesx
Table of Figures xi
Acknowledgmentsxvii
Dedicationxix
Chapter 1. Introduction1
1.1 Motivation1
1.2 Objectives
1.3 Literature review
1.3.1 Model Predictive Control3
1.3.2 Stochastic receding horizon control4
1.3.3 Tube based MPC

1.3.4	SLAM and Active SLAM6
1.4	Contributions
1.5	Thesis organization11
Chap	ter 2. Robust yet probing motion planning scheme13
2.1	Overview
2.2	Problem statement
2.2.1	Deterministic Motion Planning16
2.3	State Estimation and Prediction19
2.4	Tube-based MPC 21
2.4.1	Relation between Set-theoretic and Stochastic Frameworks: An Illustrative
	Example
2.4.2	Developing Heuristic Relation between Set-theoretic and Stochastic Frameworks
	for Unicycle28
2.4.3	Nominal MPC 30
2.4.4	Ancillary controller
2.5	Simulation Results

2.5.1	Effect of Covariance Trace Optimization on Planning and Estimation Quality 42
2.6	Experimental Results
2.6.1	Experimental Testbed 44
2.6.2	Experimental Results
2.7	Summary 47
Chap	ter 3. Unscented Model Predictive Control of Chance Constrained Nonlinear
	Systems
3.1	Overview
3.2	Problem Statement
3.2.1	The nonholonomic system model53
3.2.2	The motion control formulation55
3.2.3	Constraint tightening
3.3	Unscented Model Predictive Motion control60
3.3.1	Unscented transformed motion equations60
3.3.2	Unscented model predictive motion control using output feedback
3.4	The Overall System Architecture64

3.5	Simulation Results and Discussions6	5
3.6	Summary and Future Work7	6
Chap	ter 4. Robust Active iSAM7	8
4.1	Overview7	8
4.2	Problem Formulation7	9
4.2.1	iSAM 2 algorithm7	9
4.2.2	Active SLAM problem	1
4.2.3	Proposed solution overview8	4
4.2.4	Trajectory Optimization8	6
4.3	Robust motion planning for active slam8	7
4.3.1	Active iSAM8	7
4.4	Ancillary controller9	1
4.5	The overall motion planning algorithm9	2
4.6	Simulation results9	2
4.6.1	Nominal MPC9	3
4.6.2	The effect of the ancillary controller9	4

4.7	Summary	99
Chap	ter 5. Conclusions and Future Work	101
5.1	Conclusions	101
5.2	Future work	104
Refer	ences	106

Table of Tables

Table 3-1 Approximation of the mean and covariance matrix norm.							
Table 3-2 The relative values of fuel index and CPU time (UMPC`s value over CMPC`s							
value).	74						

Table of Figures

Figure 2-1 Schematic of the two-wheeled mobile robot17
Figure 2-2 The state uncertainty evaluation using the Monte Carlo simulation (color
dots) and the set-theoretic framework (black bars) versus time
Figure 2-3 The time evolution of the confidence region for $\beta=99.73\%$ (blue bars)
and the set-theoretic formulation (red bars).
Figure 2-4 Robot motion with constant rotational and translational velocity with only
rotational disturbance (a), and with only translational disturbance (b).
The nominal path is shown as a blue line, and the stochastic and set-
theoretic uncertainty discretions are shown with blue and red points,
respectively27
Figure 2-5 The schematic of the confidence region in the plane, the exact confidence
region (ellipse), and the circumscribed circle and square over
approximating it27
Figure 2-6 The exact uncertainty set versus time related to $\mathrm{A1}$ and $\mathrm{A2}$ shown in black
rectangles and red rhombic shape, respectively. The stochastic
approximation of each system is shown in green and blue ellipses,
respectively

Figure 2-13	Integration of Matlab based motion planning and C++ image acquisition	
	and processing	. 46

Figure 2-15 The experimental scenario: the robot path (black line), the planned paths
(blue dashed line), the target (red circle) and the robot final pose
uncertainty indicator (green circle)49
Figure 2-16 The nominal, ancillary and overall rotational control signals
Figure 2-17 The nominal, ancillary and overall translational control signals
Figure 3-1 Schematic of the two-wheeled mobile robot54
Figure 3-2 The obstacle location mean (the blue dot in the middle), the collision
region without considering uncertainties (the inner red circle), and the
collision region considering uncertainties (the outer green circle)
Figure 3-3 The function block diagram of the predictive controller (red), mobile robot
(blue), and the state estimator (green)65
Figure 3-4 Robot path in the first case; the robot target is the origin. The path
produced by UMPC and CMPC is shown in blue and red, respectively. The
estimated and true paths are shown with the dotted and solid lines,
respectively69
Figure 3-5 Simulation II - The robot path is calculated by the UMPC and CMPC from
the initial location (-20,0) to the destination (0,0) in presence of a
stationary obstacle denoted by a green partial ellipse tentatively

proportional to the covariance of the obstacle location. The path produced by UMPC and CMPC is shown in blue and red, respectively. 69

Figure 4-1 Schematic of the two-wheeled mobile robot......82

- Figure 4-4 Planning with Euclidean cost function, the robot and the iSAM estimated paths are shown in blue and green, respectively. The true and estimated landmark locations are shown using black and red dots, respectively........95

Figure	4-11	The	Monte	Carlo	simulation	of	the	active	iSAM	controller	with	the	
	a	ncill	ary cont	roller.		•••••			•••••				. 99

Acknowledgments

This thesis would not have been possible without the support and help of many people. I would like to express my gratitude to my PhD supervisor Dr. Homayoun Najjaran. It is hard to imagine if this thesis would have been successfully completed without his support, inspiration, persuasion, and ideation. I consider him my role-model in both my life and career.

I would like to thank my supervisory and doctoral committee: Drs. Yang Cao, Steven Waslander, Jahangir Hossain, Rudolf Seethaler, and Edward Park. Their input, valuable discussions and critiques played an important role in shaping this thesis.

Additionally I would like to thank my colleagues and co-authors at the UBC ACIS Lab and the UBC School of Engineering staff and faculties for their support during my PhD studies.

During my graduate studies at UBC, I met individuals who had a profound influence on my life. I lived with them, enjoyed their company and support, and gained valuable experience from them. In particular, I am blessed with Ali Ahmadi's company whose friendship and kindness have supported me greatly during these years. I also wish to thank my great friends Milad Abolhasani, Parnia Vafaei-kia, Mohsen Javdani, Sina Jomeh, Kamyar Keramatian, and Seyed Mahdi Modirzadeh, for their support, inspiration, and company. In particular, I should thank Mahdi for the music training that gave me insight to the fact that I made the right choice in pursuing engineering rather than music. I would also like to thank Mojtaba Jalalpour and Reza Nouri (R.I.P.) for their kindness, inspiration, and brotherhood during my postsecondary education.

My PhD studies, and other achievements, would not have been possible and without the unconditional love and support of my family. I would like to express my deepest gratitude to my father and mother, Akhtar Salarian and Ali Asghar Farrokhsiar, brothers, Mani and Mazdak, and my sister, Vida, for their kindness, support, and encouragement. I am eternally thankful to my beloved and beautiful wife who provides unending support and constant inspiration towards my involvement in PhD research and all that lies beyond.

To my beloved Mom and Dad:

Akhtar Salarían and Alí Asghar Farrokhsíar

Chapter 1. Introduction

1.1 Motivation

The main motivation of this thesis is to enhance the autonomy of mobile robots in dynamic unknown environments. During the past two decades, there has been a continuously increasing interest in developing autonomous robots that are able to accomplish complicated tasks in structured manufacturing environments; autonomous robotic systems are becoming favorable in various applications [6]. Some illustrative examples of this trend includes the United States (US) army's investment in the Defense Advanced Research Projects Agency (DARPA) robotic grand challenge, as well as initiatives such as California Partners for Advanced Transportation Technology (PATH) and autonomous flight coordination [7, 8].

Among various aspects of the mobile robot autonomy, motion planning and control in the uncertain environment is selected as the main focus of this research. A solution to this problem is a precursor to realization of fully autonomous unmanned vehicles for real-world applications [9, 10]. In robotics literature, the problem of motion planning involves finding the control actions that drive a robot from a given initial state to a desired target [11]. The objective of this research is focused on dealing with challenges associated with this problem, and is discussed in the proceeding section.

1.2 Objectives

The main objective of this research is to contribute to develop a practical motion planning and control scheme for a nonholonomic mobile robot in unknown and dynamic environments. Uncertainties in sensing and control signals make the motion planning a challenging task. Planning algorithms are subject to different constraints including nonholonomic motion, collision avoidance, environment occlusion and dynamicity, and actuator saturation. In addition, real-time motion planning requires computationally efficient algorithms. In this research, motion control refers to the combination of motion planning and kinematic control of a robot [11], which were traditionally solved as separate problems [12]. Recently, it has been shown that for nonholonomic systems, integrating the planning and kinematic control can increase the overall system performance [13].

This thesis contributes to the motion planning literature by considering three related, yet distinct, problems. The first problem is to investigate whether or not a robust, as well as probing, motion planning scheme is feasible. In this context, probing refers to the ability of a robot to actively localize itself. The second, and practically important question in robotics, addresses which linearization method is the best algorithm for stochastic systems. The last question discussed in this thesis is whether the probing robust algorithms developed earlier can be modified to be applied to the problem of active SLAM (simultaneous localization and mapping). Before elaborating on these questions, the related literature review material is presented in the following section.

1.3 Literature review

As mentioned in Section 1.2, the main focus of this research is on motion planning and control of robots in uncertain environments. Such a control system is subject to various constraints, including motion constraints, control signal saturations, and collision avoidance. Additionally, a practical algorithm should be able to handle imperfect information as well. As a wellestablished approach, model predictive control (MPC) that is able to handle system constraints systematically [14] has been chosen as the proper control scheme. To provide a better background, a brief review of the model predictive control as well as the SLAM problem is presented.

1.3.1 Model Predictive Control

MPC refers to a set of optimal control schemes that can effectively control constrained systems [15-17]. The MPC scheme solves the control problem (i.e. computes the control action) by developing a control plan over a finite horizon based on the current states, prediction of future states, and desired output of a system. The controller executes a part of the plan and updates the system states as new measurements become available. Having updated the states, the controller sequentially refreshes the plan until the control objective is met. In essence, the plan horizon gradually recedes in this recursive process, which is why receding horizon control (RHC) is another common name for MPC. An extensive review on this topic can be found in previous literature [18, 19]. The MPC scheme can take into account explicit state and control action constraints, which can be formulated as chance constrains in stochastic systems. This property of MPC has made it a powerful tool for tackling robot

motion planning and obstacle avoidance problems which are typically subject to different constraints on the system states and control actions [3, 20, 21].

1.3.2 Stochastic receding horizon control

Most often, motion planning problems are subject to different constraints on the system states and control laws. Obstacles and control action limitations are well-known examples of such constraints. The RHC approach has been widely used to address problems involving robot localization uncertainty [22, 23]. In unknown environments with probabilistic uncertainties, the RHC collision avoidance constrains must be reformulated as chance constrains, which ensure that the probability of collision with surrounding obstacles is below a given threshold [24, 25]. Blackmore has proposed the RHC-based probabilistic particle control to avoid the assumptions related to system linearity and Gaussian noise [26]. The collision probability threshold can be assumed to be constant in all iterations or, as discussed in previous literature, a risk allocation scheme can be used to assign a variable collision probability [27]. An important issue in the RHC motion planning is incorporating the future information into the planning algorithm. Ignoring the future information may lead to unbounded uncertainties. The unbounded growth of the uncertainty will then require an exceedingly conservative planning solution to avoid obstacles [28]. In a similar work, the closed-loop covariance of the linear Gaussian system has been discussed [29].

In all of the stated stochastic RHC motion planning methods, the motion model has been assumed to be linear. However, in practice, many vehicles including unmanned aerial vehicles (UAVs), mobile robots, and automobiles are usually modeled as nonlinear and nonholonomic systems. During the past decade, while there has been an increasing interest in deterministic MPC of nonholonomic systems [30-32] and related stability issues [33-35], the MPC design for nonholonomic systems with stochastic constraints should be further studied. This topic will be reviewed in more detail in Chapter 3.

1.3.3 Tube based MPC

The use of MPC in uncertain environments requires handling of important issues, as a motion planning scheme may have several problems [36]. Among these issues, one of the most important aspects to consider is that of the robustness of the open-loop implementation of MPC, which can easily be compromised by process disturbance and noise. Different approaches such as min-max MPC [37], feedback MPC [38], and constraint tightening [24] have been proposed to improve robustness. Typically, these solutions are either computationally expensive or sometimes inefficient [39, 40]. A computationally viable solution with proven robustness is tube-based MPC that is a dual control scheme consisting of an open-loop MPC with constraint tightening to plan a nominal trajectory and an inner feedback loop to allow for increased robustness in the system trajectory [36]. Intuitively, tubes are the sets that contain the state space trajectories at each time instance. Tube-based MPC has been proven to be a robust and computationally efficient control method for linear systems under different types of uncertainty, including nondeterministic state feedback systems [40], nondeterministic output feedback systems [41], and stochastic systems with chance constraints [42]. Also, the tube-based method has been successfully applied to the state feedback nonlinear systems [36, 40, 43]. Rawlings and Mayne [36] have suggested an

implementation of the tube-based MPC for the output feedback control of nonlinear systems. In this research, I have adopted and modified their approach for the robust motion planning and control of mobile robots with imperfect state information. In this method, the MPC has two parts: *nominal MPC* and *ancillary MPC*. The nominal MPC is an MPC with constraint tightening that plans a nominal trajectory to the desired final target using the estimated states. On the other hand, the ancillary MPC guarantees that the estimated states follow the trajectory planned by the nominal MPC. In the original implementation discussed [36], the constraint tightening is a computationally efficient scheme that relies on *a priori* heuristic information, but this method disregards the existing relationship between state estimation and control processes in nonlinear systems and hence it may result in either overly conservative or unsafe performance of the system. A modification to the tube-based method is discussed in Chapter 2.

1.3.4 SLAM and Active SLAM

The last few decades have witnessed increasing demand for the autonomous robots in both military and civil applications. As the robotic rescue and covering missions become more complicated, one aspect of the SLAM problem becomes more important: active SLAM. in which robot autonomously explore the environment while tries to minimize the SLAM uncertainty. While SLAM has been under investigation for multiple decades, active SLAM is a relatively new issue which is still considered an open problem.

SLAM is the ability of the robot to incorporate available information to map the environment and locate itself inside the map in the absence of GPS-like devices. Since the seminal paper of Smith et al [44], during the past decade various frameworks, including Extended Kalman Filtering (EKF) [45], Particle Filtering [46], topologic solutions [47], Expectation Maximization [48], and soft computing [49], have been used to solve the SLAM framework. For the past several decades, filtering methods have been the dominant framework of the SLAM problem [50]; until just a few years ago, there has been major shift from the filtering methods to stateof-the-art optimization methods [51]. The recent optimization-based methods utilize the specific structure of SLAM problem that allows for incremental algorithms [52, 53] that can provide more accurate and yet real-time solutions. This topic will be revisited in Section 2 where the iSAM 2.0 method is discussed in more detail.

The active SLAM, a more recent problem than SLAM, refers to the problem of optimizing the robot trajectory to improve the SLAM algorithm performance. Though its general principle can be traced back to dual control theory [54], the first experience in the SLAM framework is the work by Davison and Murray, in which they used active vision to increase the accuracy of the localization of the robot [55]. Later, Sim and Roy (2005) utilized an information-theoretic approach by introducing the *information surface*, to analyze the relation between robot trajectory and the SLAM algorithm performance [56]. In a closely related work, Haung et al. discussed the MPC look-ahead SLAM algorithm in which an MPC scheme is used to minimize the trace of predicted covariance of the EKF [57]. Fang et al have proposed a d-optimal MPC scheme for the minimum-time bearing-only SLAM problem [58]. In an inspiring work, Leung et al. proposed to use an artificial short-term goal, named attractor point, to the MPC-based active SLAM; the attractor point is chosen by a higher-level decision making algorithm, such as the exploration algorithm [59, 60]. In order to use the MPC-based active SLAM scheme, a

complicated dynamic programming problem must be solved. Ny and Pappas have utilized a value iteration method to solve a similar dynamic programming problem [61]. In addition to EKF-based methods, particle filtering has been extensively used in various active SLAM schemes [62]. Many researchers have utilized reinforced learning algorithms to solve the active SLAM problem [63-65]. Among them, Kollar and Roy's method in previous work has suggested that the active SLAM problem can be formulated as a Partial Observable Markov Decision Making Process (POMDP). However, solving the resulted POMDP is not practical. Therefore, a two-stage algorithm is proposed:

Stage 1 (*Exploration*): In this stage, a set of constrained points are chosen to maximize the likelihood of completing the map coverage based on *a priori* knowledge. These points can be chosen using the art-gallery algorithms [66, 67].

Stage 2 (*active localization*): In this stage, the robot path is generated by utilizing reinforcement learning to pass through the constrained points. Similar to MPC-based solutions [57], the results indicate that the a-optimal solution outperforms the d-optimal solution. Though not related to the focus of this research, it is worth noting that recently Carrillo et al (2012) has proposed a modified d-optimal criterion that outperforms the a-optimal criterion [68]. Reviewing the existing literature of active SLAM reveals that most of the existing methods have two points in common:

- 1- They use a receding horizon scheme on the motion planning level.
- 2- They utilize filtering based methods.

The MPC scheme solved numerically is sensitive to disturbance and noise. Also, while the focus on the SLAM society has shifted to the optimization method, the active SLAM literature is still focused on the filtering based methods. In this project, the enhancement of the performance of the current active SLAM algorithms was trialed by focusing on these points. Chapter 4 is dedicated to this issue.

1.4 Contributions

In this section, we briefly review the contribution of this dissertation presented in each chapter.

Chapter 2: The main contribution discussed in this chapter is the introduction of a probing tube-based MPC algorithm in an integrated motion planning and control algorithm. The output-feedback tube-based MPC controller structure [36] has been adopted and further improved, and includes a nominal MPC and an ancillary MPC. In this work, we have enhanced the probing component of the controller by incorporating a partially closed-loop strategy for probing and state constraint tightening. We have developed our algorithm by integrating output-feedback tube-based nonlinear MPC and partially closed-loop strategy that use two different frameworks to describe uncertainty. Specifically, the uncertainty is modeled as nondeterministic (set-theoretic) and probabilistic in tube-based MPC [40] and partially closed-loop strategy [28], respectively. To take advantage of the robustness of the tube-based MPC and at the same time incorporate the future measurements effectively, we have developed a relationship between the set-theoretic and stochastic paradigms in linear systems to unify the two uncertainty frameworks (see Section 2.3.1); the application of this

relationship was further extended to nonlinear systems under certain assumptions. Then we have utilized the same relationship, as a heuristic, for the unicycle model which is smoothly nonlinear. As a result, we can claim that our proposed approach for constraint tightening has been carried out in a more systematic way than of that previously discussed in literature [36]. In addition to constraint tightening, the proposed approach allows for the inclusion of an active probing term in the controller cost function that can in turn facilitate active localization which is an important feature for real-world robotics applications.

Chapter 3: The main contribution of this chapter is to introduce a new systematic way to generalize the linear stochastic MPC methods to nonlinear systems by approximating the state transition using an unscented transform that is a statistical linearization method. Our linearization approach offers two advantages over the Taylor expansion linearization approach. First, statistical linearization outperforms analytical linearization methods in terms of prediction and estimation error [69]. Second, there is no need to calculate the Jacobian of the system and measurement models. The latter can be a critical advantage when the system model is complex, or simply unavailable. Although the unscented transform is computationally more expensive, the additional computational complexity is not a burden for robotic systems, especially for low dimensional systems [69].

Chapter 4: This chapter is dedicated to a robust MPC based active SLAM that utilizes an optimization based method to solve the SLAM problem. Among the various methods that have been discussed to increase the robustness of MPC, we are interested in the latter method: tube-based MPC; due to its practical philosophy and proper structure which suits

the robotic applications. In the nonlinear tube based MPC, the controller consists of two MPC controller: one nominal MPC which is a conventional MPC with constraint tightening, and an ancillary MPC which stabilizes the system around the nominal trajectory. Also, in the proposed active SLAM method, the nominal controller utilizes an optimization based SLAM scheme, iSAM 2 algorithm, instead of the EKF or PF which are the main focus of the existing literature on the active SLAM. The tube-based MPC can be implemented using the set algebra efficiently, whereas the iSAM 2 algorithm utilizes Gaussian distributions to model the uncertainty. Therefore, an important part of developing the new active SLAM method is to find a relationship between set-theoretic and stochastic frameworks.

1.5 Thesis organization

Chapter 2 starts with elaborating on different aspects of the motion planning problem. These include the robot motion and sensing models, state estimation processes and the motion planning formulation in uncluttered deterministic environments. The relationship between the set-theoretic and stochastic frameworks and the proposed probing tube-based MPC are discussed in Section 2.3. Section 2.4 verifies the proposed method through both simulation and experiments with a two wheel differential drive robot. Some concluding remarks have been provided in Section 2.5.

Chapter 3 is dedicated to the unscented MPC. The nonholonomic system model description, motion control formulation, and chance constraints will be discussed in Section 3.2. In Section 3.3., the formulation of the unscented predictive motion control will be discussed. Section

3.4 presents the validation of the proposed motion control algorithm through numerical simulation. Section 3.5 includes the concluding remarks and future work.

Chapter 4 is dedicated to the active iSAM algorithm. Before elaborating on the proposed MPC based motion planning and control scheme, the overall system setting, assumptions, active SLAM scheme, and incremental smoothing and mapping (ISAM 2) methods are discussed in Section 4.2. The proposed robust active SLAM method is discussed in Section 4.3. The proposed algorithm performance is examined through numerical simulations, which is discussed in Section 4.4, and are followed by the concluding remarks in Section 4.6.

Chapter 2. Robust yet probing motion planning scheme

2.1 Overview

This chapter is dedicated to present the idea of a robust and probing MPC based motion planning control scheme. The use of MPC as a motion planning scheme has several problems. One problem is that most MPC schemes have limited attractive regions [70], and this is particularly limiting in cluttered environments. One possible solution to enlarge attractive regions is to extend the horizon, but this solution will remarkably increase the computational load, which is not desirable for real-time implementations. We have used a more computationally effective approach for expanding the attractive region based on the *coarse dynamic* method [22, 71] to approximate the cost-to-go for MPC.

Another important problem, discussed in the introduction, is that the robustness of the openloop implemented MPC may be affected by process disturbance and noise. In this chapter, we have adopted and modified Rawlings and Mayne's tube-based approach [36] for the robust motion planning and control of mobile robots with imperfect state information. As mentioned earlier in the introduction, the MPC architecture is composed of two sections: *nominal MPC* and *ancillary MPC*. The first section, the nominal MPC, is a conventional MPC which is used to plan a nominal trajectory from the initial position to the target pose. The second section, the ancillary MPC, guarantees that the system trajectory is stabilized about the trajectory planned by the nominal MPC. The state estimation and control processes in nonlinear systems are coupled. The relation between optimal control and estimation has been discussed in the dual control theory literature during the past few decades [54]. In his seminal work, Bar Shalom showed that in nonlinear systems, any optimal control action must have three components: the certainty equivalent, cautiousness, and probing [72]. The certainty equivalent component controls the system irrespective of uncertainty, while the cautiousness component incorporates the impact of uncertainty, and finally the probing component minimizes uncertainty using future measurements [72]. In general, discarding the probing effect by ignoring future measurements previously formulated [36], leads to a so called open-loop optimization within the MPC context. This approach may result in an overly conservative controller. This issue has been raised in the set-theoretic control as well as probabilistic framework [73, 74]. On the other extreme, probing by incorporating all possible future measurements is the fully closedloop approach which, in essence, is just a theatrical concept and practically infeasible. A viable yet effective solution for incorporating future measurements, which we have used in this research, is the partially closed-loop strategy [75]. In this strategy, the most likely measurement is assumed as the future measurement. Du Toit has shown that for linear Gaussian systems, this assumption introduces the least information gain to the controller [28]. This strategy has been used frequently in the active sensing literature [57, 59, 60].

The main contribution of this chapter is the introduction of a probing tube-based MPC algorithm in an integrated motion planning and control algorithm. We have adopted and further improved the output-feedback tube-based MPC controller structure [36] that includes a nominal MPC and an ancillary MPC. In this work, we have enhanced the probing

component of the controller by incorporating a partially closed-loop strategy. The probing component is a component of the control action that actively reduces the uncertainty. We have developed our algorithm by integrating output-feedback tube-based nonlinear MPC and partially closed-loop strategy that use two different frameworks to describe uncertainty. Specifically, the uncertainty is modeled as nondeterministic (set-theoretic) and probabilistic in tube-based MPC [40] and partially closed-loop strategy [28], respectively. To take advantage of the robustness of the tube-based MPC and at the same time incorporate the future measurements effectively, we have developed a relationship between the settheoretic and stochastic paradigms in linear systems to unify the two uncertainty frameworks (see Section 2.3.1), and then we further extended the application of this relationship to nonlinear systems under certain assumptions. Then we have utilized the same relationship, as a heuristic, for the unicycle model which is smoothly nonlinear. As a result, we can claim that our proposed approach for constraint tightening has been carried out in a more systematic way than previously thought [36]. In addition to constraint tightening, the proposed approach allows for the inclusion of an active probing term in the controller cost function that can in turn facilitate active localization which is an important feature for realworld robotics applications.

Before elaborating the proposed algorithm, different aspects of the motion planning problem are discussed in the next section. These include the robot motion and sensing models, state estimation processes and the motion planning formulation in uncluttered deterministic environments. The relationship between the set-theoretic, stochastic frameworks and the proposed probing tube-based MPC are discussed in Section 2.3. Section 2.4 verifies the proposed method through both simulation and experiments with a two wheel differential drive robot. Some concluding remarks have been provided in Section 2.5.

2.2 Problem statement

2.2.1 Deterministic Motion Planning

The main focus of this research is on motion planning of a unicycle robot shown in Fig. 2-1, which is used extensively in the motion planning literature [76-78]. The robot position and heading angle $p = [x, y, \theta]^T$ are used to define the robot states. Considering linear and angular velocities as the robot input, $u = [\vartheta, \omega]^T$, the robot motion model can be defined as:

$$\begin{cases} \dot{x} = \vartheta \cos(\theta) \\ \dot{y} = \vartheta \sin(\theta) \\ \dot{\theta} = \omega \end{cases}$$
(2.1)

In Eq. (2.1), the control signal and disturbances are the deriving signals that act on the system in the same way. Without slipping, the robot kinematics satisfies the mobility constraint:

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0, \ \theta \ge 0$$
 (2.2)



Figure 2-1 Schematic of the two-wheeled mobile robot.

The typical challenges of motion planning for such a system are the system nonholonomicity and control action constraints. The emphasis on nonholonomicity is due to the fact that based on the well-known work of Brockett (1983), the nonholonomic robot model is uncontrollable for time-invariant smooth feedback control laws [79]. Also, in real situations, saturation of the control policies is inevitable. The focus of this research is on the discrete-time control of the nonholonomic systems. The discretized equations of motion can be derived from Eq. (1) with sampling time T as:

$$\begin{cases} x^{+} = x + \vartheta(\sin(\theta^{+}) - \sin(\theta))/\omega \\ y^{+} = y - \vartheta(\cos(\theta^{+}) - \cos(\theta))/\omega \\ \theta^{+} = \theta + \omega T \end{cases}$$
(2.3.a)

if $\omega \rightarrow 0$:

$$\begin{cases} x^{+} = x + \vartheta \cos(\theta) T \\ y^{+} = y + \vartheta \sin(\theta) T \\ \theta^{+} = \theta + \omega T \end{cases}$$
(2.3.b)
where x, y and θ are robot position and heading at sample time k. Also the superscript + denotes the value of the variable at sample time k + 1.

In motion planning of autonomous systems, the problem is to drive the robot from the initial position p_0 to the final position p_f under the robot input and state constraints. In the deterministic non-cluttered environment, where there is no noise or disturbance and the perfect state information is available, the motion planning problem at time k can be formulated as a dynamic programming problem:

$$\boldsymbol{u}^{*} = \underset{\boldsymbol{u}}{\operatorname{argmin}} \left(\sum_{j=0}^{N-1} l_{j} \left(p_{k+j} - p_{f}, u_{k+j} \right) + F_{N} \left(p_{k+N} - p_{f} \right) \right)$$
(2.4.a)

subject to:

$$p^+ = f(p, u) \tag{2.4.b}$$

$$u_{k+j} = \mathbb{U} \tag{2.4.c}$$

$$p \in \mathbb{P}_{safe} \tag{2.4.d}$$

$$p_N \in \mathbb{P}_f \tag{2.4.e}$$

where $\mathbf{U}^* = \{u_{k+j}\}, j = 0$ to N - 1, is the sequence of optimal control actions over the next N horizon. The stage cost and final cost are denoted by l_j and F_N , respectively. The stage cost l_j is chosen as an Euclidian norm: $l_j = \|\mathbf{p}_{k+j} - \mathbf{p}_f\|_{W_1} + \|\mathbf{u}_{k+j}\|_{W_2}$, where $\|\mathbf{v}\|_A = \|\mathbf{v}A\mathbf{v}^T\|_2$. The desired output can be shaped by tuning the matrices W_1 and W_2 [18, 19]. In the receding horizon strategy, the first M control actions of \mathbf{U}^* are applied before the new

set of the optimal control actions is calculated, where N and M are the prediction horizon and control horizon, respectively. The constraints of optimization are expressed in Eqs. (2.4.b) to (2.4.e). The Eq. (2.4.b) is the motion model of the robot. In Eq. (2.4.c and d), \mathbb{U} and \mathbb{P}_{safe} are the admissible sets of control actions and safe positions, respectively. The last constraint, Eq. (2.4.e) is an optional part of optimization that influences the stability and attractive region of the control system [18]. Beside this conventional MPC formulation, Eqs. (2.4) can be implemented with both control and state variables as the decision variables using large-scale sparse optimization toolboxes for practical use. The algorithm proposed in [80] is an example of this deterministic motion planning algorithm with proven stability and robustness.

The applicability of an oversimplified deterministic motion planning algorithm is limited by the assumptions of perfect state information and limited attractive region. The perfect state information is an invalid assumption in many real-world applications where positioning is noisy. Also, motion planning algorithms with a limited attractive region are impractical in cluttered environments.

2.3 State Estimation and Prediction

A large portion of the nonlinear state estimation in robotics literature is focused on the stochastic Bayesian framework which has been reviewed thoroughly [81]. In this research, we use the same framework mainly because of its computational efficiency [82]. A computationally efficient nonlinear filtering solution for stochastic Bayesian problems is the Extended Kalman Filter (EKF) that is commonly used in robotics applications despite its

shortcoming [83]. In the presence of disturbance and unmodeled dynamics, the equation of motion can be expressed as:

$$p^+ = f(p, u, \delta) \tag{2.2.5}$$

Where $\delta \in \mathbb{R}^2$ is composed of independent translational and rotational disturbances, and acts similar to input u on the robot. In order to use EKF, the process disturbance δ of Eq. (2.2.5) is approximated as a Gaussian white distribution $\delta_p \sim \mathcal{N}(0, Q_k)$ where Q_k is the disturbance covariance. We will elaborate on this approximation in the next section. The measurement equation can be expressed as:

$$y = h(p, v) \tag{2.2.6}$$

where v is the measurement noise and can be approximated with Gaussian white noise, $v \sim \mathcal{N}(0, R_k)$. In our experimental setup, which uses an overhead camera system as the measurement tool, the measurement model can be expressed as:

$$h(p, \mathbf{v}) = [x, y, \theta]^T + \mathbf{v}$$
(2.7)

and the measurement noise covariance $R_k = R_k(r)$ and $r = \sqrt{x^2 + y^2}$.

Using the *a priori* position estimation $p_k \sim \mathcal{N}(\hat{p}_{k|k}, \Sigma_{k|k})$, observation at the current time y_{k+1} , and the above motion and measurement models, the EKF algorithm can be expressed as [81]:

Prediction step:

$$\hat{p}_{k+1|k} = f(\hat{p}_{k|k}, u_k) \tag{2.8}$$

$$\Sigma_{k+1|k} = F_k \Sigma_{k|k} F_k^{\ T} + L_k Q_k L_k^{\ T}$$
(2.9)

Update step:

$$K_{k} = \sum_{k+1|k} H_{k}^{T} (H_{k} \sum_{k+1|k} H_{k}^{T} + M_{k} R_{k} M_{k})^{-1}$$
(2.10)

$$\hat{p}_{k+1|k+1} = \hat{p}_{k+1|k} + K_k(y_{k+1} - h(\hat{p}_{k+1|k}, 0))$$
(2.11)

$$\Sigma_{k+1|k+1} = (I - K_k H_k) \Sigma_{k+1|k}$$
(2.12)

where
$$F_k = \frac{\partial f}{\partial p}|_{\hat{p}_{k|k}, u_k}, L_k = \frac{\partial f}{\partial u}|_{\hat{p}_{k|k}, u_k}, M_k = \frac{\partial h}{\partial v}|_{\hat{p}_{k+1|k}}$$
 and $H_k = \frac{\partial h}{\partial p}|_{\hat{p}_{k+1|k}}$.

The last topic related to the state estimation in nonlinear systems is active probing. In the active probing systems, the control law or action is designed to minimize an index related to the system uncertainty, e.g. trace or determinant of the covariance matrix. As it is suggested in [56], for the robot exploration problem the a-optimal systems, minimizing the covariance matrix trace, outperform the d-optimal systems, minimizing the determinant of the covariance matrix.

2.4 Tube-based MPC

This section details the proposed integrated motion planning and control scheme based on the tube-based MPC [36]. The integrated motion planning and control problem is solved using a two-stage MPC consisting of a nominal MPC and an ancillary MPC. The nominal MPC is used for motion planning and features probing and also avoid trap situations. On the other hand, the proposed ancillary controller stabilizes the kinematic control around the planned trajectory and also increases the robustness of the nominal controller. The total control action is $u_k = u_k + v_k$, where u_k is the control action produced by the nominal MPC, and v_k is the control action produced by the ancillary MPC.

2.4.1 Relation between Set-theoretic and Stochastic Frameworks: An Illustrative Example

In order to formulate a robust probing MPC, we established a relationship between the stochastic framework [60] and set-theoretic framework [40]. In set-theoretic framework, the main algebraic operations of a set are defined as:

- a. Minkowski set addition: $A \oplus B \coloneqq \{a + b | a \in A, b \in B\}$
- b. Set subtraction (erosion): $A \ominus B \coloneqq \{x | \{x\} \oplus B \subseteq A\}$
- c. Set multiplication: $KA \coloneqq \{Ka | a \in A\}$

To illustrate the relationship between set-theoretic and stochastic frameworks, we consider the following linear system with deterministic disturbances w,

$$x^+ = ax + \delta \tag{2.13}$$

where $\delta \in \mathbb{W}$ and $\mathbb{W} = [-w, w]$ and $x_0 = 0$. The system state x_k belongs to the tube $S_k[84][35][30]$:

$$\mathcal{S}_{k} = \sum_{j=0}^{k-1} a^{j} \mathbb{W} = \mathbb{W} \bigoplus a \mathbb{W} \bigoplus \dots \bigoplus a^{k-1} \mathbb{W}$$
(2.14)

For a process with N + 1 steps:

$$S_N = \frac{1 - a^N}{1 - a} \mathbb{W} = \left[-\frac{1 - a^N}{1 - a} w, \frac{1 - a^N}{1 - a} w \right]$$
(2.15)

If δ is approximated by Gaussian white noise $\delta_p \sim \mathcal{N}(0, P_w)$ where P_w is the covariance of the distribution. In the stochastic framework, the state distribution can be approximated as:

$$x_k \sim \mathcal{N}(0, P_k) \tag{2.16}$$

$$P_k = \sigma_k^2 = \sum_{j=0}^{k-1} a^j P_w a^j$$
(2.17)

By comparing Eq. (2.15) and Eq. (2.17), one can see that in both cases the uncertainty propagates as a geometric series. In order to approximate the set-theoretic description with the stochastic description, we assume that S_N is a subset of a confidence interval of the state distribution in step N:

$$\left[-\frac{1-a^N}{1-a}w,\frac{1-a^N}{1-a}w\right] \subseteq \left[-\mathcal{C}\sigma_N,\mathcal{C}\sigma_N\right]$$
(2.18)

where the constant *C* is the confidence interval and σ_N is the standard deviation at time *N*; the smaller the confidence interval the more conservative the approximation. Eq. (2.18) can be written as:

$$\left(\frac{1-a^N}{1-a}w\right)^2 \le C^2 \sigma_N^2 \tag{2.19}$$

Considering Eq. (2.17)

$$\left(\frac{1-a^N}{1-a}w\right)^2 \le C^2 \frac{1-a^{2N}}{1-a^2} P_w \tag{2.20}$$

SO

$$\frac{(1-a^N)(1+a)}{(1+a^N)(1-a)}\frac{w^2}{C^2} \le P_w \tag{2.21}$$

For stable systems ($a \le 1$), the greatest noise impact occurs when $a \to 1$ that is:

$$P_w \to N \frac{w^2}{c^2} \tag{2.22}$$

Figure 2-2 shows how the uncertainties from the stochastic and set-theoretic frameworks propagate where a = 1, w = 1 and C = 3. Shown in Figure 2-2, the Monte Carlo simulation of the state evolution is close to the nondeterministic set-theoretic framework results. As a result, Eq. (2.22) is completely different from the conventional method of approximation of a set with a Gaussian noise with equal mean and covariance, which is common in estimation literature [85]. More precisely, using the conventional framework the covariance is calculated as: $P_w = \frac{w^2}{3}$. Considering these two methods, we will heuristically approximate each disturbance independently with a zero mean Gaussian noise, that is:

$$P_w = \max\{N\frac{w^2}{c^2}, \frac{w^2}{3}\}$$
(2.24)

An important point to note is whether or not Eq. (2.24) is applicable to the nonlinear system described by Eq. (2.3). The result of extensive numerical simulations shown in Fig. 2-4 verifies the applicability of the proposed approach to nonlinear systems because the stochastic description utilizing Eq. (2.24) successfully over-approximates the set-theoretic description

of the uncertainty. In Figure 2-5, the uncertainty has been added as the 5% of the feasible control signal. The results are the Monte Carlo simulations of the nonlinear model system, described in Eq. (2.3), with the set and probabilistic disturbances.

The remaining issue in relating the two frameworks is to replace a distribution with an infinite support to a bounded set. This can be achieved by truncating the distribution to the region corresponding to a confidence level, β , which satisfies the overall performance of the system in a realistic situation. For example, in the linear system example introduced previously, the time evolution of the confidence region $X_{\beta,k}$, for $\beta = 99.73\%$, is compared with the settheoretic formulation S_k in Figure 2-3. The confidence region corresponds to the confidence interval C selected earlier. As it is shown in Figure 2-3, the tube-based description and confidence regions hold a constant relation described as $S_k \subseteq X_{\beta,k}$ during the process. Extending this method to the multivariate situation has been discussed prior to this research [86]. Since collision occurs in the 2D geometry, the bivariate distribution is particularly important. In this project, we have utilized an over approximation of the confidence region instead of the exact confidence region to reduce the computational load. This concept has been shown in Figure 2-5 for a bivariate distribution. The basic idea is to over approximate the ellipse with a circumscribed circle or rectangle. The ellipse major diameter is an indicator of the confidence region and proportional to the square root of the largest eigenvalue of the covariance matrix. This method may be overly conservative, but it is computationally efficient. In summary, to approximate distribution $s \sim \mathcal{N}(0, \Sigma)$ with the set S the following algorithm is used:

25



Figure 2-2 The state uncertainty evaluation using the Monte Carlo simulation (color dots) and the set-theoretic framework (black bars) versus time.



Figure 2-3 The time evolution of the confidence region for $\beta = 99.73\%$ (blue bars) and the set-theoretic formulation (red bars).



Figure 2-4 Robot motion with constant rotational and translational velocity with only rotational disturbance (a), and with only translational disturbance (b). The nominal path is shown as a blue line, and the stochastic and settheoretic uncertainty discretions are shown with blue and red points, respectively.



Figure 2-5 The schematic of the confidence region in the plane, the exact confidence region (ellipse), and the circumscribed circle and square over approximating it.

Algorithm 1:

- Find the (over)approximation of λ , the largest eigenvalue of Σ
- \mathbb{S} is a circle to the radius of $c\sqrt{\lambda}$ or a square with the side equal to $2c\sqrt{\lambda}$

We will now utilize these relationships to introduce a probing robust MPC scheme. The main idea is that probing is carried out through a partially closed-loop strategy in the stochastic framework, and robustness is guaranteed using a set-theoretic framework. Details of the proposed scheme are discussed in the next section.

2.4.2 Developing Heuristic Relation between Set-theoretic and Stochastic Frameworks for Unicycle

To extend the results of Section 2.4.1 to the unicycle model, consider a general form of the linear systems of dimension *n*:

$$x^+ = Ax + \delta \tag{2.25}$$

As well as a linear transformation:

$$x = Rz, \delta = Rd \tag{2.26}$$

Where *R* is computed from the eigen-decomposition of the matrix *A*:

$$A = R\Lambda R^{-1} \tag{2.27}$$

Where $\Lambda = \text{diag}(\lambda_1, ..., \lambda_n)$, and λ_i is the *i*th eigenvalue of *A*. The system model based on *z* can be written as:

$$Rz^+ = ARz + \delta \tag{2.28}$$

As well as by left-multiplication of R^{-1} :

$$z^+ = \Lambda z + d \tag{2.29}$$

Since matrix Λ is a diagonal matrix, each row in Eq. (2.29) can be regarded as a decomposed set of first order linear systems analyzed in Section 2.4.1.

A similar analysis has been carried out on two systems:

$$x^+ = A_1 x + \delta$$
 and $x^+ = A_2 x + \delta$ where $A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0.75 \end{bmatrix}$, $A_2 = \begin{bmatrix} 0.875 & 0.125 \\ -0.125 & 0.875 \end{bmatrix}$.

Also, $\delta = [\delta_1, \delta_2]^T$ where $\delta_1 = \delta_2$, and $\delta_1 \in [-1,1]$. The uncertainty propagation in both systems is approximated with a probabilistic disturbance. As it is shown in Figure 2-6, the exact uncertainty set versus time related to A_1 and A_2 are shown in black squares and red rhombic shape, respectively. The stochastic approximation of each system is shown in green and blue ellipses, respectively.

Now consider the unicycle robot model, e.g., Eq. (2.3.b), as:

$$p^+ = f(p, u)$$
 (2.30)

Under the assumption that disturbance acts as a variation around the nominal trajectory, the disturbance linearized model is given by:

$$x^{+} = Ix + \begin{bmatrix} \cos(\theta) T & 0\\ \sin(\theta)T & 0\\ 0 & T \end{bmatrix} [\delta_{1}, \delta_{2}]$$
(2.31)

where δ_1 and δ_2 are the variation in velocity and rotational velocity, respectively. Now consider the following system:

$$z^{+} = Iz + T \begin{bmatrix} \delta_{11} \\ \delta_{12} \\ \delta_{2} \end{bmatrix}$$
(2.32)

Clearly, the uncertainty in Eq. (2.32) is over-approximation of Eq. (2.31) using the transformation Eq. (2.26).



Figure 2-6 The exact uncertainty set versus time related to A_1 and A_2 shown in black rectangles and red rhombic shape, respectively. The stochastic approximation of each system is shown in green and blue ellipses, respectively.

2.4.3 Nominal MPC

The nominal MPC is the extension of the deterministic motion planning to the cluttered dynamic environment in which only imperfect state information is available. In nominal MPC, the disturbance is initially approximated with the Gaussian white noise using the heuristics

developed earlier. Using this approximation, the future measurement can be incorporated using a Kalman filter-like algorithm. For incorporating the future measurements, the Gaussian description is computationally more efficient than the set-theoretic description [82]. However for collision avoidance, the state uncertainty is converted back to the set-theoretic description since integrating a Gaussian distribution over an arbitrary polyhedral set is computationally more demanding than adding the equivalent set using the Minkowski set addition. To clarify the notation, we need to distinguish between $p_{k+j|k}$ which is predicted by the overall motion planning scheme and $z_{k+j|k}$ which denotes the predicted state based on the nominal MPC only. Before describing the nominal motion planning algorithm, we focused on the partially closed-loop description of the state evolution. Using the EKF formulation we have:

$$\hat{z}_{k+1|k} = f(\hat{z}_{k|k}, \mathfrak{u}_k) \tag{2.33}$$

$$\Sigma^{-}_{k+1|k} = F_k \Sigma_{k|k} F_k^{\ T} + L_k Q_k L_k^{\ T}$$
(2.34)

$$K_{k} = \Sigma^{-}_{k+1|k} H_{k}^{T} (H_{k} \Sigma^{-}_{k+1|k} H_{k}^{T} + M_{k} R_{k} M_{k})^{-1}$$
(2.35)

$$\Sigma_{k+1|k} = (I - K_k H_k) \Sigma_{k+1|k}^{-}$$
(2.36)

where Σ , K and \mathfrak{u}_k are the covariance matrix, Kalman gain and the control action produced by the nominal MPC algorithm, respectively. Also $z_{k|k}$ and $p_{k|k}$ are identical distributions. As it is discussed in [36], \mathfrak{u}_k belongs to a subset of \mathbb{U} , $\mathfrak{u}_k \in \alpha \mathbb{U}$. Also, previous literature [75] has provided a detailed discussion on how the Eqs. (2.34-36) limit the growth of the uncertainty bound and reduce the overall uncertainty of the state prediction. In other words, incorporating future measurement will modify the uncertainty propagation from what is shown in Figure 2-2 and 3. Using Algorithm 1 discussed in the previous section, the prediction distribution $s_{k+j|k} \sim \mathcal{N}(0, \Sigma_{k+j|k})$ is replaced with the set $\mathbb{S}_{k+j|k}$ which corresponds to a 99.73% confidence interval. Therefore, we have:

$$z_{k+j|k} \in \hat{z}_{k+j|k} \bigoplus \mathbb{S}_{k+j|k} \tag{2.37}$$

This equation is later used to tighten the optimization constraints. The nominal MPC optimization can be formulated as:

$$\mathfrak{U}^{*} = \underset{\mathfrak{U}}{\operatorname{argmin}} (\sum_{j=0}^{N-1} l_{j} (\hat{z}_{k+j|k} - p_{f}, \mathfrak{u}_{k+j}) + F_{N} (\hat{z}_{k+N|k}, \Sigma_{k+N|k}))$$
(2.38.a)

subject to

$$\hat{z}^+ = f(\hat{z}, \mathfrak{u}) \tag{2.38.b}$$

$$\mathbf{u}_{k+j} \in \alpha \mathbb{U}, 0 < \alpha < 1 \tag{2.38.c}$$

$$\hat{z}_{k+j|k} \in \mathbb{P}_{safe} \ominus \mathbb{S}_{k+j|k} \tag{2.38.d}$$

In Eq. (2.38.c), α tightens the admissible control action set in for the nominal MPC and allows the ancillary control to compute the necessary control action to reject the disturbance. The Eq. (2.38.d) tightens the constraint equation to ensure the robust feasibility of the optimization. The final cost, or cost-to-go, consists of different parts:

$$F_N(\hat{z}_{k+N|k} - p_f, \Sigma_{k+N|k}) = F_{norm}(\hat{z}_{k+N|k} - p_f) + F_{LongTerm}(\hat{z}_{k+N|k} - p_f) + F_{probing}(\Sigma_{k+N|k})$$
(2.39)

The first term $F_{norm}(\hat{z}_{k+N|k} - p_f)$ is a cost proportional to the norm square of the estimated error, $\hat{p}_{k+N|k} - p_f$. The second term is related to the *coarse dynamic* planning cost. This term is especially important in cluttered environments with concave objects that can develop the trapping situation. To compute this term, the environment is discretized to a grid cell. Each cell is connected to its neighbors as a graph node if the neighbor node is not occluded by any obstacle. The occluded nodes are isolated islands and disconnected from the graph; the graph edges have the identical cost. The coarse dynamic cost of each node is proportional to the square of the shortest path length, $l_{shortestpath}$, calculated by the wavefront algorithm [11]. The occluded nodes are not in the feasible region, so no value is assigned to them. Once the shortest paths, from the entire graph nodes to the target point, are calculated offline, the long-term cost of each point can be computed using an online interpolation without solving the optimization problem based on the new states. To clarify this concept, the map used in simulation, the corresponding shortest paths, and the cost maps are shown in Figure 2-7 (a), (b) and (c), respectively. Shown in Figure 2-7 (c), the long-term cost map is not convex, but it has only one unique minimum point. The offline wavefront optimization can be solved using a lower frequency or an event-based scheme in the case of dynamic environments. The last term of the cost-to-go in Eq. (2.39) is the active probing term. As it was mentioned in the introduction, the estimation and control performance indices are coupled in nonlinear systems [87]. There are different ways to measure the estimation quality e.g., the trace or determinant of the covariance matrix. The trace of the covariance matrix has been used in this project as a more consistent measure for estimation quality. Substituting for each cost term, we have:

$$F_N(\hat{z}_{k+N|k}, \Sigma_{k+N|k}) = \gamma_0 \left\| \hat{z}_{k+N|k} - p_f \right\|_2 + \gamma_1 l_{shotestpath}^2 + \gamma_2 \operatorname{trace}(\Sigma_{k+N|k})$$
(2.40)

Where γ_0, γ_1 and γ_2 are nonnegative tuning parameters. The result of Eq. (2.38) is the sequence of the planned control action $\mathbf{u}^* = \{\mathbf{u}_k^* \quad \dots \quad \mathbf{u}_{k+N-1}^*\}$ which produces the optimal nominal trajectory $\mathbf{z}_k^* = \{\hat{z}_{k|k}^* \quad \dots \quad \hat{z}_{k+N|k}^*\}$. In a receding horizon strategy, the optimal trajectory is recomputed after the first M control action is executed. The ancillary controller robustifies robot motion about the optimal nominal trajectory. Details of this controller are discussed in the next subsection.

2.4.4 Ancillary controller

The ancillary controller is to ensure that the robot will follow the optimal nominal trajectory. Under the assumption that the control action is the sum of the control actions generated by both nominal and ancillary controllers i.e., $u_k = u_k + v_k$, the MPC formulation can be represented as:

$$\mathcal{V}^* = \underset{\mathcal{V}}{\operatorname{argmin}} (\sum_{j=0}^{N-1} \ell_j (\hat{p}_{k+j|k} - \hat{z}_{k+j|k}, v_{k+j}))$$
(2.41.a)

subject to





(b)



Figure 2-7 The map for simulation scenario (a), the shortest path value for the graph nodes (b), and the extracted long term cost map (c).

$$\hat{p}_{k+j+1|k} = f(\hat{p}_{k+j|k}, \mathfrak{u}_{k+j} + \sigma_{k+j})$$
(2.41.b)

$$v_{k+i} \in (1-\alpha)\mathbb{U} \tag{2.41.c}$$

$$\hat{p}_{k+N|k} = \hat{z}_{k+N|k} \tag{2.41.c}$$

Where ℓ_j is the stage cost, and $\mathcal{V} = \{ v_k \quad \dots \quad v_{k+N-1} \}$ is the ancillary control action sequence, respectively. In general, the stage cost of the ancillary controller is greater than the stage cost of the nominal controller. The ancillary controller can work at different frequencies. In our implementation, the ancillary controller has a frequency that is 3 times higher than the nominal controller.

If the conditions of the nominal and ancillary controllers are satisfied then $k \to \infty$, $\|\hat{p}_k - \hat{z}_k\| \to 0$ and $\|p_f - \hat{p}_k\| \to 0$, which mean that the estimation of the position states will merge to the final (desired) value. Simultaneously, by minimizing the covariance trace in the nominal controller, the estimation quality is enhanced. The stability conditions for deterministic nonlinear tube-based MPC have been discussed [16].

In summary, the proposed robust probing MPC algorithm can be summarized as follows:

Repeat

- Step 1. Compute the cost map that can be used for the long term (initially and when necessary).
- Step 2. Generate the nominal trajectory using Eq. (29) (every *M* sample).
- Step 3. Stabilize around the nominal trajectory using Eq. (32).

Step 4. Add and apply the control signal produced in Step 1 and Step 2 until the robot reaches the target.

The main thesis of this research is that the proposed algorithm is robust yet able to incorporate the future measurements. To verify this claim, the proposed integrated motion planning and control algorithm has been examined through numerical simulations and experiments in the next two sections.

2.5 Simulation Results

In order to investigate the feasibility of the proposed algorithm, extensive numerical simulations have been carried out. The simulation geometry and robot kinematics have been realistically defined based on the Quanser Qbot [88] which is the same robot used in the experiments. The scenario is designed to show the abilities of the motion planning and control scheme to be computationally efficient and robust. Shown in Figure 2-8, the robot must start from the initial point of $[-8,0,0]^T$ and reach the target point $[6,0,0]^T$. There are three obstacles in the pathway composing of a larger horseshoe-shape (concave) object. The robot is subject to both measurement noise and process disturbance. The maximum velocity and rotational velocity of the robot are 0.5 m/s and 2.5 rad/s, respectively. It is assumed that the maximum disturbance of the translational and rotational motions are 0.01 (m/s) and 0.05 (rad/s), respectively. Also the observation noise covariance is given by:

$$R_k = \text{diag}(1 + 0.02r, 1 + 0.02r, 1 + 0.05r)R_0$$
(2.42)

Where $R_0 = \text{diag}(0.1, 0.1, 0.0.5)$ and $r = \sqrt{x^2 + y^2}$. To analyze the performance of the proposed robust probing MPC algorithm, four simulations have been carried out.

Simulation I: This simulation evaluates the effect of the Euclidian term, long-term and probing term on the performance of the nominal MPC. Figure 2-8(a) shows the performance with all three terms in the cost-to-go function, whereas in Figure 2-8 (b), (c), (d), each lacks the Euclidian term, long-term and probing term, respectively. A comparison of Figure 2-8 (a) and (b) shows that the norm cost term smoothens the generated trajectory and reduces the steady state errors, whereas comparison of Figure 2-8 (a) and (c) indicates that without the coarse dynamic term, the motion planning algorithm converges to a local minimum and not to the target point. Furthermore, a comparison between Figure 2-8 (a) and (d) shows that the active probing term has a hardly noticeable effect on the path generated, although it was anticipated to bring the generated path closer to the origin. However, it should be noted that the active probing term will be more significant if the measurement model is nonlinear.

Simulation II: This simulation helps to understand the effect of the ancillary controller. Monte Carlo simulation of the motion planning process is carried out in two different cases: without ancillary controller, Figure 2-9 (a), and with ancillary controller, Figure 2-9(b). Each case was repeated seventy times. A qualitative comparison of Figure 2-9 (a) and (b) shows that the ancillary controller has successfully increased the system robustness, as the results remain more consistent in the presence of disturbance and measurement noise.



Figure 2-8 The nominal MPC performance with all term (a), without the norm cost (b), without the coarse dynamic term (c), and without the probing cost (d).



Figure 2-9 The Monte Carlo simulations of the motion planning process; without the ancillary control (a) and with ancillary controller (b).

Simulation III verifies the performance of the proposed method in dynamic environments. The long-term cost can be recalculated as the environment changes. To illustrate this concept, in this simulation, the third obstacle appears in the middle of the simulation process. The results of simulation without and with recalculating the long-term cost are shown in Figure 2-10 (a) and (b), respectively. It is not surprising that without recalculating the long-term cost the robot may be trapped in local minima. Even in that situation shown in Figure 2-10 (a), the robot is trapped but the path is still safe since there is no collision with the obstacles. However, recalculation of the long-term cost, shown in Figure 2-10 (b), enables the robot to avoid local minima and hence effectively handles the dynamicity of the environment. It should be noted that the offline cost computation using the wavefront algorithm is relatively expensive and therefore not suitable for dynamic environments with persistent rapid changes [11].



Figure 2-10 Coping with dynamic environments (a) without long-term cost recalculation, and (b) with long-term cost recalculation.

Simulation IV verifies the adequacy of the proposed method in tackling occlusion or the situation where the knowledge of the robot over its environment evolves over time. Shown in Figure 2-11, the robot starts from the starting point and moves towards the target point, where it is only aware of the obstacle closest to the starting point. At the point denoted by Event 1, the robot becomes aware of two other obstacles initially occluded by the first obstacle. Similarly at the point denoted by Event 2, the robot becomes aware of the last obstacle which is close to the target point. The planned path clearly shows that the motion planning scheme can incorporate incomplete information and at the same time take advantage of additional information as it becomes available.

In summary, it is shown that the nominal controller is able to produce a robust and probing motion plan capable of avoiding local minima. Also, the simulation demonstrates that the

ancillary controller reduces the disturbance effect on the system and stabilizes the robot trajectory about the nominal trajectory.



Figure 2-11 Robot path in occluded environment; robot motions begins from

2.5.1 Effect of Covariance Trace Optimization on Planning and Estimation Quality

As it is discussed in Section 2.3.2, by including the covariance trace in the nominal MPC cost function, the estimation quality is enhanced. However, in our experiment and simulations, the nonlinearity of the measurement model is not significant, and thus insignificant in the simulations. In this appendix, we have designed a new scenario to clarify the effect of this term. Hypothetically, we assume that:

$$R_k = \operatorname{diag}(1 + 0.05r^4, 1 + 0.05r^4, 1 + 0.05r^4)R_0$$
(2.43)

where $R_0 = \text{diag}(0.1, 0.1, 0.05)$ and $r = \sqrt{x^2 + y^2}$.

Also, since there is no obstacle in the simulation, the long term cost of the final cost-to-go is ignored. Therefore, the cost function of the planner only consists of the Euclidian and probing terms. The results of the planning with and without considering the covariance trace are shown in the Figure 2-12 (a-d). Comparison of the robot paths and covariance matrix norms without the probing term can be seen in Figure 2-12 (a), and (b), with the probing term; Figure 2-12 (c) and (d), indicates that the planning algorithm in the second case has improved the estimation quality at the price of a longer path and increased computational cost. Shown in Figure 2-12, addition of the probing term has enhanced the final value and reduction rate of the state uncertainty.





Figure 2-12 The nominal MPC performance without probing term: the robot path (a) and the covariance matrix norm (b); and with the active probing term: the robot path (c) and the covariance matrix norm (d).

2.6 Experimental Results

Real experiments on a robot were conducted to validate the feasibility of the proposed integrated motion planning and control system. The main goal of the experiment is to examine the performance of the integrated planning and motion control algorithm in a realworld situation. This includes the low-level unmodeled dynamic, suboptimal optimization results and unmodeled uncertainties in sensing and control signals. Before discussing the results, the testbed used for the experiments is introduced.

2.6.1 Experimental Testbed

The mobile robot used is a Quanser Qbot which is based on the iRobot Create platform. The Qbot is a differential drive robot that can be readily integrated with Matlab Simulink. A

Pixelink *PL-B778G* overhead camera is used to provide the exact robot pose as it moves within the environment. The experimental testbed developed in our laboratory can efficiently integrate different parts of the system regardless of their software platform. For example, the overhead camera image acquisition and object tracking software is implemented in C++, motion planning and motion execution is programmed in Matlab, and the robot control and networking is in Simulink. The interconnectivity of the testbed software is portrayed in Figure 2-12. Figure 2-13 shows the actual testbed composed of the overhead camera, Qbot robot marked by blue and red circles, and a host computer.

The optimization in the proposed MPC method is solved using Matlab fmincon function, which is not developed for real-time implementation. To address this issue, the maximum velocity of Qbot is set to 0.1 m/s, and the sampling time is limited to 1 sec. Also, to reduce the computational load, the probing weight is set to zero. An extended EKF is used to reduce the observation noise. The prediction and control horizons are set to 15 and 5, respectively.

2.6.2 Experimental Results

The environment of the experiment is a $3m \times 3m$ flat square. The robot starts from the initial pose $\begin{bmatrix} x & y & \theta \end{bmatrix}^T = \begin{bmatrix} -1 & -1 & 0 \end{bmatrix}^T$ and moves to the target point $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^T$ through two obstacles located along the y-axis at = -0.5, x = 0.5, respectively. The motion planning algorithm is unaware of the second obstacle at the beginning of the process.



Figure 2-13 Integration of Matlab based motion planning and C++ image acquisition and processing.

The robot observes the second obstacle as it passes by the first obstacle. Figure 2-15 shows the path taken by the robot, as well as the motion plan generated by the proposed motion planning method denoted by dashed lines (i.e. tangent to the robot path). The breakdown of the nominal, ancillary and overall control signal for each of the rotational and translational control commands are shown in Figure 2-15 and 2-16, respectively..

In the analysis of the experiment results, a few important points must be considered. Shown in Figure 2-15, the planned paths are tangential to the actual path. This is an indicator that the motion planning algorithm has sufficient robust performance. Also, comparison of

ancillary signals in Figure 2-16 and 2-17 indicates that the rotation signal is in the saturation more often than the translational signal. This means that the rotation is more prone to deviate from the planned trajectory. The result is consistent with the performance of the Qbot robots which are prone to more disturbances in the rotational maneuvers.

Another important observation is associated with the time intervals in which both the rotational and translational control signals drop to very small values that cannot overcome the internal nonlinearities such as internal friction of the robot mechanisms. More precisely, the computed control signals at those intervals are about 3 mm/s, which are attenuated by the unmodeled friction. One way to solve this problem is to set a threshold value u_{min} , which is the control signal at either zero or greater than u_{min} . The drawback of adding a threshold value is that the optimization would be mixed integer nonlinear programming (MINP), and not proper for practical implementation using the optimization tools used in our work. Modeling the effect of friction at low velocities and more computationally efficient optimization algorithms is one of the topics of future work.

2.7 Summary

This chapter presents a robust integrated motion planning and control scheme. The proposed scheme has two parts: a nominal MPC planner and an ancillary MPC controller. To prevent overly conservative behavior, future measurement is incorporated into the nominal controller by utilizing the partially closed-loop strategy. Incorporation of the future measurements not only enables the controller to be probing but also tightens the constraints in a systematic way. On the other hand, robustness of the planning algorithm is guaranteed

47

using the tube-based set theoretic framework. In the tube-based approach, the constraints are tightened in nominal MPC, and also an ancillary controller guarantees that the robot follows the nominal optimal trajectory. These two frameworks, namely stochastic and set theoretic, are related through a relationship developed for the linear systems, but it is shown that the approach can be equally effective for nonlinear systems. Extensive numerical simulations and experiments with a mobile robot have been carried out to examine the proposed algorithm performance and the functionality of its components.



Figure 2-14 The experimental setup including the Qbot robot, the overhead camera and the workstation which is used to process the data.



Figure 2-15 The experimental scenario: the robot path (black line), the planned paths (blue dashed line), the target (red circle) and the robot final pose uncertainty indicator (green circle).



Figure 2-16 The nominal, ancillary and overall rotational control signals.



Figure 2-17 The nominal, ancillary and overall translational control signals.

Based on the simulation results, it can be concluded that global performance of the algorithm is closely related to the coarse dynamic term, whereas the steady-state error and the planned trajectory smoothness is related to the norm cost. Since the observation equation is approximately linear, the active probing term does not have a significant effect on the simulation results. However, in the systems with a highly nonlinear observation model, this term can significantly improve the performance of the control system. The Monte Carlo simulation results indicate that the ancillary motion controller successfully stabilizes the robot motion about the nominal planned trajectory. Also, the motion planning algorithm was successfully examined in dynamic cluttered environments. In the next stage, the feasibility of real-time implementation is studied. The experimental results show that motion planning can be implemented in near real-time and successfully drive the robot to a target point.

Chapter 3. Unscented Model Predictive Control of Chance Constrained Nonlinear Systems

3.1 Overview

MPC-based methods have been used for motion control of mobile robots in the presence of localization uncertainty [89, 90] where the system dynamics is usually assumed to be linear. However, in practice, the dynamics of unmanned ground and aerial vehicles can be described more realistically using nonlinear and nonholonomic systems. Although there are previous studies on the application of MPC on both deterministic and stochastic nonlinear systems [17, 28, 30, 31, 42, 80], these studies have generally used Taylor series expansion for linearization [42, 75]. Our main contribution in this chapter is to present a new systematic way to generalize the linear stochastic MPC methods to nonlinear systems by approximating the state transition using an unscented transform that is a statistical linearization method. Our linearization approach offers two advantages over the traditional Taylor expansion linearization approach. First, statistical linearization outperforms analytical linearization methods in terms of prediction and estimation error [69]. Second, there is no need to calculate the Jacobian of the system and measurement models. The latter can be a critical advantage when the system model is complex, or simply unavailable. Although the unscented transform is computationally more expensive, the additional computational complexity is not a burden for robotic systems, especially for low dimensional systems [69].

The proposed unscented probabilistic MPC method is applied to the nonholonomic nonlinear robot motion control problem. The use of unscented transform allows for approximating the nonlinear state transition as a Gaussian distribution. The unscented transform extends the closed-form solutions available for linear systems to nonlinear systems. In this way, the algorithm tackles both the deterministic nonholonomic constraints and probabilistic collision avoidance chance constraints. The preliminary work on the proposed method has been briefly discussed [3]. In this Chapter, we have analyzed the advantages and limitations of the proposed approach in comparison with the conventional Taylor expansion based MPC.

The nonholonomic system model description, motion control formulation, and chance constraints will be discussed in Section 3.2. In Section 3.3, the formulation of the unscented predictive motion control will be discussed. Section 3.4 presents the validation of the proposed motion control algorithm through numerical simulation. Section 3.4 includes the concluding remarks and future work.

3.2 **Problem Statement**

3.2.1 The nonholonomic system model

Figure 3-1 shows the schematics of a two-wheeled mobile robot as a well-known nonlinear nonholonomic system to formulate the proposed motion control method. The robot states are defined using the robot position and rotation angle, $X = [x, y, \theta]^T$. The configuration space, *C*, has been considered as the state space; $X \in C = \mathbb{R}^2 \times S$. Considering linear and angular velocities as the robot input, $u = [\vartheta, \omega]$, the robot motion model can be defined as:

53
$$\begin{cases} \dot{x} = \vartheta \cos(\theta) \\ \dot{y} = \vartheta \sin(\theta) \\ \dot{\theta} = \omega \end{cases}$$
(3.1)

In Eq. (3.1), the deriving signals include both control signal and disturbances. If the robot moves without slipping, the robot kinematics satisfies a nonholonomic constraint:

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0, \ \theta \ge 0 \tag{3.2}$$



Figure 3-1 Schematic of the two-wheeled mobile robot [Figure 2-1 repeated].

The control system design for the stated nonholonomic system is a challenging task. The emphasis on nonholonomicity is due to the fact that based on the well-known work of Brockett (1983), the nonholonomic robot model is considered non-controllable for the time invariant feedback control laws [79]. Also, in real situations, saturation of the control policies is inevitable. The focus of this research is on the discrete-time control of the nonholonomic systems. By discretizing Eq. (2.1) based on the sampling time *T*, motion equations can be approximated as:

$$\begin{cases} x^{+} = x + \vartheta(\sin(\theta^{+}) - \sin(\theta))/\omega \\ y^{+} = y - \vartheta(\cos(\theta^{+}) - \cos(\theta))/\omega \\ \theta^{+} = \theta + \omega T \end{cases}$$
(3.3.a)

If $\omega \rightarrow 0$:

$$\begin{cases} x^{+} = x + \vartheta \cos(\theta) T \\ y^{+} = y + \vartheta \sin(\theta) T \\ \theta^{+} = \theta + \omega T \end{cases}$$
(3.3.b)

Where x_k , y_k , and θ_k compose the state vector X(k). It can be assumed that in the motion control problem, the pose error is defined based on the difference between the final destination and the current pose:

$$\chi_k = X_f - X_k \tag{3.4}$$

Therefore, by assuming $\chi_k = [x_e, y_e, \theta_e]^T$, the error dynamics can be defined as:

$$\begin{cases} x_e^+ = x_e - \vartheta \cos(\theta_e) T \\ y_e^+ = y_e - \vartheta \sin(\theta_e) T \\ \theta_e^+ = \phi(\theta_e - \omega T) \end{cases}$$
(3.5)

where $\phi \colon \mathbb{R} \to (-\pi \pi]$ is any mapping that provides a unique value for all the physically identical angles that have different numerical values. Now, the problem is to solve the motion control problem for the system defined in Eq. (5) which can be summarized as:

$$\chi_{k+1} = f(\chi_k, u_k) \tag{3.6}$$

3.2.2 The motion control formulation

The problem considered in this chapter is to find an optimal sequence of control policies to ensure that the robot reaches the goal region in finite time and avoids obstacles during the planning process. The optimality criterion is defined in terms of minimizing a cost function, which is usually an indicator of the efficacy of the controller. In the deterministic systems with perfect state information, the optimization can be defined as:

$$\min_{u} J(N, \chi, \{u_{n+i}\}) = \chi_{n+N}^{T} W_{N} \chi_{n+N}$$

$$+ \sum_{i=0}^{N-1} (\chi_{n+i}^{T} S_{i} \chi_{n+i} + u_{n+i}^{T} T_{i} u_{n+i})$$
(3.7)

subject to

$$\chi_{n+i+1} = f(\chi_{n+i}, u_{n+i}) \quad (a)$$

$$\chi_{n+i} \in X_i \quad (b) \qquad (3.8)$$

$$u_{n+i} \in U_i \quad (c)$$

where $u = \{u_{n+i}\}$ is the sequence of control actions over the control horizon *N*. The optimization result, $u^* = \arg\min(J)$, is the sequence of the optimal control policies over the control horizon *N*. Also, *n* denotes the current time sequence. Moreover, *S_i*, *T_i*, *W_N* are three positive definite weight matrices. The robot motion model is introduced to the optimization equation as one of the constraints. Eq. 3.8 (a) is the error dynamic. Also, Eqs. (3.8.b and 3.8.c) indicate that the robot states and control policies must be inside the feasible region of the optimization. In the motion control in unknown dynamic environments, different sources of uncertainties are introduced to the problem. These uncertainties may stem from different sources [26]:

- The imperfect state information; i.e. the robot and environment states are known as a probability distribution over possible states,
- The system model approximations and,

• Unmodeled disturbances.

By considering the existing disturbances, the system has to be modeled as:

$$\begin{aligned} \chi_{n+i+1} &= f(\chi_{n+i}, u_{n+i}, v_{n+i}) \quad (a) \\ y_{n+i} &= g(\chi_{n+i}, u_{n+i}, v_{n+i}) \quad (b) \end{aligned}$$
(3.9)

where y_n the measurement output, is related to system states and control inputs through a zeroth-order Markov chain g(.). Also, v_{n+i} and v_{n+i} are the motion disturbances and measurement noise, respectively. It is assumed that the system dynamics for the control input u_{n+i} and disturbance are the same. In order to handle the stated uncertainties, a stochastic dynamic programming (SDP) is defined as:

$$\min_{u} J(N, \chi, \{u_{n+i}\}) = E(\chi_{n+N|I_n}^{I} W_i \chi_{n+N|I_n} + \sum_{i=0}^{N-1} (\chi_{n+i|I_n}^{T} S_i \chi_{n+i|I_n} + u_{n+i|I_n}^{T} T_i u_{n+i|I_n}))$$
(3.10)

subject to:

$$\chi_{n+i+1} = f(\chi_{n+i}, u_{n+i}, v_{n+i})$$

$$y_{n+i} = g(\chi_{n+i}, u_{n+i}, v_{n+i})$$

$$P(\chi_{n+i|I_n} \in X_i) \ge p_i$$

$$u_{n+i} \in U_i$$

(3.11)

E(.) is the conditional expectation value operator, and I_i is the set of information, $I_i = \{I_{i-1}, u_{i-1}, y_i\}$ and $I_0 = y_0$.

In a stochastic environment, Yan and Bitmead have proposed a new stochastic dynamic programming (SDP) to solve the motion control problem [24]:

$$\min_{u} J(N, \hat{\chi}, u_{n}^{n+N-1}) = \hat{\chi}_{n+N}^{T} P_{N} \hat{\chi}_{n+N}$$

$$+ \sum_{i=0}^{N-1} (\hat{\chi}_{n+i}^{T} Q_{i} \hat{\chi}_{n+i} + u_{n+i|I_{n}}^{T} R u_{n+i|I_{n}})$$
(3.12)

subject to

$$P(\chi_{n+i|I_n} \in \mathbf{X}_i) \ge p_i$$

$$u_{n+i} \in U_i$$
(3.13)

where $\hat{\chi}_i$ is defined as the mean of the expected value of the robot states conditioned to the available information, $\hat{\chi}_i = E(\chi_i | I_i)$. The probabilistic constraint $P(\chi_{n+i|I_n} \in X_i) \ge p_i$ is a general expression for the chance constraints. This constraint will be discussed elaborately in the next subsection.

3.2.3 Constraint tightening

Since a good example to illustrate the benefit of the Gaussian state distribution is the ability to find the closed form solution for the collision avoidance constraints, we have dedicated this subsection to present the closed solution of the collision avoidance chance constraint based on methods previously discussed [91]. However, developing the closed form solutions for chance constraint is not the focus of this chapter.

In addition to uncertainty of the robot states, in unknown dynamic environments, the information of obstacles and other robots is obtained from an estimation process which is inherently uncertain. Therefore, the collision avoidance constraints in Eq. (8) must be expressed in terms of chance constraints $P(x_{i+1} \notin X_{free} | I_i) \leq \partial$, where X_{free} denotes the free space without any obstacles and ∂ denotes a confidence level. In order to satisfy this condition, the feasibility region should be tightened. As discussed earlier, *if state distributions are Gaussian or can be approximated by Gaussian distributions, it is possible to find a closed solution for some chance constraints* [24]. An example of such chance constraints is the non-collision chance constraint which is discussed in detail [28]; similarly to this previous work, in

this chapter, motion control involves a disk robot with radius ε trying to avoid point obstacles. The collision condition can be expressed as:

$$C: x^{o} \in \mathcal{B}(x^{R}, \varepsilon) \tag{3.14}$$

where x^{o} and x^{R} are the obstacle and robot location. Also $B(x^{R},\varepsilon)$ is the disk ball with the center of x^{R} and radius ε . The probability of collision can be defined as [28]:

$$P(C) = \int_{x^{R} x^{O}} I_{C}(x^{R}, x^{O}) p(x^{R}, x^{O}) dx^{R} dx^{O}$$
(3.15)

where I_c , the indicator function is defined as:

$$I_{C}(x^{O}, x^{R}) = \begin{cases} 1 & x^{O} \in B(x^{R}, \varepsilon) \\ 0 & Otherwise \end{cases}$$
(3.16)

As it is shown [91], if the robot state and obstacle distributions are Gaussian, the collision avoidance chance constraint can be calculated as:

$$\left(\hat{x}^{R} - \hat{x}^{o}\right)^{T} \Sigma_{C}^{-1} \left(\hat{x}^{R} - \hat{x}^{o}\right) \ge -2 \ln \left(\sqrt{\det(2\pi\Sigma_{C})} \frac{\partial}{V_{B}}\right)$$
(3.17)

where $V_{_B}$ is the area of the robot and $\Sigma_{_C} = \Sigma_{_{\chi}R} + \Sigma_{_{\chi}O}$.

The inequality of Eq. (3.17) is a deterministic equation that tightens the feasibility region of the optimization. An example of constraint tightening in Eq. (3.17) is shown in Figure 3-2. In this example, the values of \hat{x}° , Σ_{c} , $\hat{\partial}$ and V_{B} are chosen as $[0,0]^{T}$, I2, 0.01 and 1, respectively. The mean of the obstacle location distribution is shown in a blue point, and the infeasibility (collision) regions in deterministic and stochastic cases are shown in the inner red circle and outer green circle, respectively.



Figure 3-2 The obstacle location mean (the blue dot in the middle), the collision region without considering uncertainties (the inner red circle), and the collision region considering uncertainties (the outer green circle).

3.3 Unscented Model Predictive Motion control

As it is discussed in the previous section, collision avoidance constraints are introduced to stochastic dynamic programming (SDP) as chance constraints. In this section, we will elaborate on the unscented transform that provides Gaussian approximation for state distributions, and thus, a closed form solution for the chance constraints. At the end of Section 3.3, the final formulation of the unscented predictive motion control will be presented.

3.3.1 Unscented transformed motion equations

The unscented transform relies on neither analytical approximation methods such as the Taylor expansion nor the Monte Carlo random sampling. In this transform, the *points* are extracted deterministically from the input distribution. By mapping these points through nonlinear functions, the Gaussian approximation of the output distribution would be constructed.

Assuming that χ_{n+i} is known as an n_d -dimensional normal distribution with mean and covariance matrix of μ_{χ} and Σ_{χ} , respectively, and the approximated distribution of χ_{n+i+1} is constructed by utilizing the sigma points, the $2n_d + 1$ sigma points $\chi^{[j]}$ are calculated by the following rule:

$$\chi^{[0]} = \mu_{\chi}$$

$$\chi^{[j]} = \mu_{\chi} + \left(\sqrt{(n_d + \lambda)\Sigma_{\chi}}\right)_j \quad \text{for } j = 1, \dots, n_d \quad (18)$$

$$\chi^{[j]} = \mu_{\chi} - \left(\sqrt{(n_d + \lambda)\Sigma_{\chi}}\right)_{j-n_d} \quad \text{for } j = n_d + 1, \dots, 2n_d$$

where $\lambda = \alpha^2 (n_d + \kappa) - n_d$, with α and κ are scaling parameters of the approximating Gaussian distribution [69]. In order to calculate the approximated mean and covariance, each sigma point $\chi^{[j]}$ has two weights: $w_m^{[j]}$ related to the mean and $w_c^{[j]}$ related to the covariance.

$$w_m^{[0]} = \frac{\lambda}{n_d + \lambda}$$

$$w_c^{[0]} = \frac{\lambda}{n_d + \lambda} + \left(1 - \alpha^2 + \beta\right)$$

$$w_c^{[i]} = w_m^{[i]} = \frac{\lambda}{2(n_d + \lambda)} \quad \text{for } j = 1, ..., 2n_d$$
(19)

The parameter β is used to incorporate the knowledge about the distribution, which is approximated by the Gaussian distribution. For an exact Gaussian distribution, $\beta = 2$ is the optimal choice [69]. By mapping through the nonlinear function, the estimation mean $\overline{\chi}_{n+i+1}$ and covariance $\overline{\Sigma}_{n+i+1}$ of χ_{n+i+1} is estimated as:

$$\overline{\chi}_{n+i+1} = \sum_{j=0}^{2n_d} w_m^{[j]} \overline{\chi}^{[j]} \quad (a)$$

$$\overline{\Sigma}_{n+i+1} = \sum_{j=0}^{2n_d} w_c^{[j]} (\overline{\chi}^{[j]} - \overline{\chi}_{n+i+1}) (\overline{\chi}^{[j]} - \overline{\chi}_{n+i+1})^T + R_{n+i} \quad (b)$$
(20)

where R_{n+i} is the noise covariance of the system disturbance v_{n+i} , and also:

$$\overline{\chi}^{[j]} = f(\chi^{[j]}, u_{n+j})$$
(21)

It is possible to summarize Eq. (3.20) as a nonlinear mapping approximating the system equation and write it as:

$$[\overline{\chi}_{n+i+1}, \overline{\Sigma}_{n+i+1}] = h(\overline{\chi}_{n+i}, \overline{\Sigma}_{n+i}, u_{n+i}, R_{n+i})$$
(22)

The mapping *h* is Gaussian, so, it can be used to find the closed solution of the chance constraints. It should be noted that the unscented transform described in this section must be modified for the non-additive disturbance. A well-known modification of the unscented transform for non-additive disturbance is augmented unscented transform [92]. It should be noted that $\bar{\chi}_{n+i}$ is the open-loop estimation of χ_{n+i} in which future measurement is ignored, thus showing that Eq. (3.22) is overly conservative. The idea of incorporating future measurements is discussed in the following sections.

3.3.2 Unscented model predictive motion control using output feedback

The last piece of the unscented model predictive motion control algorithm is the incorporation of the future measurements. The idea of utilizing future measurement information can be traced back to the *dual control theory* [54]. Also, this notion is widely used in active sensing literature [57]. Incorporating the future measurement information into the model predictive algorithm prevents the control system to be overly conservative while preserving the system cautiousness [28]. In the unscented model predictive motion control algorithm, the future information is incorporated to the system through the UKF. Assuming

that g(.) in Eq. (3.9b) is a nonlinear function with additive noise covariance, Q_{n+i} . In the UKF algorithm, incorporating the measurement information commences with computing χ^*_{n+i+1} , which is the vector of sigma points of the normal distribution of $\mathcal{M}(\bar{\chi}_{n+i+1}, \bar{\Sigma}_{n+i+1})$. The algorithm proceeds as:

$$\begin{split} \overline{Y}_{n+i+1} &= g(\chi_{n+i+1}^{*}) \\ \hat{y}_{n+i+1} &= \sum_{j=0}^{2n_{d}} w_{m}^{[j]} \overline{Y}_{n+i+1}^{[j]} \\ S_{n+i+1} &= \sum_{j=0}^{2n_{d}} w_{c}^{[j]} (\overline{Y}_{n+i+1}^{[j]} - \hat{y}_{n+i+1}) (\overline{Y}_{n+i+1}^{[j]} - \hat{y}_{n+i+1})^{T} + Q_{n+j+1} \\ \Sigma_{n+i+1}^{x,y} &= \sum_{j=0}^{2n_{d}} w_{c}^{[j]} (\chi_{n+i+1}^{*[j]} - \overline{\chi}_{n+i+1}) (\overline{Y}_{n+i+1}^{[j]} - \hat{y}_{n+i+1})^{T} \\ K_{n+i+1} &= \sum_{n+i+1}^{x,y} S_{n+i+1}^{-1} \\ \hat{\chi}_{n+i+1} &= \overline{\chi}_{n+i+1} \\ \Sigma_{n+i+1} &= \overline{\Sigma}_{n+i+1} - K_{n+i+1} S_{n+i+1} K_{n+i+1}^{T} \end{split}$$
(23)

In essence, the future measurement y_{n+i+1} is a virtual quantity equal to the most likely measurement [59, 75]. This assumption does not influence the predicted state means, $\hat{\chi}_{n+i+1} = \overline{\chi}_{n+i+1}$ but reduces the estimation covariance, $\overline{\Sigma}_{n+i+1}$, and increases the feasibility region for the optimization Eq. (3.10). Finally, it is possible to express the unscented predictive motion control in its final form:

$$\min_{u} J(N, \hat{\chi}, u_{n}^{n+N-1}) = \hat{\chi}_{n+N}^{T} W_{N} \hat{\chi}_{n+N} + \sum_{i=0}^{N-1} (\hat{\chi}_{n+i}^{T} S_{i} \hat{\chi}_{n+i} + u_{n+i|I_{n}}^{T} T_{i} u_{n+i|I_{n}})$$
subject to Eqs. (17, 22, 23) and $u_{n+i} \in U_{i}$. (24)

The cost function in Eq. (3.24) is the certainty equivalent of the cost function of Eq. (3.7), whereas Eqs. (3.17, 3.22, and 3.23) are the collision avoidance, state transition, and the

uncertainty evolvement constraints, respectively. It should be noted that since the unscented transform is a deterministic transform, Eq. (3.24) has the same dimension as the optimization in Eq. (3.10). The other important point is that the proposed control approach is an output feedback approach. An appropriate observer is required for a complete MPC; the convergence and efficiency of the observer is also an important research subject, but it is outside the scope of this work. An EKF is used as an observer in our simulations. The overall system block diagram is shown in the Appendix to distinguish between the EKF estimator and the UKF-like predictive algorithms. Also, the tracking information comes from a virtual filter with asymptotic convergence. In the next section, the performance of the proposed unscented MPC (UMPC) is examined through different numerical simulations.

3.4 The Overall System Architecture

The motion planning algorithm discussed in this chapter is an output feedback predictive control scheme for the nonlinear stochastic systems. In the architecture of such a control scheme, there are two similar sub-systems with distinct functionalities: a stochastic predictor to predict the system state, and a stochastic filtering scheme to estimate the system states. This architecture is shown in Figure 3-3.

64



Figure 3-3 The function block diagram of the predictive controller (red), mobile robot (blue), and the state estimator (green).

Based on Figure 3-3, the difference between the UMPC and conventional MPC (CMPC) schemes goes back to the stochastic predictor of the inside of the controller, which is a UKF-like algorithm in UMPC, and EKF-like algorithm in CMPC. However, both schemes utilizes the same state estimator, e.g. EKF, in this chapter.

3.5 Simulation Results and Discussions

In this section, the accuracy of the unscented transform approximation is compared with an analytical approximation in one point. To do so, it is assumed that $\chi_0 = \mathcal{M}([0,0,0]^T, [1 \ 0 \ 0,0 \ 1])$ is selected as the initial pose distribution. It is assumed that $u = [1,1]^T$ and $R = [1 \ 0;0 \ 0.1]^T$. The χ_1 distribution is constructed by both the Monte Carlo simulation and the unscented transform. Ten thousand particles are used in the Monte Carlo simulation. Since the disturbance is not in the additive form, an augmented unscented transform is used

[92]. The unscented transform parameters are selected as $\alpha = 0.001$, $\kappa = 1$, and $\beta = 2$. The results of the mean $\overline{\chi}_1$ and Euclidian norm of the covariance matrix $\|\overline{\Sigma}_1\|_2$ are shown in Table 3-1.

Table 3-1 Approximation of the mean	and covariance	matrix norm.
-------------------------------------	----------------	--------------

	Monte Carlo simulation	Unscented Transform	Analytical Linearization
$\overline{\chi}_1$	[0.0620,-0.0017,-0.0012] ⁷	$\begin{bmatrix} 0.0677, 0, 0.1 \end{bmatrix}^{T}$	[0.1, 0, 0.1] ⁷
$\left\ \overline{\Sigma}_{1}\right\ _{2}$	1.0592	1.0363	1.0010

The comparison between analytical linearization and unscented transform indicates that the Mahalanobis distance of the unscented transform approximation is 90 percent of the analytical linearization approximation.

In this section, the performance of the proposed motion control approach is examined for different situations and compared to that of the conventional (linearization-based) MPC (CMPC) approach, in which the motion model is approximated by an analytical linearization method [28]. In the CMCP control scheme, the uncertainty propagation is approximated using an analytical linearization, similar to the method used in the EKF algorithm [12]. In the output feedback control systems, the state estimation quality plays an important role in the overall control system efficiency. As a result, the estimation process outputs are shown for all of the

simulations to gain better understanding of the system performance. The comparison between UMPC and CMPC is done both qualitatively by comparing the path as well as quantitatively using a fuel index and the *relative CPU time*. The fuel index is defined as: $FuelIndex = \|\sum |u_i|\|$, where the summation operates over the entire control process time. The relative CPU time is defined as the ratio of total optimization time of UMPC over the total optimization time of CMPC.

The unscented predictive motion control algorithm is applicable to both linear and nonlinear measurement systems. However, it is assumed that the measurement equation is linear:

$$g(\chi_{n+i}, \upsilon_{n+i}) = \chi_{n+i} + \upsilon_{n+i}$$
(3.25)

 $_{v}$ is the additive noise signal. The system disturbance and measurement noise are considered as a zero mean Gaussian with the covariance matrices of $R = I_2$ and $Q = 0.1 \times I_3$ where *I* denotes the identity matrix. Also *R* and *Q* are covariance of the motion disturbance and measurement signals, respectively. The numerical values are tentatively close to a typical fast indoor robot such as Pioneer LX Research Platform [93]. The sensor model is the simplified model for the over-head camera localization systems which is used in UBC ACIS Laboratory and is discussed in more details in [1]. However, it is important to evaluate the performance of a controller, not necessarily for the numerical value per se, but to analyze the signal to noise ratio of control and the measurement signal. The control inputs have a saturation magnitude of 5. The collision avoidance confidence level and robot area are selected as $\delta = 0.99$ and $V_B = 1$, respectively. In all cases, the initial robot pose covariance is $P_0 = I_3$. The MPC weight matrices are chosen as $S_i = I_3$, $T_i = I_3$ and $W_N = 100 \times I_3$. Also, it is assumed that the estimation of the location of obstacles, both static and dynamic, is available via an external estimator. In this chapter, it is assumed that the uncertainty related to each obstacle location is stationary through simulation.

In the first simulation, the robot motion is planned in absence of any obstacle. In the second simulation, the robot travels in a static environment. Finally, in the third case, the robot travels in a dynamic environment where an obstacle moves perpendicular to the direction of the robot.

To have a case for comparison, in the first simulation, it is assumed that the robot motion control is only influenced by the uncertain initial pose, severe driving signal disturbances, and measurement noise. The estimated initial pose is assumed as $\hat{X}_0 = \begin{bmatrix} -20 & 0 & \pi/2 \end{bmatrix}^T$. The executed and estimated robot paths are shown in Figure 3-4. In the second simulation, it is assumed that the robot is initially posed at $X_0 = \begin{bmatrix} -20 & 0 & 0 \end{bmatrix}^T$. The simulation target is to reach the origin. The obstacles are stationary and their location estimation is subject to a distribution, which is defined as $x^0 = \mathcal{M}([-10 \ 0]^T, I_2)$. The robot paths obtained in the presence of an obstacle using the UMPC and CMPC approach is shown in Figure 3-5.



Figure 3-4 Robot path in the first case; the robot target is the origin. The path produced by UMPC and CMPC is shown in blue and red, respectively. The estimated and true paths are shown with the dotted and solid lines, respectively.



Figure 3-5 Simulation II - The robot path is calculated by the UMPC and CMPC from the initial location (-20,0) to the destination (0,0) in presence of a stationary obstacle denoted by a green partial ellipse tentatively proportional to the covariance of the obstacle location. The path produced by UMPC and CMPC is shown in blue and red, respectively.

The obstacle is shown in a green ellipse with its size tentatively proportional to the covariance of obstacle location. The time evolution of the robot in x-direction and y-direction are shown in Figure 3-6 and 3-7, respectively. These figures indicate that UMPC and CMPC perform differently to avoid the obstacle.

In the third simulation, the motion control algorithm is examined in a dynamic environment. In general, other moving objects can be treated as cooperative, neutral or adversarial to the robot. In this simulation, it is assumed that the other object has its own neutral motion, which is perpendicular to the robot path. The robot starts from the same initial pose of the second simulation. Also, it is assumed that there is a reliable tracking system (an oracle) that is able to provide a proper estimation of the object location. The obstacle location estimation is defined as $x^{O} = \mathcal{M}[-10 \ 1-0.035t]^{T}$, I_{2}), where *t* denotes the time. The robot mission is to avoid the dynamic object and reach the origin. The robot paths using UMPC and CMPC methods are shown in Figure 3-8. Also, the robot motion in *x*-direction and *y*-direction are shown in Figure 3-9 and 3-10, respectively.

As it was mentioned earlier, in addition to the planned and executed paths, the fuel index and computational time are other important quantities in comparison with the control algorithms. The relative values of these quantities (UMPC's value over CMPC's value) are presented in Table 3-2.

The results of the first simulation indicate that in the absence of any obstacles the performances of the UMPC and CMPC are very similar. This observation is not surprising; since the optimal path is close to a straight line and robot states remain in a limited range

and therefore system nonlinearity does not have a significant effect. Based on our analysis, as the nonlinearity becomes more significant, which is the case for the second and third simulations, the performance of UMPC and CMPC will be more distinct.



Figure 3-6 Simulation II - The robot motion in x-direction. The performances of the UMPC and CMPC are shown in blue and red, respectively.



Figure 3-7 Simulation II - The robot motion in y-direction. The performances of the UMPC and CMPC are shown in blue and red, respectively.



Figure 3-8 Simulation III - The robot path is calculated by the UMPC and CMPC from the initial location (-20,0) to the destination (0,0) in presence of a dynamic obstacle denoted by an ellipse tentatively proportional to the covariance of the obstacle location.



Figure 3-9 Simulation III - The robot motion in x-direction. The performances of the UMPC and CMPC are shown in blue and red, respectively.

The second simulation examines the obstacle avoidance ability of the proposed UMPC method and compares it with that of CMPC, where the obstacle and robot pose are known only in terms of an uncertain belief. Both CMPC and UMPC have reached the target and avoided the obstacle, as well as maintained the collision probability below 0.01. Similar to previous research [24], the collision probabilities are found by:

$$P(C) = V_B \times \frac{1}{\sqrt{\det(2\pi\sum_{C})}} \exp\left[-\frac{1}{2}(\hat{x}^R - \hat{x}^O)^T \sum_{C} (\hat{x}^R - \hat{x}^O)\right]$$
(3.26)

In this way, the maximum collision probabilities for UMPC and CMPC are 0.0016 and 0.0001, respectively, which are both considerably less than the specified threshold. However, it is worth to add that a more efficient planning approach is the one that remains closer to the desired collision probability and prevent an overly conservative path. Clearly, a safer planning can still be achieved by reducing the collision probability to a lower desired threshold.



Figure 3-10 Simulation III - The robot motion in y-direction. The performances of the UMPC and CMPC are shown in blue and red, respectively.

The CMPC solution however seems more conservative as the planned path is significantly further away from the obstacle. Also, the relative fuel index of UMPC is smaller than that of the CMPC. The only drawback is that the UMPC has a higher processing time than CMPC.

	Relative fuel index	Relative CPU time
Second case	0.9269	1.12
Third case	0.9251	1.25

The third simulation examines collision avoidance in a dynamic environment. As it is shown in Figure 3-8, the obstacle starts moving in the horizontal plane. The solid cyan ellipse is tentatively proportional to the covariance of the obstacle. Two different ellipses are used in Figure 3-8 to show two important instants of the simulation; first, at the beginning of the simulation, and second, while the robot is passing the major axis of the ellipsoid. By comparing the maximum deviation in the *y*-direction in Figure 3-9 and 3-10, it can be concluded that the UMPC approach shows a less conservative performance. Also, similar to the static obstacle case, the relative fuel index of UMPC is smaller than CMPC. However, UMPC is computationally more expensive. The other criterion for the comparison of the UMPC and CMPC is the steady state error. As shown in Figure 3-10 and 3-7, the steady state error of the UMPC is smaller than the CMPC. It should be noted that no matter how close the robot is to the target, the disturbance does not allow error to remain zero.

In our analysis, the improved performance of the UMPC can be explained with regards to two major factors. First, UMPC can handle the Jacobian free input-output relationship, so the system equation can be introduced as the original continuous time, such as Eq. (3.1), instead of the discrete time approximation such as Eq. (3.3). The other important factor is the established fact (in estimation literature) that unscented transform handles the nonlinearity better than the analytical linearization [57] and thus, the UMPC utilizes a better prediction scheme.

An avid reader may wonder why overly conservative planning is not desirable. In modern motion planning literature, the risk of failure is considered as a planning resource which can

75

be adopted to create a fine balance between safety and efficiency. More precisely, an efficient motion planning algorithm may slightly increase the probability of collision to gain a larger profit [29]. The over conservative planning tends to disregard this utility and therefore gains a suboptimal outcome. This concept may be clearer in the context of two robots racing to reach a common target while maintaining the probability of collision under 0.01. Although an overly conservative planning may result in a remarkably lower collision probability <<0.01, a more desirable solution is the one that maintains the probability closer to the desired 0.01 threshold but yields the shorter path to win the race. The proposed UMPC solution provides a mean for manipulating the probability more precisely.

In summary, the UMPC has important advantages over the CMPC, i.e. it is a Jacobian free method, and it is more convenient to be applied to complicated systems. Also, it shows a less conservative behaviour, and it has a relatively smaller fuel index. However, these advantages come at the price of a higher computational cost.

3.6 Summary and Future Work

The use of unscented transform for approximation of state transition is introduced for model predictive control (MPC) of nonlinear stochastic systems. The proposed MPC approach is applied to the motion control problem of a unicycle in unstructured environments. The unscented transform MPC (UMPC) incorporates different types of constraints, including kinodynamic and collision avoidance chance constraints where the unscented transform is used to find a closed-form solution for the chance constraints. The performance of the proposed predictive motion control algorithm is examined and compared to conventional

MPC (CMPC) by simulating the unicycle motion control problems in three different scenarios including: i) a static environment with no obstacle, ii) a static environment with an obstacle, and iii) a dynamic environment with a moving obstacle.

In summary, the proposed UMPC has several advantages over CMPC. First, the UMPC is a derivative or Jacobian free formulation. Second, it can avoid an overly conservative response with respect to the motion planning problem, which may lead to planning infeasibility. Finally, the proposed approach will result in lower energy consumption, denoted by the fuel index in our simulations. However, these advantages come at the price of a higher computational cost. Thus, the preference of UMPC over CMPC is a trade-off between the energy versus computational efficiencies. More precisely, CMPC will still be a better choice for the problems with high computational complexity, whereas UMPC will be desirable to improve the controller performance in terms of prediction error as well as energy consumption.

Chapter 4. Robust Active iSAM

4.1 Overview

Introduced in Chapter 1, SLAM is the ability of a robot to build a map of its surrounding and locate itself within the map. The focus of this chapter is on the active SLAM that is the problem of optimizing the robot trajectory to improve the performance of the SLAM algorithm. Also, it was shown in Chapter 1 that most current active SLAM algorithms commonly use:

- 1- An MPC scheme on the motion planning level.
- 2- Filtering based methods for SLAM.

However, as it has been shown in Chapter 2, the MPC scheme solved numerically is sensitive to disturbance and noise. Also, modern optimization based SLAM methods have not been used in the active SLAM context. In this chapter, a new active SLAM algorithm is suggested by focusing on the following two points: i) robustifying the MPC scheme, and ii) the use of optimization instead of filtering to solve the SLAM problem. Among the various methods introduced that have been discussed to increase the robustness of MPC, we will explore the tube-based MPC, which was elaborated in Chapter 2. As it is discussed earlier, in the nonlinear tube based MPC, the controller consists of two MPC controllers: one nominal MPC that is a conventional MPC with constraint tightening, and one ancillary MPC that stabilizes the system around the nominal trajectory. Also, in the proposed active SLAM method, the nominal controller utilizes an optimization based SLAM scheme, viz. iSAM 2 algorithm, instead of the common filtering methods such as EKF or particle filtering. The tube-based MPC can be implemented using the set algebra efficiently, whereas the iSAM 2 algorithm utilizes Gaussian distributions to model the uncertainty. Therefore, an important part of developing the new active SLAM method is to relate uncertainty in these two frameworks.

Before elaborating the proposed MPC based motion planning and control scheme, the assumptions and overall system setting, active SLAM scheme, and incremental smoothing and mapping (iSAM 2) method are discussed in Section 4.2. The proposed robust active SLAM method is discussed in Section 4.3. The proposed algorithm performance is examined through numerical simulations discussed in Sections 4.4. Section 4.5 includes the concluding remarks.

4.2 **Problem Formulation**

4.2.1 iSAM 2 algorithm

In the smoothing and mapping (SAM) algorithm, the estimation algorithm is represented graphically using a factored graph. A factored graph is bipartite graph G, with two sorts of nodes, *factor nodes* $f_i \in \mathcal{F}$, and variables nodes $\theta_j \in \Theta$. Each edge $\varepsilon_{ij} \in \mathcal{E}$ connects a variable node to a factor node [94]. Based on the graph G, the function $f(\Theta)$ can be factored as:

$$f(\Theta) = \prod_{i} f_{i}(\Theta_{i}) \tag{4.1}$$

where Θ_i is the set of variables θ_j connected to the factor f_i through ε_{ij} . In the SLAM problem, Θ consists of landmarks and robot history states. The estimation goal is to find the variable set Θ^* such that:

$$\Theta^* = \operatorname*{argmax}_{\Theta} f(\Theta) \tag{4.2}$$

Under the assumption of Gaussian noise for the measurement models [44, 45]:

$$f_i(\Theta_i) \propto \exp\left(-\frac{1}{2} \|h_i(\Theta_i) - y_i\|_{\Sigma_i}^2\right)$$
 (4.3)

where h_i and y_i are a nonlinear measurement function and a measurement value, respectively. Also, $\|.\|_{\Sigma}^2$ is the squared Mahalanobis distance with the covariance matrix Σ . Using Eq. (4.1) and (4.3), Eq. (4.2) can be written as:

$$\Theta^* = \arg\min_{\Theta}(-\log f(\Theta)) = \arg\min_{\Theta}\left(\frac{1}{2}\sum_i \|h_i(\Theta_i) - y_i\|_{\Sigma_i}^2\right)$$
(4.4)

The Eq. (4.4) is a nonlinear least squares optimization, which can be solved using Gauss-Newton or Levenberge-Marquardt methods [95]. A typical SLAM problem leads to a relatively large optimization problem, which can be solved only as a batch optimization [96, 97]. However, Kaess et al recently proposed a iSAM 2.0 algorithm that utilizes the Bayes tree data structure to solve the Eq. (4.4) incrementally [53]. The iSAM2 algorithm utilizes *incremental variable re-ordering* and *fluid relinearization* to update only the nodes that are affected by the new measurement, and thus perform the partial and incremental update of the SLAM optimization solution. In this way, iSAM2 provides both real-time and highly accurate results. To solve the optimization problem, the least squares problem is solved through iterative linearization around a linearization point:

$$\operatorname{argmin}_{\Delta}(-\log(f(\Delta))) = \operatorname{argmin}_{\Delta}(\|A\Delta - \mathbf{b}\|^2)$$
(4.5)

where A, Δ and \mathbf{b} are the Jacobian matrix, the variation vector, and a constants vector, respectively. At the end of each iteration, the new linearization point is given as $\Delta \oplus \Theta$. The operator \oplus may be the simple addition or the addition suitable for over-parameterized 3D representation. The norm in Eq (4.5) is the new representation of the Mahalanobis distance as:

$$\|\boldsymbol{\Delta}\|_{\Sigma}^{2} = \boldsymbol{\Delta}^{T} \boldsymbol{\Sigma}^{-1} \boldsymbol{\Delta} = \boldsymbol{\Delta}^{T} \boldsymbol{\Sigma}^{-\frac{T}{2}} \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\Delta} = \left\|\boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\Delta}\right\|^{2}$$
(4.6)

After convergence to Θ^* , the information matrix $\mathcal I$ can be retrieved as:

$$\mathcal{I} = A^T A, \tag{4.7}$$

As well as the covariance matrix $P = \mathcal{I}^{-1}$.

4.2.2 Active SLAM problem

In this work, we will demonstrate the concepts through trajectory planning for a unicycle robot shown in Figure 2-1, which a platform commonly used in planning literature [76-78]. The robot state vector, $p = [x, y, \theta]^T$, consists of the robot location and heading angle. The robot kinematic model is defined as:

$$\begin{cases} \dot{x} = \vartheta \cos(\theta) \\ \dot{y} = \vartheta \sin(\theta) \\ \dot{\theta} = \omega \end{cases}$$
(4.8)

where the robot control input, $u = [\vartheta, \omega]^T$, consists of linear and angular velocities. In the no-slip condition, the robot kinematics satisfies the mobility constraint:

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0, \ \theta \ge 0 \tag{4.9}$$



Figure 4-1 Schematic of the two-wheeled mobile robot [Figure 2-1 repeated].

The focus of this research is on the discrete-time control of the nonholonomic systems. The discretized equations of motion can be derived from Eq. (4.7) with sampling time *T*:

$$\begin{cases} x^{+} = x + \vartheta(\sin(\theta^{+}) - \sin(\theta))/\omega \\ y^{+} = y - \vartheta(\cos(\theta^{+}) - \cos(\theta))/\omega \\ \theta^{+} = \theta + \omega T \end{cases}$$
(4.10.a)

If $\omega \rightarrow 0$:

$$\begin{cases} x^{+} = x + \vartheta \cos(\theta) T \\ y^{+} = y + \vartheta \sin(\theta) T \\ \theta^{+} = \theta + \omega T \end{cases}$$
(4.10.b)

where x, y and θ are the robot position and heading at sample time k. Additionally, the superscript + denotes the value of the function at sample time k + 1. Considering the disturbance, Eq. (4.10.b) can be expressed as:

$$p^+ = f(p, u, \delta) \tag{4.11}$$

where $\delta \in \mathbb{D} \subset \mathbb{R}^2$ is the disturbance (which is additive to the control input signal), and δ_i belongs to a compact set. A typical autonomous robot is equipped with sensors that can be modeled as a zero-order Markov chain:

$$y = h(p, \mathbb{W}, \nu) \tag{4.12}$$

where ν is probabilistic noise, often considered as a Normal distribution, and \mathbb{W} stands for the world model e.g., a set of landmarks' states. The MPC-based active SLAM can be expressed as the following mathematical programming:

$$\boldsymbol{\mathcal{U}}^* = \operatorname*{argmax}_{\boldsymbol{\mathcal{U}}}(J(I_N)) \tag{4.13.a}$$

subject to:

$$p^+ = f(p, u, \delta) \tag{4.13.b}$$

$$y = h(p, \mathbb{W}, \nu) \tag{4.13.c}$$

$$u \in \mathbb{U}, \delta \in \mathbb{D}, \ \nu \sim \mathcal{N}(0, R) \tag{4.13.d}$$

$$p_i \in \mathbb{P}_{safe} \ i = 1, \dots, N \tag{4.13.e}$$

where $\mathcal{U}^* = \{u_{k+j}\}, j = 0$ to N - 1, is the sequence of optimal control actions over the next N-step horizon. The objective function H can be the information or coverage of the map. The variable I stands for the available information to the algorithm and it is defined as $I_0 = y_0$ and $I_i = \{I_{i-1}, y_i, u_i\}$. In the receding horizon strategy, the first M control actions of \mathcal{U}^* are applied before the new set of the optimal control actions, and are calculated where N and M

are the prediction horizon and control horizon, respectively. The constraints of optimization are expressed in Eqs. (4.13.b) to (4.13.e). In Eq. (4.13.c and d), \mathbb{U} and \mathbb{P}_{safe} are the admissible sets of control actions and safe positions, respectively.

In practice, it is impossible to solve the active SLAM problem in the naïve format of Eq. (4.13). In order to solve the active SLAM, we must first break it into smaller problems in the way that will be discussed in the next section.

4.2.3 Proposed solution overview

Many active SLAM algorithms have been designed based on the feature-based map [60]. The main reason for this choice is that the collection of information can be formulated easily as a constrained optimal control problem. However, the use of a feature-based map in active SLAM has several drawbacks, including:

- The optimization search space is extremely large and impractical to be solved in real time [65].
- A considerable part of the map remains uncovered [60].

For these reasons, the active SLAM problem will be solved using a hybrid map in this research. An occupancy grid map is used to promote the map coverage, and at the same time, a feature-based map is used to enhance the accuracy of localization. In fact, similar to [65, 60], the algorithm will have two steps: exploration, and active localization.

84

Exploration is defined as the ability of the robot to actively map the environment. In fact, the main question in exploration is *where to go next to gain the most feasible information*. A good exploration algorithm is able to map the environment as completely as possible and in a timely manner. To solve the exploration problem, Yamauchi has introduced the *frontierbased* exploration method [98]. Frontiers, as it is shown in Figure 4-2, are boundary regions between open areas and unknown areas. The core idea of frontier-based exploration is that, in order to maximize the information gain, the robot should move to boundaries of the known region. Typically, there is more than one frontier in the map. Hence, the next fundamental question is which frontier, or which point inside the frontier, should be the next target point. In Yamauchi's original algorithm, the closet frontier is selected as the next target. Rocha et al has combined the idea of frontier method [100]. In their approach, frontiers are identified by the maximum gradient of the cell entropy. Also, the exploration strategy is to send the robot to the highest gradient inside the map.



Figure 4-2 A partially explored map; Unknown area is shown in gray, the open area is shown in white, the robot is shown as a circle, and the occupied regions (obstacles) are shown in black. The boundaries between the open area and unknown area, i.e., frontiers, are shown with dashed lines.

In fact, the exploration level is the decision making level in which the next target point of the robot (the attractor point) is determined. Though practically important, this decision making process is not the focus of this section. It is assumed that the where-to-go problem is solved and that the robot's next pose is already determined.

4.2.4 Trajectory Optimization

The accuracy of the map is strongly related to the accuracy of the localization; the higher the accuracy of localization, the higher the accuracy of the map. Specifically, one of the well-known theoretical facts about simultaneous localization and mapping (SLAM) is that the

lower bound of the map accuracy is related to the initial robot pose accuracy [45]. The algorithm at this level tries to optimize the robot trajectory to the target point, which is determined by the map-covering algorithm to reduce the uncertainty of the localization at the same time.

The overall active SLAM algorithm structure is shown in Figure 4-3. In this figure, the exploration level generates the attractor point for the active localization, and the active localization produces an optimal trajectory and enhances the exploration quality. The focus of this chapter is on the formulation of the robust motion planning for the active SLAM. The details of such an algorithm are discussed in the next section.



Figure 4-3 The active SLAM algorithm structure.

4.3 Robust motion planning for active slam Active iSAM

The proposed robust motion planning utilizes a tube-based MPC in which the nominal MPC is called the *active iSAM*. The nominal controller has two aims: first, to reach the target point set by an exploration algorithm, and second, to enhance the localization accuracy. In the active iSAM algorithm, the sensor noise and system disturbance are all approximated with

Gaussian white noise using the heuristics developed in Section 4.2. The Gaussian approximation facilitates the use of the SLAM algorithm, such as iSAM 2.0 that is used in this work. Meanwhile, the set algebra discussed in Section 2.4 is computationally more efficient than integrating a Gaussian distribution over an arbitrary polyhedral set, and the collision avoidance is described using the set algebra. At this stage, differences must be distinguished between $p_{k+j|k}$, which is the robot pose predicted by the overall motion planning scheme, and $z_{k+j|k}$, which denotes the predicted state based on the nominal MPC, such that:

$$\hat{z}_{k+1|k} = f(\hat{z}_{k|k}, \mathfrak{u}_k) \tag{4.14}$$

where u_k is the control action produced by the nominal MPC algorithm. The $z_{k|k}$ and $p_{k|k}$ are identical distributions. As it is discussed in [36], u_k belongs to a subset of \mathbb{U} , $u_k \in \alpha \mathbb{U}$, $0 < \alpha < 1$. Assuming that the nominal state distribution is predicted as $z_{k+j|k} \sim \mathcal{N}(\hat{z}_{k+j|k}, \Sigma_{k+j|k})$, in the set theoretic framework:

$$z_{k+j|k} \in \hat{z}_{k+j|k} \bigoplus \mathbb{S}_{k+j|k} \tag{4.15}$$

where the set $S_{k+j|k}$ corresponds to a 99.73% confidence interval of $s_{k+j|k} \sim \mathcal{N}(0, \Sigma_{k+j|k})$. It should be noted that future measurement is incorporated using the partially closed loop strategy [28], during which active SLAM closes the loop and reduces the uncertainty. Therefore, it can be concluded that $S_{k+j|k} \subseteq S_{k|k}$. The details of uncertainty reduction using the partially closed loop strategy is discussed in prior research [101]. This strategy is used to reduce the computational cost of the allowable set for the nominal MPC optimization. The nominal MPC can be formulated as:

$$\mathfrak{U}^{*} = \underset{\mathfrak{U}}{\operatorname{argmin}} (\sum_{j=0}^{N-1} l_{j} (\hat{z}_{k+j|k} - p_{f}, \mathfrak{u}_{k+j}) + F_{N} (\hat{z}_{k+N|k}, \Sigma_{k+N|k}))$$
(4.16.a)

subject to:

$$\hat{z}^+ = f(\hat{z}, \mathfrak{u}) \tag{4.16.b}$$

$$\mathbf{u}_{k+j} \in \alpha \mathbb{U}, 0 < \alpha < 1 \tag{4.16.c}$$

$$\hat{z}_{k+j|k} \in \mathbb{P}_{safe} \ominus \mathbb{S}_{k|k} \tag{4.16.d}$$

In Eq. (4.16.a), l_j and F_N stand for the stage cost and final cost-to-go, respectively. Eqs. (4.16.c and 4.16.d) tighten the admissible control action and the state sets. The final cost, or cost-to-go, consists of different parts:

$$F_N(\hat{z}_{k+N|k} - p_f, \Sigma_{k+N|k}) = F_{norm}(\hat{z}_{k+N|k} - p_f) + F_{LongTerm}(\hat{z}_{k+N|k} - p_f) + F_{probing}(\Sigma_{k+N|k})$$
(4.17)

The first term $F_{norm}(\hat{z}_{k+N|k} - p_f)$ is an Euclidian norm of the estimated error, $\hat{p}_{k+N|k} - p_f$. The second term is related to the *coarse dynamic* planning cost. Details of Eq. (4.17) is discussed in Section 2.2. The only difference in computation of Eq. 4.17 is related to the probing term.

As discussed earlier, in the smoothing and mapping (SAM) algorithm, the estimation covariance matrix can be estimated using the Jacobian matrix of the incremental solution to the least squares programming, $P = \mathcal{I}^{-1}$, where \mathcal{I} is calculated using Eq. (4.7). The active iSAM algorithm only utilizes the covariance of the robot pose at the step N, $\Sigma_{k+N|k}$. Fast methods to retrieve such a marginal covariance from the information matrix have been discussed [102]. Amongst various ways of measuring the estimation quality, the covariance
matrix trace has been selected as a more consistent measure for estimation quality. Substituting for each cost term, we have:

$$F_N(\hat{z}_{k+N|k}, \Sigma_{k+N|k}) = \left\| \hat{z}_{k+N|k} - p_f \right\|_{W_0}^2 + \gamma_1 l_{shotestpath}^2 + \gamma_2 \operatorname{trace}(\Sigma_{k+N|k})$$
(4.18)

where γ_1 and γ_2 are nonnegative tuning parameters. Also, W_0 is a positive semi-definite matrix.

Since the final cost in Eq. (4.18) requires a closed-form solution to a least-squares programming, it is practically impossible to solve the mathematical programming of Eq. (4.16) over a continuous decision space. Therefore, the control action is considered to be solved over a finite set with the cardinality of \mathcal{T} :

$$\mathfrak{u}_{k+j} \in \{\pi_1, \dots, \pi_T\} \text{ for } j = 1, \dots, N$$
 (4.19)

By assuming u_k to be discrete, the nominal MPC optimization can be solved using dynamic programming:

$$\mathfrak{u}_{k+N-j}^{*} = \operatorname*{argmin}_{\{\mathfrak{u}_{1}, \dots, \mathfrak{u}_{T}\}}(\mathfrak{V}_{j-1} + \min(\mathfrak{V}_{j-2})) \text{ for } j = 2, \dots, N$$
(4.20.a)

subject to:

$$\hat{z}^+ = f(\hat{z}, \mathfrak{u}) \tag{4.20.b}$$

$$\mathbf{u}_{k+j} \in \alpha \mathbb{U}, 0 < \alpha < 1 \tag{4.20.c}$$

$$\hat{z}_{k+j|k} \in \mathbb{P}_{safe} \ominus \mathbb{S}_{k|k} \tag{4.20.d}$$

and

$$\mathfrak{u}_{k+N-1}^{*} = \operatorname*{argmin}_{\{\mathfrak{u}_{1}, \dots, \mathfrak{u}_{T}\}} \mathfrak{V}_{0}$$

$$(4.21)$$

where $\mathfrak{B}_{N-j} = l_j (\hat{z}_{k+j|k} - p_f, \mathfrak{u}_{k+j}) + \mathfrak{B}_{N-j-1}$ and $\mathfrak{B}_0 = F_N (\hat{z}_{k+N|k}, \Sigma_{k+N|k})$. The result of Eq. (35) is the sequence of the planned control action $\mathfrak{U}^* = {\mathfrak{u}_k^* \dots \mathfrak{u}_{k+N-1}^*}$, which produces the optimal nominal trajectory $\mathbf{z}_k^* = {\hat{z}_{k|k}^* \dots \hat{z}_{k+N|k}^*}$. To ensure the robot robust motion around the nominal trajectory, a low-level controller is used. The details of this low-level controller are discussed in the following section.

4.4 Ancillary controller

The ancillary controller increases the robustness of the system around the nominal trajectory. Recalling the fact that the sum of the control actions generated by both nominal and ancillary controllers, i.e. $u_k = u_k + v_k$, we can present the ancillary MPC quadratic programming (QP) as:

$$\boldsymbol{\mathcal{V}}^* = \underset{\boldsymbol{\mathcal{V}}}{\operatorname{argmin}} (\sum_{j=0}^{N-1} \ell_j (\hat{p}_{k+j|k} - \hat{z}_{k+j|k}, \boldsymbol{\upsilon}_{k+j}))$$
(4.22.a)

subject to

$$\hat{p}_{k+j+1|k} = f(\hat{p}_{k+j|k}, \mathfrak{u}_{k+j} + \mathfrak{v}_{k+j})$$
(4.22.b)

$$\boldsymbol{v}_{k+j} \in (1-\alpha)\mathbb{U} \tag{4.22.c}$$

$$\hat{p}_{k+N|k} = \hat{z}_{k+N|k} \tag{4.22.d}$$

where ℓ_j is the stage cost and $\mathcal{V} = \{ v_k \quad \dots \quad v_{k+N-1} \}$ is the ancillary control action sequence. In general, the ancillary controller is designed to be more aggressive than the

nominal controller. The ancillary controller can work at different frequencies. In our implementation, the ancillary controller possesses a frequency 3 times higher than the nominal controller.

4.5 The overall motion planning algorithm

In summary, the proposed robust active SLAM motion planning algorithm can be summarized as:

Step 0. Receive the target point from the exploration algorithm.

Repeat:

- Step 1. Compute the cost map that can be used for long term (if required),
- Step 2. Generate the nominal trajectory using Eq. (4.20) (every *M* sample),
- Step 3. Stabilize around the nominal trajectory using Eq. (4.22),
- Step 4. Add and apply the control signal produced in Step 1 and Step 2

until the robot reaches the target.

If the conditions of the nominal and ancillary controllers are satisfied, then for $k \to \infty$, the covariance trace in the nominal controller is minimized, and thus the estimation quality regarding robot position, p_k , is enhanced. At the same time $||p_f - \hat{p}_k|| \to 0$, which means that the robot converges to the final (desired) value.

4.6 Simulation results

As explained earlier, the main difference of the active iSAM and the active localization algorithm is due to the probing term in Eq. (4.17). In this section, the focus of the simulations

would be on analysis of this term. Two different sets of simulations are carried out. In the first set, the nominal MPC performance for active iSAM is examined. In the second set, the effect of the ancillary controller has been examined.

The simulation map includes a grid of point landmark. The data association is considered to be solved using the unique identifier for each point. The robot is equipped with both bearing and range sensors.

4.6.1 Nominal MPC

As explained previously, the main focus of the numerical simulations is on the probing behavior of the controller. In the absence of the obstacle, the planner behavior is determined by the weight of Euclidian distance versus the probing term.

In the first simulation, the weight of the probing term is assumed to be zero. This simulation provides a benchmark to examine the effect of the probing term. The results of the simulation are shown in Figure 4-4 to 4-5. The robot path and the iSAM estimated robot path are shown in Figure 4-4. The robot control inputs, both translational and rotational velocities, are shown in Figure 4-5. In the second simulation the Euclidian norm is set to zero, and thus the controller cost is only composed of the probing term. The robot path and its estimation, and the control inputs are shown in Figure 4-6 to 4-7. In addition to these extreme cases, the third simulation is set up with weights that incorporate the effect of both Euclidian and probing costs. The robot paths and control input of the third simulations are shown in Figure 4-8 and 4-9, respectively.

The comparison of the planner performance in the three cases reveals the effect of relative weight in the cost function. As expected, in the first simulation, the controller tries to minimize the Euclidian distance as fast as possible. Since the control action is discrete, the robot is not able to reach the target pose. So to maintain the planning cost function at a minimum, it starts loitering around the target point. In the second simulation, the robot minimizes the probing term by demonstrating loop-closing of the loop. This is compatible with the fact that the loop closure reduces the uncertainty in the SLAM process [103]. In the third simulation, the cost function is the aggregation of the two previous cases. Therefore, it is predicted to observe a similar trend in the robot paths; when the Euclidean term is more important, the robot shows exploratory behavior by moving toward the target point. Yet as the probing term becomes more important, the robot benefits from the existing information by the ability to perform the loop closure.

4.6.2 The effect of the ancillary controller

The performance of the ancillary controller has been analyzed using Monte Carlo simulation of the controller performance in presence and absence of the ancillary controller. It is



Figure 4-4 Planning with Euclidean cost function, the robot and the iSAM estimated paths are shown in blue and green, respectively. The true and estimated landmark locations are shown using black and red dots, respectively.



Figure 4-5 The robot control inputs, both translational and rotational velocities, for planning with the Euclidean cost function.



Figure 4-6 Planning with the probing cost function, the robot and the iSAM estimated paths are shown in blue and green, respectively. The true and estimated landmark locations are shown using black and red dots, respectively.



Figure 4-7 The robot control inputs, both translational and rotational velocities for planning with the probing cost function.



Figure 4-8 Planning with both Euclidean and probing cost functions, the robot and the iSAM estimated paths are shown in blue and green, respectively. The true and estimated landmark locations are shown using black and red dots, respectively.



Figure 4-9 the robot control inputs, both translational and rotational velocities when the cost function consists of both probing and Euclidean.

assumed that disturbance is a bivariate noise with covariance equal to 5% of the control input. The robot paths obtained without and with the ancillary controller are shown in Figure 4-10 and 4-11, respectively. A qualitative comparison of these two figures indicates that the ancillary controller has successfully contributed to the robustness of the controller performance. The disturbance has a larger influence on the rotational motion of the robot, which is in agreement with the experimental results of Chapter 2.



Figure 4-10 The Monte Carlo simulation of the active iSAM controller without the ancillary controller.



Figure 4-11 The Monte Carlo simulation of the active iSAM controller with the ancillary controller.

4.7 Summary

In this chapter, a new active SLAM method, named active iSAM, has been introduced. The proposed method is based on the optimization-based SLAM method, or incremental smoothing and mapping, iSAM 2.0. The objective function of the control scheme is composed of various terms including probing, Euclidian, and long term cost-to-go that can be tuned to achieve the best possible performance of the system depending on the application in hand. In this work, the optimization is solved using the dynamic programming method to meet the computational load expected in real-time applications.

This research takes a close look at the effect of the probing and Euclidian terms, as well as their combined effect when the two costs are aggregated. Based on numerical simulations, the control scheme with only Euclidian cost function tends to move the robot toward the target point in the shortest path. As the robot approaches the target point, it begins to loiter around the target point, in order to maintain the minimum possible cost function. The probing term, on the other hand, causes the robot to reduce the uncertainty by loop closure. When the planner uses the aggregated cost function, the robot path shows a behavior that is attributed partially to the Euclidian cost function and partially to the probing term. This degree of the similarity to each of these cases depends on which term, Euclidian or probing, is more significant from the optimization point of view.

In addition to introducing the idea of active iSAM, a tube-based scheme has been used to robustify the planning scheme. Using an ancillary controller, the control response has become more robust and less sensitive to the external disturbances.

Chapter 5. Conclusions and Future Work

The main objective of this thesis is to develop and analyze model predictive control based motion planning and control scheme for uncertain environments. Every chapter of this thesis deals with one important aspect of this topic. In this chapter, a summary of conclusions and future work of each chapter is presented.

5.1 Conclusions

Chapter 2 presents a tube-based active localization planning scheme. The proposed scheme has two components: a nominal MPC planner and an ancillary MPC controller. To guarantee that the controller is both robust and probing, two different frameworks, stochastic and setbased, have been used to model the uncertainties involved. The stochastic framework has been used to incorporate future measurements and maintain the controller probing behavior. On the other hand, robustness of the planning algorithm is guaranteed using the tube-based set theoretic framework. Chapter 2 focuses on unifying these two frameworks in a systematic way rather than ad-hoc heuristics. Extensive numerical simulations and experiments with a mobile robot have been carried out to demonstrate the functionality of the components of the proposed algorithm and examine its performance.

Based on the simulation results, it can be concluded that global performance of the algorithm is closely related to the coarse dynamic term, whereas the norm cost affects the steady-state error and the planned trajectory smoothness. Also, it has been shown that in systems with a highly nonlinear observation model, the probing term can contribute to the informationseeking performance of the controller. However, the probing term effect reduces as the measurement model becomes linear. Using Monte Carlo simulations, the impact of the ancillary controller was proven to be successful in increasing the nominal MPS robustness. To gain more insight, the algorithm performance was studied in a dynamic cluttered environment. Finally, the algorithm performance was examined using the UBC ACIS testing platform. The experiment results show that while the motion planning can be implemented and drive the robot to a target point successfully, the algorithm performance deteriorate as the low-level system dynamics becomes nontrivial.

Chapter 3 introduces an unscented MPC (UMPC) algorithm. In this algorithm, the state transition is computed using the unscented transform instead of the conventional Taylor expansion. The UMPC approach is analyzed in the application of the motion control problem of a unicycle in unstructured environments. The UMPC is able to incorporate both deterministic and stochastic chance constraints. These include motion constraints, collision avoidance chance constraints, and control action limitations. A closed-form solution for the chance constraints is calculated using the unscented transform approximation. In three different scenarios, including: i) a static environment with no obstacle, ii) a static environment with a moving obstacle, the UMPC based motion control algorithm is compared to the conventional MPC (CMPC) based algorithm.

As numerical simulations indicate, the proposed UMPC offers several important advantages over CMPC. First, as a derivative free algorithm, the UMPC can be easily implemented to

102

complicated systems has no analytical models. Second, by providing more accurate approximation, an overly conservative planning can be avoided. Finally, in the case of motion planning example, it has been shown that the proposed approach will result in a lower energy consumption. However, CMPC is computationally less expensive and still a better choice to reduce the computational load.

Chapter 4 focuses on a novel the active iSAM algorithm of a mobile robot based on an MPCbased algorithm. The proposed method utilizes the iSAM 2.0, an incremental optimizationbased SLAM algorithm, as the prediction algorithm. The objective function consists of the probing, Euclidian, and the cost-to-go terms. This thesis presents an in-depth analysis of the aggregation of the probing and Euclidian terms. As it is predicted, the numerical simulation indicated that the robot moves toward the shortest path when using a Euclidean objective function. As the robot reaches its minimum distance around the target point, it starts to loiter in position to maintain the minimum possible cost function. On the other hand, the controller uses the result of the optimization based on the objective function consisting of the probing term alone. In the meantime, the algorithm tries to minimize the robot localization uncertainty attributed to loop closure. When the objective function is composed of both Euclidian and probing terms, the controller response may be similar to the controller with either Euclidian objective function or the controller with the probing term. The relative influence of each term, specified by their weight, determines the combined controller response.

103

The proposed active iSAM approach has been implemented as a tube-based scheme. As a result of the use of ancillary controller, the system robustness against external disturbance is improved.

5.2 Future work

In **Chapter 2**, it has been demonstrated that the unmodeled low-level dynamics deteriorates the performance of the proposed integrated motion planning and control algorithm. How to properly handle the unmodeled low-level dynamic could be the topic of further studies. One suggestion would be to replace the current system model with a detailed model that is able to capture the low-level dynamics better. Another possible solution would be to use a detailed model only for the ancillary MPC. The advantages of the latter solution is that no additional complexity will be added to the Nominal MPC. However, there could be a feasibility discrepancy between the nominal and ancillary MPCs which should be discussed in the future studies.

In **Chapter 3**, the idea of UMPC was introduced, but there remain questions regarding the efficiency of the real-time implementation of the UMPC for practical applications. As it has been shown in Chapter 3, UMPC is not computationally as efficient as CMPC. Hence, one possible direction for the future studies would be to increase the computational efficiency of the UMPC algorithm. The unscented transform has a great potential for parallel implementation. Therefore, algorithmic optimization of the UMPC to take advantage of the parallel implementation is the starting point to increase the UMPC efficiency. Another practical issue regarding UMPC is the sampling weights of the unscented transform. To

implement UMPC successfully, these weights must be able to capture the underlying distribution properties. In our implementation, these weights have to be determined by trial and error, or an independent statistical analysis of the disturbance and noise. This issue will limit the practicality of UMPC. Therefore, another possible research direction is to develop an adaptive UMPC that requires limited a priori knowledge about the process disturbance and noise.

Chapter 4 presented the basis of the active iSAM algorithm. However, there are many aspects of this algorithm left for future studies. The experimental study of the active iSAM will provide more insight to the algorithm performance and limitations. A basic assumption of this chapter is that the data association required for the SLAM algorithm never fails. In practice, data association is not a trivial problem. One possible future direction to increase the robustness of the SLAM algorithm is to investigate on the relation of the trajectory planning and performance of the data association algorithm. Another possible future direction is to investigate on the optimization structure of the active iSAM algorithm to optimize the computational load of the algorithm. At last, and from practical point of view, the combination of the active iSAM algorithm with a frontier based exploration algorithm is another important topic for future work.

105

References

[1] M. Farrokhsiar, G. Pavlik and H. Najjaran, "An integrated robust probing motion planning and control scheme: A tube-based MPC approach," *Robotics and Autonomous Systems*, 2013.

[2] M. Farrokhsiar and H. Najjaran, "An unscented predictive control approach to formation control of multiple nonholonomic mobile robots," in *2012 IEEE International Conference on Robotics and Automation (ICRA 2012)*, pp. 1576-1582.

[3] M. Farrokhsiar and H. Najjaran, "Unscented predictive motion planning of a nonholonomic system," in 2011 IEEE International Conference on Robotics and Automation (ICRA 2011), 2011, .

[4] M. Farrokhsiar and H. Najjaran, "Analysis of future measurement incorporation into unscented predictive motion planning," in *Autonomous and Intelligent Systems*, M. Kamel and and Others., Eds. Springer, 2011, pp. 112-123.

[5] M. Farrokhsiar and H. Najjaran, "Unscented model predictive control of chance constrained nonlinear systems," *Advanced Robotics*, vol. 28, pp. 257-267, 2014.

[6] J. Yuh, "Design and control of autonomous underwater robots: A survey." *Autonomous Robots*, vol. 8, pp. 7-24, 2000.

[7] R. M. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control,* vol. 129, pp. 571, 2007.

[8] Y. Liu and G. Nejat, "Robotic urban search and rescue: A survey from the control perspective ," *Journal of Intelligent & Robotic Systems*, vol. 72, pp. 147-165, 2013.

[9] N. Roy, G. Gordon and S. Thrun, "Planning under uncertainty for reliable health care robotics," in *Field and Service Robotics*, 2006, pp. 417-426.

[10] S. Tadokoro, M. Hayashi, Y. Manabe, Y. Nakami and T. Takamori, "On motion planning of mobile robots which coexist and cooperate with human," in *Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference On,* 1995, pp. 518-523 vol.2.

[11] S. M. LaValle, *Planning Algorithms*. Cambridge Univ Pr, 2006.

[12] H. M. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki and S. Thrun, *Principles of Robot Motion*. MIT press, 2005.

[13] K. Yang, S. K. Gan and S. Sukkarieh, "An Efficient Path Planning and Control Algorithm for RUAV's in Unknown and Cluttered Environments," *Journal of Intelligent and Robotic Systems,* vol. 57, pp. 101-122, 2010.

[14] G. C. Goodwin, M. Seron and J. De Dona, *Constrained Control and Estimation: An Optimisation Approach.* Springer Verlag, 2005.

[15] Y. Kuwata, "Trajectory planning for unmanned vehicles using robust receding horizon control," 2007.

[16] P. Li, M. Wendt and G. Wozny, "A probabilistically constrained model predictive controller," *Automatica*, vol. 38, pp. 1171-1176, 7, 2002.

[17] A. Burlacu and C. Lazar, "Reference Trajectory-Based Visual Predictive Control," *Adv. Rob.*, vol. 26, pp. 1035-1054, 2012.

[18] D. Q. Mayne, J. B. Rawlings, C. V. Rao and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789-814, 6, 2000.

[19] M. Morari and H. Lee, "Model predictive control: past, present and future," *Comput. Chem. Eng.*, vol. 23, pp. 667-682, 1999.

[20] M. Hoy, A. S. Matveev and A. V. Savkin, "Collision free cooperative navigation of multiple wheeled robots in unknown cluttered environments," *Robotics and Autonomous Systems*, 2012.

[21] C. Goerzen, Z. Kong and B. Mettler, "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance," *Journal of Intelligent and Robotic Systems,* vol. 57, pp. 65-100, 2010.

[22] Y. Kuwata, A. Richards, T. Schouwenaars and J. P. How, "Distributed robust receding horizon control for multivehicle guidance," *Control Systems Technology, IEEE Transactions On,* vol. 15, pp. 627-641, 2007.

[23] D. Li, F. Qian and P. Fu, "Optimal nominal dual control for discrete-time linear-quadratic Gaussian problems with unknown parameters," *Automatica*, vol. 44, pp. 119-127, 1, 2008.

[24] J. Yan and R. R. Bitmead, "Incorporating state estimation into model predictive control and its application to network traffic control," *Automatica*, vol. 41, pp. 595-604, 4, 2005.

[25] D. Van Hessem and O. Bosgra, "Stochastic closed-loop model predictive control of continuous nonlinear chemical processes," *J. Process Control,* vol. 16, pp. 225-241, 2006.

[26] L. Blackmore, "A probabilistic particle control approach to optimal, robust predictive control," 2006.

[27] M. Ono and B. C. Williams, "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference On,* 2008, pp. 3427-3432.

[28] N. E. Du Toit, "Robot Motion Planning in Dynamic, Cluttered, and Uncertain Environments: the Partially Closed-Loop Receding Horizon Control Approach," 2010.

[29] M. P. Vitus and C. J. Tomlin, "Closed-loop belief space planning for linear, gaussian systems," in *Proceedings of the International Conference on Robotics and Automation (ICRA),* 2011, .

[30] D. H. Shim, H. Chung and S. S. Sastry, "Conflict-free navigation in unknown urban environments," *IEEE Robotics & Automation Magazine*, pp. 28, 2006.

[31] J. Wan, C. G. Quintero, N. Luo and J. Vehi, "Predictive motion control of a mirosot mobile robot," in *Automation Congress, 2004. Proceedings. World,* 2004, pp. 325-330.

[32] Y. Zhu and U. Ozguner, "Constrained model predictive control for nonholonomic vehicle regulation problem," in *Proceedings of the 17th IFAC World Congress*, 2008, pp. 9552–9557.

[33] G. Grimm, M. J. Messina, S. E. Tuna and A. R. Teel, "Nominally robust model predictive control with state constraints," *IEEE Transactions on Automatic Control*, vol. 52, pp. 1856-1870, 2007.

[34] D. Gu and H. Hu, "Receding horizon tracking control of wheeled mobile robots," *IEEE Trans. Control Syst. Technol.*, vol. 14, pp. 743-749, 2006.

[35] H. Michalska and D. Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 38, pp. 1623-1633, 1993.

[36] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Nob Hills, 2009.

[37] P. Scokaert and D. Mayne, "Min-max feedback model predictive control for constrained linear systems," *Automatic Control, IEEE Transactions On,* vol. 43, pp. 1136-1142, 1998.

[38] A. Bemporad and M. Morari, "Robust model predictive control: A survey," *Robustness in Identification and Control,* pp. 207-226, 1999.

[39] R. González, M. Fiacchini, J. L. Guzmán, T. Álamo and F. Rodríguez, "Robust tube-based predictive control for mobile robots in off-road conditions," *Robotics and Autonomous Systems*, 2011.

[40] D. Q. Mayne, E. C. Kerrigan and P. Falugi, "Robust model predictive control: Advantages and disadvantages of tube-based methods," in *World Congress*, 2011, pp. 191-196.

[41] D. Mayne, S. Raković, R. Findeisen and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems," *Automatica*, vol. 42, pp. 1217-1222, 2006.

[42] M. Cannon, B. Kouvaritakis, S. V. Rakovic and Q. Cheng, "Stochastic tubes in model predictive control with probabilistic constraints," *Automatic Control, IEEE Transactions On*, vol. 56, pp. 194-200, 2011.

[43] D. Q. Mayne, E. C. Kerrigan, E. van Wyk and P. Falugi, "Tube-based robust nonlinear model predictive control," *International Journal of Robust and Nonlinear Control,* 2011.

[44] R. Smith, M. Self and P. Cheeseman, "A stochastic map for uncertain spatial relationships," in *The Fourth International Symposium of Robotics Research*, 1988, pp. 467-474.

[45] M. W. M. G. Dissanayake, "A solution to the simultaneous localization and map building (SLAM) problem," *Robotics and Automation, IEEE Transactions On,* vol. 17, pp. 229-241, 2001.

[46] M. Montemerlo, S. Thrun, D. Koller and B. Wegbreit, "FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *International Joint Conference on Artificial Intelligence*, 2003, pp. 1151-1156.

[47] G. Dissanayake, S. Huang, Z. Wang and R. Ranasinghe, "A review of recent developments in simultaneous localization and mapping," in *Industrial and Information Systems (ICIIS), 2011 6th IEEE International Conference On,* 2011, pp. 477-482.

[48] S. Thrun, "Learning occupancy grids with forward sensor models," *Autonomous Robots*, vol. 15, pp. 111–127, 2003.

[49] L. Moreno, S. Garrido, D. Blanco and M. L. Muñoz, "Differential evolution solution to the SLAM problem," *Robotics and Autonomous Systems*, vol. 57, pp. 441-450, 4/30, 2009.

[50] H. F. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, pp. 99-110, 2006.

[51] H. Strasdat, J. Montiel and A. J. Davison, "Real-time monocular slam: Why filter?" in *Robotics and Automation (ICRA), 2010 IEEE International Conference On,* 2010, pp. 2657-2664.

[52] M. Kaess, A. Ranganathan and F. Dellaert, "iSAM: Incremental smoothing and mapping," *Robotics, IEEE Transactions On*, vol. 24, pp. 1365-1378, 2008.

[53] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31, pp. 216-235, 2012.

[54] N. M. Filatov and H. Unbehauen, "Survey of adaptive dual control methods," *Control Theory and Applications, IEE Proceedings -,* vol. 147, pp. 118-128, 2000.

[55] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 865-880, 2002.

[56] R. Sim and N. Roy, "Global a-optimal robot exploration in slam," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference On,* 2005, pp. 661-666.

[57] S. Huang, N. M. Kwok, G. Dissanayake, Q. P. Ha and G. Fang, "Multi-step look-ahead trajectory planning in SLAM: Possibility and necessity," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference On,* 2005, pp. 1091-1096.

[58] G. Fang, G. Dissanayake, N. Kwok and S. Huang, "Near minimum time path planning for bearing-only localisation and mapping," in *Intelligent Robots and Systems, 2005. (IROS 2005).* 2005 IEEE/RSJ International Conference On, 2005, pp. 850-855.

[59] C. Leung, S. Huang and G. Dissanayake, "Active SLAM in structured environments," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference On,* 2008, pp. 1898-1903.

[60] C. Leung, S. Huang and G. Dissanayake, "Active SLAM using model predictive control and attractor based exploration," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, .

[61] J. Le Ny and G. J. Pappas, "On trajectory optimization for active sensing in gaussian process models," in *Decision and Control, 2009 Held Jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference On,* 2009, pp. 6286-6292.

[62] C. Stachniss, *Robotic Mapping and Exploration*. Springer Verlag, 2009.

[63] R. Martinez-Cantin, N. de Freitas, A. Doucet and J. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," in *Proc. of Robotics: Science and Systems*, 2007, .

[64] R. Martinez-Cantin, N. de Freitas, E. Brochu, J. Castellanos and A. Doucet, "A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot," *Autonomous Robots*, vol. 27, pp. 93-103, 2009.

[65] T. Kollar and N. Roy, "Trajectory optimization using reinforcement learning for map exploration," *The International Journal of Robotics Research*, vol. 27, pp. 175, 2008.

[66] J. Urrutia, "Art gallery and illumination problems," *Handbook of Computational Geometry*, pp. 973-1027, 2000.

[67] A. Ganguli, J. Cortes and F. Bullo, "Distributed deployment of asynchronous guards in art galleries," in *American Control Conference, 2006,* 2006, pp. 6 pp.

[68] H. Carrillo, I. Reid and J. A. Castellanos, "On the comparison of uncertainty criteria for active SLAM," in *Robotics and Automation (ICRA), 2012 IEEE International Conference On,* 2012, pp. 2080-2087.

[69] R. Van Der Merwe and E. Wan, "The square-root unscented kalman filter for state and parameter-estimation," in *IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS SPEECH AND SIGNAL PROCESSING*, 2001, pp. 3461-3464.

[70] M. Reble and F. Allgöwer, "Unconstrained model predictive control and suboptimality estimates for nonlinear continuous-time systems," *In Automatica*, 2011.

[71] A. Richards and J. P. How, "Robust distributed model predictive control," *Int J Control*, vol. 80, pp. 1517-1531, 2007.

[72] Y. Bar-Shalom, "Stochastic dynamic programming: Caution and probing," *Automatic Control, IEEE Transactions On,* vol. 26, pp. 1184-1195, 1981.

[73] A. Aswani, H. Gonzalez, S. S. Sastry and C. Tomlin, "Provably safe and robust learningbased model predictive control," *Arxiv Preprint*, 2011.

[74] Z. Artstein and S. V. Raković, "Set invariance under output feedback: a set-dynamics approach," *Int. J. Syst. Sci.*, vol. 42, pp. 539-555, 2011.

[75] N. E. Du Toit and J. W. Burdick, "Robot Motion Planning in Dynamic, Uncertain Environments," *Robotics, IEEE Transactions On*, pp. 1-15, 2012.

[76] C. Belta, V. Isler and G. J. Pappas, "Discrete abstractions for robot motion planning and control in polygonal environments," *Robotics, IEEE Transactions On,* vol. 21, pp. 864-874, 2005.

[77] G. Conte, S. Longhi and R. Zulli, "Motion planning for unicycle and car-like robots," *Int. J. Syst. Sci.*, vol. 27, pp. 791-798, 1996.

[78] M. Egerstedt, X. Hu, H. Rehbinder and A. Stotsky, "Path planning and robust tracking for a car-like robot," in *Proceedings of the 5th SympoREFERENCES 29 Sium on Intelligent Robotic Systems*, 1997, .

[79] R. W. Brockett, "Asymptotic stability and feedback stabilization," in *Differential Geometric Control Theory*. Birkhauser 1983, pp 181-191.

[80] Y. Zhu and U. Ozguner, "Robustness analysis on constrained model predictive control for nonholonomic vehicle regulation," in *American Control Conference, 2009. ACC'09.* 2009, pp. 3896-3901.

[81] S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics*. MIT press, Cambridge, Massachusetts, USA, 2006.

[82] F. Blanchini and S. Miani, Set-Theoretic Methods in Control. Springer, 2008.

[83] T. Bailey, "Consistency of the EKF-SLAM algorithm," in 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 3562.

[84] I. Kolmanovsky, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical Problems in Engineering*, vol. 4, pp. 317, 1998.

[85] J. M. M. Montiel, J. Civera and A. J. Davison, "Unified inverse depth parametrization for monocular SLAM," in *Proc. Robotics: Science and Systems*, 2006, .

[86] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*. Prentice hall Upper Saddle River, NJ, 2002.

[87] M. Hovd and R. R. Bitmead, "Interaction between control and state estimation in nonlinear MPC," Dynamics and Control of Process Systems, pp. 119, 2004.

[88] Huq, Rajibul et al, " Qbot: An educational mobile robot controlled in MATLAB Simulink environment," in *Proc of 2010 Canadian Conference on Electrical and Computer Engineering,* 2010.

[89] A. Richards and J. How, "Robust distributed model predictive control," *Int J Control,* vol. 80, pp. 1517-1531, 2007.

[90] Y. Kuwata, T. Schouwenaars, A. Richards and J. How, "Robust constrained receding horizon control for trajectory planning," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* 2005, .

[91] N. E. Du Toit and J. W. Burdick, "Probabilistic collision checking with chance constraints," *Robotics, IEEE Transactions On,* vol. 27, pp. 809-815, 2011.

[92] Y. Wu, D. Hu, M. Wu and X. Hu, "Unscented Kalman filtering for additive noise case: augmented versus nonaugmented," *Signal Processing Letters, IEEE*, vol. 12, pp. 357-360, 2005.

[93] Pioneer LX Research Platform, http://www.mobilerobots.com/ResearchRobots/ PioneerLX.aspx. [94] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT press, 2009.

[95] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal and C. Sagastizabal, *Numerical Optimization*. Springer, 2003.

[96] K. Konolige, G. Grisetti, R. Kummerle, W. Burgard, B. Limketkai and R. Vincent, "Efficient sparse pose adjustment for 2d mapping," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference On,* 2010, pp. 22-29.

[97] F. Dellaert and M. Kaess, "Square Root SAM: Simultaneous localization and mapping via square root information smoothing," *The International Journal of Robotics Research*, vol. 25, pp. 1181-1203, 2006.

[98] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium On,* 1997, pp. 146-151.

[99] W. Burgard, D. Fox and S. Thrun, "Active mobile robot localization," in *INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE*, 1997, pp. 1346-1352.

[100] R. Rocha, J. Dias and A. Carvalho, "Cooperative multi-robot systems::: A study of visionbased 3-D mapping using information theory," *Robotics and Autonomous Systems*, vol. 53, pp. 282-311, 2005.

[101] N. E. Du Toit and J. W. Burdick, "Robotic motion planning in dynamic, cluttered, uncertain environments," in *Robotics and Automation (ICRA), 2010 IEEE International Conference On,* 2010, pp. 966-973.

[102] M. Kaess, A. Ranganathan and F. Dellaert, "iSAM: Incremental smoothing and mapping," *Robotics, IEEE Transactions On*, vol. 24, pp. 1365-1378, 2008.

[103] C. Stachniss, G. Grisetti, D. Hähnel and W. Burgard, "Improved rao-blackwellized mapping by adaptive sampling and active loop-closure," in *In Proc. of the Workshop on Self-Organization of AdaptiVE Behavior (SOAVE)*, 2004, .