UNIVERSITY OF MINNESOTA

This is to certify that I have examined this copy of a doctoral dissertation by

ANASTASIOS MOURIKIS

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

STERGIOS ROUMELIOTIS

Name of Faculty Adviser

Signature of Faculty Adviser

Date

GRADUATE SCHOOL

CHARACTERIZATION AND OPTIMIZATION OF THE ACCURACY OF MOBILE ROBOT LOCALIZATION

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

ANASTASIOS MOURIKIS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

STERGIOS ROUMELIOTIS, ADVISOR

JUNE 2008

Acknowledgements

This work would never have been possible without the help and support of many people, to whom I am deeply grateful. First, I would like to thank my parents, for being beside me all these years, even though they live so far away. I have devoted five years of my life to my doctoral studies, but my parents have devoted many more years of their lives, to give me the opportunity to be here today. Without their guidance, I would not have made it.

My advisor, Stergios Roumeliotis has spared no effort in providing me with support, help, and mentoring. For the past five years he has been an *advisor*, in every possible sense of the word. Not just in academic matters, but also during difficult times in my graduate life. He has created an incredible learning environment in our lab, and has worked tirelessly to provide us with great research and career opportunities. I am grateful for all his efforts.

I would also like to thank the members of my committee, Professor Georgios Giannakis, Professor Yousef Saad and Professor Paul Schrater, for their time and valuable advice.

I owe a lot to my friends, relatives and labmates, who have helped make my time in Minnesota a great experience. My current and former labmates, Joel, Paul, Faraz, Esha, Thor-Andreas, Kyle, Nikolas, Ke and Sam, have offered me their help every time I needed it, and have made our office a great place to work. Alki, Vassilis ($\times 2$), Nikos ($\times 3$), Yiannis ($\times N$), Dimitris, Elissaios, George, Ioanna, Zafi, Bree, Vasia, Triantafyllos, Sofia, Spyros (and others, who I am surely forgetting here) have helped me maintain my sanity, through the often insane times of my PhD.

Finally, financial support for my work came from the National Science Foundation, the NASA Jet Propulsion Laboratory, the Digital Technology Center at University of Minnesota, and the University of Minnesota Doctoral Dissertation Fellowship.

Abstract

Robot localization is the task of processing sensor measurements to estimate a robot's position and orientation in real time. Accurate localization is a prerequisite for robot autonomy, since, without knowledge of its location, a vehicle cannot navigate effectively in its environment. Despite the substantial research interest in robot localization, the question of analytical performance evaluation has received very little attention in the literature, and no theoretical tools exist for predicting the localization accuracy a given system will attain. The lack of such tools also hinders the development of methods for maximizing the attainable accuracy, under constraints imposed by the, inevitably limited, system resources. In this work, we address these limitations by proposing methods for analytically characterizing and optimizing the accuracy of robot localization.

In particular, the first problem we address is that of providing performance guarantees for localization. We focus on three classes of localization algorithms: Cooperative Localization (CL), Simultaneous Localization and Mapping (SLAM), and Cooperative SLAM (C-SLAM). In each of these cases, we derive analytical expressions that determine upper bounds on the localization uncertainty, as a function of the relevant system parameters: (i) the accuracy of all available sensors, (ii) the number of robots and/or landmarks, (iii) the spatial distribution of the robots and/or landmarks, and (iv) the structure of the graph describing the robot-to-robot and robot-to-landmark measurements. The derived closed-form expressions allow us to study the asymptotic properties of localization, and examine the impact of several factors on the localization accuracy.

The second problem we address is that of optimal resource utilization during CL in robot formations. We propose a methodology for selecting the optimal set of measurements to process at each time-step, so as to attain the highest possible localization accuracy given the

robots' limited computational and communication resources. Our approach is based on first expressing the localization accuracy as a function of the rate at which each of the available sensors is utilized, and then formulating a convex optimization problem, to determine the optimal rate for each of the sensors. Due to the convex nature of the optimization problem, a globally optimal solution is computed using efficient minimization routines.

Finally, the third problem we focus on is that of optimal resource allocation during vision-based localization. One of the key challenges in this domain is the very large number of available feature measurements, which can overwhelm the limited computational capabilities of a real-time system. Our approach to this problem is based on the observation that the vast majority of features can only be tracked in a small number of frames (transient features). We thus propose an algorithm that can optimally process the measurements of such features, with computational complexity only linear in their number. We then show how this algorithm can be incorporated into a two-layer localization system that utilizes the information of both the transient features, as well as the loop-closure information that becomes available when a robot re-visits an area. The combination of these two types of positioning information produces pose estimates that are available in real time, and have bounded long-term errors.

By providing performance guarantees and methods for the optimal allocation of the available system resources, the research presented in this dissertation aims at increasing the *reliability* and *cost efficiency* of robot designs, both of which are prerequisites for their widespread use and commercial success.

Contents

List of Tables

List of Figures

Nomenclature

$\widehat{x}$	The estimate of a variable x
$\widetilde{x}$	The error in the estimate $\widehat{x}$, defined as $\widetilde{x} = x - \widehat{x}$
$\widehat{x}_{\ell\|j}$	EKF estimate of the state at time-step ℓ, given all measurements up to time-step j
$\mathbf{I}_n$	The $n \times n$ identity matrix
$\mathbf{1}_{n\times m}$	The $n \times m$ matrix of ones
$\mathbf{0}_{n\times m}$	The $n \times m$ matrix of zeros
$\text{diag}(\cdot)$	Diagonal matrix
$\mathbf{Diag}(\cdot)$	Block diagonal matrix
$\otimes$	The Kronecker product for matrices
$\mathbf{A} \succeq \mathbf{B}$	Matrix inequality in the positive semidefinite sense, i.e., $\mathbf{A} \succeq \mathbf{B}$ implies $\mathbf{A} - \mathbf{B}$ is positive semidefinite
$\odot$	Quaternion multiplication
$\lfloor \mathbf{x} \times \rfloor$	Skew-symmetric matrix corresponding to vector $\mathbf{x}$

Chapter 1

Introduction

1.1 Robot localization

The work presented in this thesis describes methods for characterizing and improving the accuracy of mobile robot localization. Localization, i.e., the task of using sensor measurements to estimate the robot position and orientation in real time, is a fundamental problem in mobile robotics. Without knowledge of its location, a robot cannot navigate and interact with its environment in a meaningful way. It should be pointed out that animals and humans themselves possess impressive navigational skills. Through millions of years of evolution, species have developed mechanisms enabling them to determine their location by processing visual information [114, 135], measurements of electric [17] and magnetic [24] fields, echolocation [146], and other sensory input. The ultimate goal of localization methods in the field of robotics is to endow robots with similar capabilities, which are a prerequisite for robots to be able to operate autonomously.

The sensors that robots employ for localization can be broadly divided in two categories: *proprioceptive* and *exteroceptive* sensors. Sensors in the first category provide measurements of quantities related to motion, such as velocity and acceleration, and are the most commonly available sensors on mobile robots. Typical examples of proprioceptive sensors are wheel encoders (odometry sensors) [72], which provide measurements of linear and rotational velocity for ground vehicles, and Inertial Measurement Units (IMUs) [160], which measure rotational velocities and linear accelerations. On the other hand, exteroceptive

sensors measure quantities of interest in the robot's surroundings, such as the range and bearing to a nearby object. Examples of exteroceptive sensors typically employed in robotics applications include cameras [28], sonars [82], laser range finders [85], and GPS receivers [158].

Due to disturbances, modeling errors, and other factors, the sensor measurements available to a robot are inevitably corrupted by *noise*. The presence of noise implies that the true pose (i.e., position and orientation) of a robot cannot, in general, be estimated with zero error; some *uncertainty* will always exist about the robot pose. Thus, the goal of a localization algorithm is to fuse the available proprioceptive and exteroceptive measurements in real time, to compute an *estimate* for the robot pose, as well as a measure of the uncertainty of this estimate. Depending on the sensors available onboard a robot, several localization approaches are possible [156]. In the following, we briefly discuss some of the approaches that will be addressed in this work.

- **Dead Reckoning (DR)** is the simplest form of localization. It consists of simply propagating the pose estimates of a mobile robot by integrating the measurements of its proprioceptive sensors. The limitation of this approach, however, is that due to the integration of the measurement noise, the pose estimates quickly drift from their true values, and eventually become unusable (cf. Fig. 1.1) [72, 107]. Therefore, for most practical purposes, additional information from exteroceptive sensors must be processed, in order to reduce or even eliminate the growth of DR errors.

- **GPS-based localization**: Vehicles moving outdoors can obtain measurements of their *absolute position*, by measuring range to the GPS satellites [158]. When this is possible, the localization problem is simplified significantly, because the availability of direct measurements of the robot's position prevents drift, and maintains bounded position errors over time (cf. Fig. 1.2). In many environments of interest, however, GPS signals are unavailable or unreliable (e.g., indoors, underwater, on other planets, in the "urban canyon"). Additionally, the accuracy of pose estimates based on GPS may be insufficient for some applications (e.g., autonomous navigation in traffic). Therefore, it is important to develop methods for localization in the absence of (or in addition to) GPS. Two such approaches are presented next.

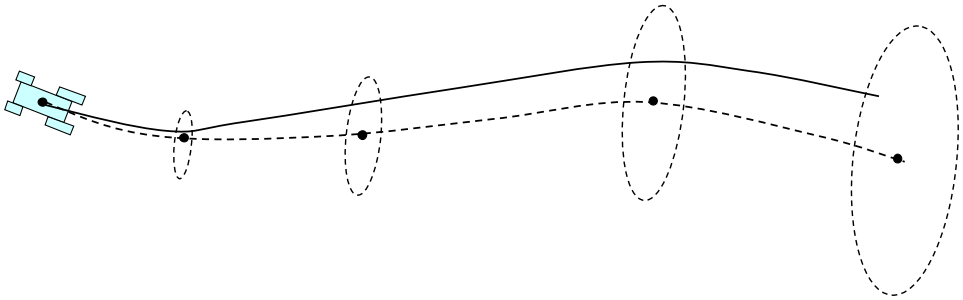


Figure 1.1: DR illustration. The solid line represents the true robot trajectory, while the dashed line represents the estimated one. The ellipses represent the uncertainty of the position estimates (they are the estimates' 99.7% confidence regions). During DR, the uncertainty of the estimates continuously increases, asymptotically approaching infinity [72]. The growth rate of the errors is determined by the accuracy of the robot's proprioceptive sensors.

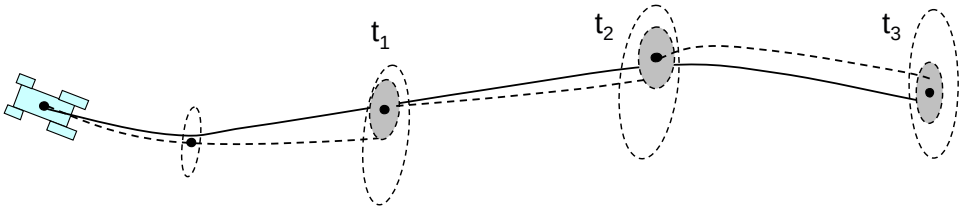


Figure 1.2: Illustration of GPS-based localization. At time instants t_1, t_2, and t_3, GPS measurements are recorded, and used for updating the robot's pose estimate. Every time an absolute position measurement is processed, the uncertainty of the resulting estimate is reduced (shaded ellipses). If the robot proceeds by periodically using GPS measurements for position updates, drift is prevented, and at steady state the uncertainty of the pose estimates is bounded (cf. Section 2.4).

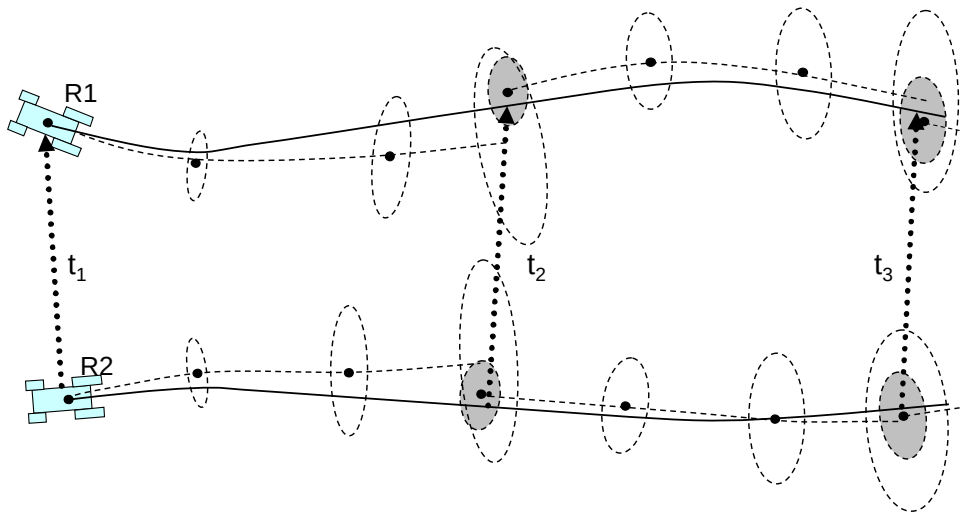
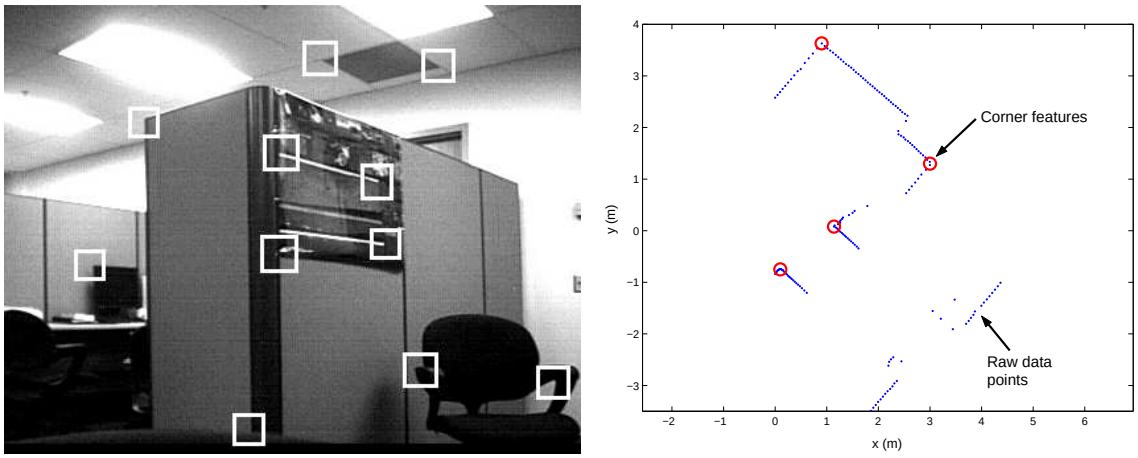


Figure 1.3: Illustration of CL: At time instants t_1, t_2, and t_3, Robot 2 records measurements of the relative position of Robot 1 (denoted by dotted arrows). Using these measurements, both robots' pose estimates are updated, and the resulting estimates have reduced uncertainty. However, in absence of any absolute position information, the relative measurements cannot fully compensate for position drift. Therefore, the position uncertainty gradually increases over time, albeit at a rate lower than that of DR (cf. Section 2.4).

- **Cooperative Localization (CL)**: In many cases, a team of robots is required to cooperate in order to accomplish a certain task (e.g., object transportation [170], construction [127], exploration [16], etc). When multiple autonomous vehicles operate in the same area, one attractive approach for aiding the localization of all robots is for them to record relative (i.e., robot-to-robot) measurements, such as distance and bearing, and to combine this information with their odometric estimates (cf. Fig. 1.3). When robots record, communicate, and process such relative pose information for localization purposes, the process is termed Cooperative Localization (CL) [78]. CL provides the means for an implicit sensor sharing, as localization information is dissipated over a wireless network to all the members of the group. In Chapter 2, it is shown that this results in considerable gains in terms of localization accuracy for all robots.

- **Simultaneous Localization and Mapping (SLAM):** When robots operate in unknown environments, they can process their sensor measurements to detect *features* (also termed *landmarks*), which can be used to aid localization. The term feature



(a) (b)

Figure 1.4: Examples of exteroceptive measurements, and the features extracted from them: (a) A camera image, with a few points of interest superimposed. These points are local maxima of the image intensity gradient. (b) A typical planar laser scan. The raw data consist of range measurements along 181 rays, evenly spaced in a 180° field of view. The detected features correspond to geometric corners.

in this context refers to a distinctive point of interest, which can be detected in the sensor data by applying an interest operator. In images, for example, features are typically selected as points where the image intensity gradient is large [56, 84], while in laser scanner data, corners and line segments are the most commonly detected features [136] (cf. Fig. 1.4). When the positions of the features in the robot's surroundings are not known in advance, they can be estimated along with the robot's trajectory (cf. Fig. 1.5). This is the task of Simultaneous Localization and Mapping (SLAM) [115, 148]. The main benefit of SLAM is that, since the point features are static, they serve as "localization beacons." Observing features over multiple time instants, or when the robot re-visits an area (this is often termed "loop closing"), results in bounded estimation errors over long time periods [105]. Thus, SLAM permits accurate, long-term localization in unknown environments, and is considered an enabling technology for robot autonomy [35].

- **Cooperative Simultaneous Localization and Mapping (C-SLAM):** When multiple robots operate in an area, and at least one of the robots is capable of obtain-

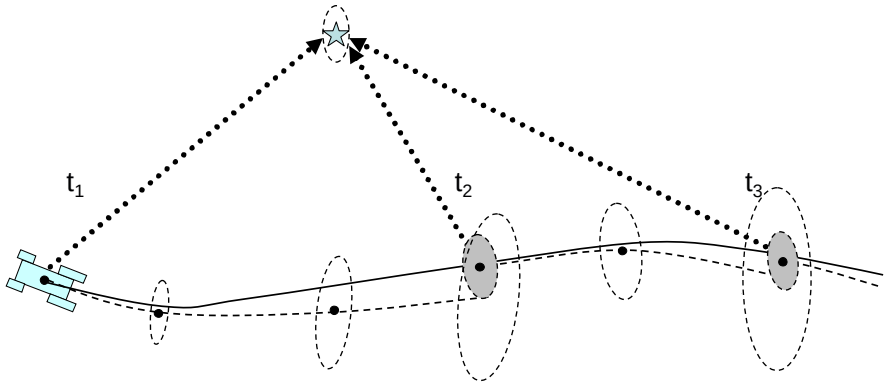


Figure 1.5: SLAM: At time t_1, the robot observes for the first time a static feature (denoted by a star), and computes an estimate of its position. At later time instants, t_2 and t_3, the robot re-observes the same feature, and employs these measurements in order to update its pose estimates, as well as the estimate of the feature's position. Every re-observation results in a reduction of uncertainty, and asymptotically, the errors of the robot's and feature's estimates remain bounded [105]. A similar situation arises when multiple cooperating robots localize as a group, while observing static features.

ing measurements to static features, then the positions of all robots and all features can be simultaneously estimated [104,108]. This process is termed Cooperative SLAM (C-SLAM), or multi-robot SLAM. The "sensor sharing" that occurs during C-SLAM, in conjunction with the use of static landmarks as localization anchors, results in bounded localization errors for all robots of the team (cf. Section 2.5).

1.2 Research objectives

CL, SLAM and C-SLAM have been very active topics of research in recent years. Specifically, mobile robot teams have attracted the interest of the robotics community, because of the increased efficiency and reliability offered when multiple robots cooperate for performing a task [16,67,127,132,168,170]. Several algorithms have been proposed for pose estimation in cooperating groups of robots (e.g., Extended Kalman Filtering (EKF) [137, 142], maximum likelihood estimation [62], particle filtering [43, 44, 63], set-based methods [87, 154], etc.), while the design of sensors for obtaining relative measurements [46, 52, 89, 132] has also been investigated.

On the other hand, SLAM (both single- and multi-robot) has been the most active

research area of mobile robotics for the past two decades. Research efforts have focused primarily on developing scalable estimation algorithms, capable of operating in large environments [53, 70, 70, 98, 119, 128, 157], while issues such as feature detection in raw sensor data [136,145], data association techniques for matching sensor readings to physical points of interest [116,144], implementation of SLAM with different sensing modalities [61,71,82,145], and designing estimators to tackle the nonlinearity of the problem [4, 30, 98] have also received considerable attention.

Despite the great research interest in the areas of CL, SLAM, and C-SLAM, the vast majority of work has focused on implementation techniques and on addressing the problem of computational complexity. As a result, the issue of *theoretical performance analysis* has largely been left unexplored. However, the lack of theoretical tools for *predicting* the attainable accuracy in a localization system is a significant impediment to robot design. Without such tools, a designer must resort to exhaustive simulations and/or experimentation to verify whether a given selection of the system parameters meets the accuracy requirements of the target application. In turn, this leads to increased requirements of time, cost, and effort for designing a robot system. Performance analysis tools, which are commonly available in other, more mature fields of engineering, are essential for the advancement and widespread adoption of robotics technology.

The lack of analytical means for predicting the accuracy of robot localization also hinders the development of methods for *accuracy optimization*. During a robot's mission, the computing resources available for localization are limited, due to the need to simultaneously perform multiple tasks (e.g., in a surveillance application, the robot might also be performing image analysis for detecting intruders). Naturally, when only limited resources are available for localization, the accuracy of the resulting estimates degrades. Therefore, determining methods for optimal utilization of the available resources, so as to achieve the highest possible estimation accuracy, is of great significance for mission success. To date, this task has been addressed mostly in ad-hoc ways, or by using a trial-and-error approach in system design, due to the absence of methods for analytically evaluating the impact of various design choices.

In this work, we address the aforementioned limitations, by proposing methods for *an-*

alytically characterizing and *optimizing* the accuracy of robot localization. Our approach is based on analytically studying the properties of localization, and then employing a rigorous methodology for optimal design of localization algorithms. Specifically, the three main contributions of this work are the following:

1.2.1 Localization performance characterization

The first key contribution of this work is the derivation of analytical expressions for characterizing the performance, in terms of positioning accuracy, of CL, SLAM, and C-SLAM. In particular, we derive analytical expressions that determine the *guaranteed accuracy* of localization, as closed-form functions of the relevant system design parameters. These parameters are (i) the accuracy of all available sensors, (ii) the number of robots and/or landmarks, (iii) the spatial distribution of the landmarks and robots, and (iv) the structure of the sensing graph, which determines which measurements are processed at each time instant. By allowing us to predict the localization accuracy attainable by a given robot design, these expressions facilitate the process of selecting system parameters for achieving the accuracy requirements of a given task. Moreover, by employing these expressions, we can analytically study the effect of the system parameters on the position uncertainty during CL, SLAM, and C-SLAM. This enables us to identify key properties of these localization methods, and develop a deeper understanding of the interactions between the various factors affecting accuracy.

1.2.2 Optimal resource utilization in robot formations

The second key contribution of this work is an algorithm that allows *optimal resource utilization* during CL in robot formations. When a team of robots localizes cooperatively, constraints on the available computational and communication resources may prohibit the robots from transmitting and processing all available sensor measurements. In this situation, a strategy for selecting which measurements to use at each time-step, so as to attain the highest possible localization accuracy, is required. To address this problem, we propose a methodology for *optimal sensor scheduling*, under the constraints imposed by the robots' limited resources. The developed algorithm computes the rate at which the measurements

of each of the available sensors should be processed, to ensure the best steady-state accuracy for the robots' pose estimates. This algorithm can be employed for determining the optimal sensing strategy for robot formations of any size and shape, comprised of robots with various types of sensors and sensing capabilities. By enabling robots to optimally utilize their available resources, the described methodology enables more cost-effective system design. Moreover, the improved localization accuracy can increase the operational capabilities of robot teams, allowing them to perform challenging tasks that require high-precision pose estimates.

1.2.3 Optimal use of transient feature information

The third key contribution of this work is a real-time algorithm that allows for optimal use of exteroceptive sensor data, at a low computational cost. In most realistic applications, a very large number of features can be detected in the exteroceptive sensor measurements (this is especially true in the case of visual sensing). An important observation is that the vast majority of these features are typically tracked for a small number of consecutive time-steps, and are never re-detected. Only a small fraction of features are tracked for longer time-periods, or when the robot re-visits an area. This motivates us to divide features into two categories: *stable* features, which are used for loop closing, and *transient* features, which are used for improving the accuracy of the robot motion estimates. In this work, we propose an algorithm, termed multi-state constraint Kalman filter (MSC-KF), that can optimally utilize the information content of the transient feature measurements, at a computational cost only linear in the number of such features. Moreover, we describe a dual-layer estimator architecture, where the MSC-KF is used in conjunction with a bundle-adjustment module, resulting in a localization system whose pose estimates are available in real-time and have bounded long-term errors. Due to its low computational requirements, and the fact that measurements are used in a statistically optimal fashion, the MSC-KF is ideal for applications where hard real-time constraints exist, and high localization accuracy is required.

1.3 Organization of the manuscript

In the following chapter, the issue of analytical performance characterization is addressed. First, the derivation of analytical expressions for the guaranteed accuracy of CL, SLAM and C-SLAM is presented, and these expressions are then employed for exploring several properties of the uncertainty in these classes of localization algorithms. Moreover, the derived theoretical results are validated through both numerical and real-world experiments. In Chapter 3, our methodology for attaining optimal resource utilization during CL in robot formations is presented. The derivation of the proposed algorithm is described, and its use is demonstrated in real-world experiments. Next, in Chapter 4, the MSC-KF algorithm, and its use in conjunction with a loop-closure module, is described in the context of vision-aided inertial navigation. Finally, Chapter 5 summarizes the key results of this work, and identifies interesting avenues for future research.

Chapter 2

Localization Performance Characterization

2.1 Introduction

In this chapter, we present analytical results that enable us to predict the accuracy of localization in the cases of CL, C-SLAM, and SLAM. Our approach is based on deriving *upper bounds* on the uncertainty of the state estimates, as closed-form expressions of all the relevant parameters of a given system. The reason for adopting this approach is the fact that, in general, the exact value of the uncertainty in a given localization task cannot be computed in advance. This uncertainty depends on the actual trajectories followed by the robots, and the placement of landmarks in the environment, in the case of (C-)SLAM. If these variables were known *a priori*, we would be able to predict the uncertainty of the robot pose estimates (cf. Chapter 3). However, in most practical cases, the robots' trajectories cannot be pre-specified, and thus an alternative method for predicting the accuracy of the pose estimates is required.

In our work, we achieve this goal by deriving the *guaranteed accuracy* of localization, as a closed-form function of the design parameters of a given system. These design parameters include (i) the characteristics of the robots' sensors, (ii) the number of robots and landmarks, (iii) the spatial distribution of landmarks, and (iv) the structure of the sensing graph describing the measurements between robots and/or landmarks. It should be pointed out

that, in most cases, these factors are either precisely known (e.g., the sensor characteristics), or their properties can be predicted in advance (e.g., the density of landmarks). Therefore, by applying the derived closed-form expressions during the design phase, informed decisions can be made, for example, about the necessary number of robots in a team, and the accuracy of their sensors, in order to meet task-imposed accuracy specifications.

Before presenting the details of our analysis, in the next section we discuss the related prior work. Section 2.3 presents the key elements of our methodology, while Sections 2.4-2.6 describe in detail the main results of this chapter. The theoretical results are validated by the experimental and simulation tests presented in Section 2.7, and finally, Section 2.8 summarizes the key findings of this chapter.

2.2 Prior work

In this section, we discuss prior approaches that study properties of the uncertainty in CL, C-SLAM and SLAM. We point out that, as discussed in Section 1.2, a substantial amount of work in the literature addresses the design of estimators suitable for real-time applications, sensor data processing, and several other implementation issues. However, since these topics are not the primary focus of our work, we will not discuss such approaches here. The interested reader is referred to [101,107,108] for an overview. In what follows, we concentrate on the (quite limited) prior work that examines the properties of the uncertainty during localization.

2.2.1 Cooperative localization

To the best of our knowledge, there exist only few cases in the literature where the properties of position uncertainty in CL have been studied. In [142], the improvement in localization accuracy is computed, after only a single update step, with respect to the previous values of position uncertainty. In this case, the robots' orientations are assumed to be perfectly known, and no expressions are derived for the propagation of the localization uncertainty with respect to time, or with respect to the accuracy of the odometric and relative position measurements.

In [133], the authors studied, in simulation, the effect of different robot-tracker sensing

modalities on the accuracy of CL. Statistical properties of the pose covariance were derived using a Monte-Carlo approach, and it was shown that, as the number of robots in the team increases, the robots' position estimates become more accurate. Moreover, the impact of the relative positions of the robots on their localization accuracy was demonstrated with numerical experiments. This work was the first to touch upon the issue of the time evolution of the uncertainty in CL; however, no analytical results were derived.

In [140, 141], the special case of N *identical* robots localizing cooperatively is considered. Under the assumption that every robot can always measure the relative positions of all other robots, an upper bound on the expected position covariance matrix is derived. The results of that work constitute the first analytical assessment of the positioning accuracy of multi-robot teams. However, the assumption of homogeneity and the requirement that every robot measures the relative position of all other robots in the team, limit their applicability to small groups of identical robots. In realistic scenarios, even teams consisting of identically built robots, may actually be heterogeneous, due to the inherent variability during the manufacturing of their sensors (cf. Section 2.7.1). In our work, the assumptions of a homogeneous group and of a complete measurement graph are removed.

2.2.2 SLAM and C-SLAM

One of the first attempts to study the properties of the covariance matrix of the state estimates in SLAM was presented in [50]. In this case, a Linear Time Invariant (LTI) SLAM model is employed, in which both the robot and the landmarks are constrained to lie in a one-dimensional (1D) space. For this simplistic model, the solution to the Riccati differential equation, that describes the time evolution of the covariance matrix of the position estimates, is derived in closed form. This work has also been extended to the case of a team of multiple vehicles performing C-SLAM in 1D [117]. Under the additional assumptions that (i) all robots can observe all landmarks, (ii) no robot-to-robot measurements occur, (iii) the initial covariance matrix of the feature map is diagonal, and (iv) the robots have perfect initial knowledge of their position, a closed-form solution to the Riccati equation is computed. These results are useful, in that they demonstrate some of the properties of the covariance matrix in single- and multi-robot SLAM. However, their

practical importance is limited by the fact that the analysis holds only for motion in 1D, and for a LTI system model.

A different set of properties of the covariance matrix in SLAM is studied in [25, 33, 118]. In particular, an LTI SLAM model is considered, and it is shown that (i) the covariance matrix of the landmarks' position estimates is decreasing monotonically, as more exteroceptive measurements are processed, and (ii) after sufficient time (i.e., asymptotically) the map estimates become fully correlated. Additionally, the authors derive a lower bound on the robot's and landmarks' covariance matrix, by considering the case in which the odometry measurements are perfect (i.e., noise-free). Since no additional uncertainty is introduced in the system during state propagation, this is the "best-case scenario," and the covariance of the state estimates in this hypothetical system defines a lower bound that depends only on the initial uncertainty of the robot's pose. These results are extended to the case of an LTI C-SLAM model in [41, 42].

In [65,66], a similar analysis is applied to EKF-based SLAM. In particular, lower bounds for the covariance are derived, by considering the cases in which a robot observes landmarks an infinite number of times, from either one or two positions. It should be pointed out, however, that in all the aforementioned approaches the derived lower bounds are independent of the accuracy of the robot's sensors, and thus cannot be employed for comparing the performance of robots with sensors of different quality. Moreover, if the robots' initial pose is perfectly known, which is a common situation in SLAM, the bounds are equal to zero, and thus they are uninformative. In contrast, the results presented in Sections 2.5 and 2.6 are applicable to this case of interest, and can be used to compute the guaranteed localization accuracy.

2.3 Overview of the approach

Robot localization is a nonlinear estimation problem, and consequently, no exact recursive estimator can be developed for it. This has led to the development of a large number of approximate estimators for robot localization (cf. Section 1.2). Depending on the particular estimator employed, the expression of the accuracy, or, equivalently, of the uncertainty, of the state estimates varies. For example, in the Extended Kalman Filter (EKF) [118], the

uncertainty is expressed by the covariance matrix, in particle filtering [98] by the sample covariance of the particles, in set-based methods [87] by the volume of the sets where the state is contained, etc. Since our goal is to derive closed-form expressions for the uncertainty, we have chosen to employ an EKF for our analysis. The reason is that the Riccati recursion (cf. (2.14)) provides the means for analytically studying the time-evolution of the state estimates' uncertainty in the EKF. However, it should be pointed out that as long as different estimators provide good approximations to the true pdf of the state (e.g., they do not fail due to large linearization errors, particle depletion, or similar issues), then their computed uncertainty estimates should all reflect the true uncertainty of the robot pose. In this case, the uncertainty metrics associated with all such estimators are equivalent, and the results of our work can be employed to characterize their performance.

In what follows, we briefly describe EKF-based localization, and introduce the Riccati recursion, which will play a key role in the ensuing analysis. In the EKF, our knowledge about the state at any time instant is described by a Gaussian probability density function (pdf) [92]. The *state estimate* is given by the mean of this pdf, while the *uncertainty* is described by its covariance matrix. The EKF evolves recursively in two steps: propagation and update. During propagation, the proprioceptive measurements are employed for *predicting* the state at the next time-step, while during updates, the exteroceptive measurements are utilized for *correcting* the estimate and improving its accuracy. The details of these two steps are described hereafter.

Consider a system, in which the state that we wish to estimate (e.g,. the position and orientation of the robots in a team) at time-step k is $\mathbf{X}_k$. The state evolves in time according to the nonlinear equation:

$$\mathbf{X}_{k+1} = \mathbf{f}(\mathbf{X}_k, \mathbf{u}_k) \tag{2.1}$$

where $\mathbf{u}_k$ is the actual control input (e.g., the robot linear and rotational velocities) at time-step k. The robots do not have direct access to the control input $\mathbf{u}_k$, but can obtain measurements of it, using their proprioceptive (e.g., odometric) sensors. These measure-

ments are described by the equation:

$$\mathbf{u}_{m_k} = \mathbf{u}_k + \mathbf{w}_k \tag{2.2}$$

where $\mathbf{w}_k$ is the measurement noise vector, modeled as a zero-mean, white Gaussian random variable, with known covariance $\mathbf{Q}_k$. In the EKF propagation phase, the measurement $\mathbf{u}_{m_k}$ is used to compute the predicted value of the state at time-step $k+1$ as follows:

$$\hat{\mathbf{X}}_{k+1|k} = \mathbf{f}(\hat{\mathbf{X}}_{k|k}, \mathbf{u}_{m_k}) \tag{2.3}$$

In this expression, $\hat{\mathbf{X}}_{k|k}$ and $\hat{\mathbf{X}}_{k+1|k}$ denote the estimates for the states at time-steps k and $k+1$, respectively, computed using all proprioceptive and exteroceptive measurements up to time-step k. The covariance matrix of the error in the state estimate $\hat{\mathbf{X}}_{k+1|k}$, denoted as $\mathbf{P}_{k+1|k}$, is computed as:

$$\mathbf{P}_{k+1|k} = \mathbf{\Phi}_k \mathbf{P}_{k|k} \mathbf{\Phi}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T \tag{2.4}$$

where $\mathbf{P}_{k|k}$ is the covariance matrix of the error in the estimate $\hat{\mathbf{X}}_{k|k}$, and

$$\mathbf{\Phi}_k = \nabla_{\mathbf{X}_k} \mathbf{f} \Big|_{\{\hat{\mathbf{X}}_{k|k}, \mathbf{u}_{m_k}\}} \tag{2.5}$$

$$\mathbf{G}_k = \nabla_{\mathbf{u}_k} \mathbf{f} \Big|_{\{\hat{\mathbf{X}}_{k|k}, \mathbf{u}_{m_k}\}} \tag{2.6}$$

are the Jacobians of $\mathbf{f}$ with respect to the state and the control input, respectively. Equations (2.3) and (2.4) describe the propagation phase of the EKF.

The exteroceptive measurements recorded by the robots at time-step $k+1$ are described by a (generally nonlinear) equation of the form:

$$\mathbf{z}_{k+1} = \mathbf{h}(\mathbf{X}_{k+1}) + \mathbf{n}_{k+1} \tag{2.7}$$

where $\mathbf{n}_{k+1}$ is the exteroceptive measurement noise vector, modeled as a white, zero-mean Gaussian random variable with covariance matrix $\mathbf{R}_{k+1}$. The EKF update proceeds by first

computing the *residual*:

$$\mathbf{r}_{k+1} = \mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1} = \mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{X}}_{k+1|k}) \tag{2.8}$$

By employing first-order Taylor series approximation of the measurement function $\mathbf{h}$, the residual can be written as:

$$\mathbf{r}_{k+1} \simeq \mathbf{H}_{k+1}\widetilde{\mathbf{X}}_{k+1|k} + \mathbf{n}_{k+1} \tag{2.9}$$

where $\widetilde{\mathbf{X}}_{k+1|k} = \mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1|k}$ is the *error* in the estimate $\hat{\mathbf{X}}_{k+1|k}$, and

$$\mathbf{H}_{k+1} = \nabla_{\mathbf{X}_{k+1}}\mathbf{h}\Big|_{\hat{\mathbf{X}}_{k+1|k}} \tag{2.10}$$

is the Jacobian of the measurement function $\mathbf{h}$ with respect to the state. Based on the linearization in (2.9), the EKF proceeds to update the state estimate similar to the Kalman filter for linear systems. Specifically, the state estimate is updated by the equation:

$$\hat{\mathbf{X}}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} + \mathbf{K}_{k+1}\mathbf{r}_{k+1} \tag{2.11}$$

where the Kalman gain, $\mathbf{K}_{k+1}$, is given by:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T \left(\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T + \mathbf{R}_{k+1}\right)^{-1} \tag{2.12}$$

while the covariance matrix of the updated state estimate is given by:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T \left(\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T + \mathbf{R}_{k+1}\right)^{-1} \mathbf{H}_{k+1}\mathbf{P}_{k+1|k} \tag{2.13}$$

By combining the last expression with the covariance propagation equation (cf. (2.4), we obtain the *Riccati recursion*, which describes the time evolution of the covariance matrix:

$$\begin{aligned}\mathbf{P}_{k+2|k+1} = \mathbf{\Phi}_{k+1} \Big(\mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T \left(\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^T + \mathbf{R}_{k+1}\right)^{-1} \mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\Big) \mathbf{\Phi}_{k+1}^T \\ + \mathbf{G}_{k+1}\mathbf{Q}_{k+1}\mathbf{G}_{k+1}^T\end{aligned} \tag{2.14}$$

To simplify the notation, we set $\mathbf{P}_k = \mathbf{P}_{k+1|k}$, and $\mathbf{P}_{k+1} = \mathbf{P}_{k+2|k+1}$. Thus the Riccati recursion assumes the simpler form:

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}_{k+1}^T\left(\mathbf{H}_{k+1}\mathbf{P}_k\mathbf{H}_{k+1}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}_{k+1}\mathbf{P}_k\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}_{k+1}\mathbf{Q}_{k+1}\mathbf{G}_{k+1}^T \tag{2.15}$$

Studying the properties of the Riccati recursion allows us to draw conclusions for the localization uncertainty of a given system. A key observation is that the right-hand side of the above expression is a function of the noise covariance matrices, $\mathbf{Q}_{k+1}$ and $\mathbf{R}_{k+1}$, which are known in advance, *and* of the Jacobian matrices $\mathbf{\Phi}_{k+1}$, $\mathbf{G}_{k+1}$, and $\mathbf{H}_{k+1}$. These Jacobian matrices depend on the values of the state estimates themselves, and as a result, the time evolution of the covariance $\mathbf{P}_k$ will also depend on the values of the state. This means that in order to predict the accuracy of localization, we would need to know in advance the trajectories followed by the robots, as well as other factors, such as the distribution of landmarks. However, this is not practical from a design standpoint, since it implies that a very large number of trajectories would need to be simulated, to guarantee that a given system meets certain accuracy specifications.

To address this problem, our approach relies on computing an upper bound on the covariance, for every time-step k. This bound provides us with the guaranteed accuracy of localization, as a function of *a priori* known parameters, regardless of the trajectory that the robots may follow. Moreover, it is the case for many practical scenarios, that even though the exact values of the states may not be known in advance, a statistical description of them may be available. For example, we may be able to model the pose of the robots by a known pdf in their operational area. We utilize this additional knowledge in order to obtain a tighter upper bound on the *expected* covariance of the position estimates in CL and C-SLAM.

The analysis in the following sections consists of the following steps: first, we formulate an appropriate Riccati recursion with special structure, for each of the localization cases considered (i.e., CL, C-SLAM, and SLAM). Subsequently, we utilize the monotonicity and concavity properties of the Riccati, in order to construct a *constant-coefficient* recursion, whose solution is provably an upper bound on the actual covariance of the state estimates.

In this step, the following two lemmas will be employed:

Lemma 1 *Consider the Riccati recursion*

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T \tag{2.16}$$

If $\mathbf{R}_u$ *and* $\mathbf{Q}_u$ *are constant matrices such that* $\mathbf{R}_u \succeq \mathbf{R}_k$ *and* $\mathbf{Q}_u \succeq \mathbf{Q}_k$*, for all* $k \geq 0$*, then the solution to the Riccati recursion*

$$\mathbf{P}_{k+1}^u = \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k^u - \mathbf{P}_k^u\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^u\mathbf{H}^T + \mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_k^u\right)\mathbf{\Phi}_{k+1} + \mathbf{G}\mathbf{Q}_u\mathbf{G}^T \tag{2.17}$$

with the initial condition $\mathbf{P}_0^u = \mathbf{P}_0$*, satisfies* $\mathbf{P}_k^u \succeq \mathbf{P}_k$ *for all* $k \geq 0$*.*

Proof: See Appendix A.1. ∎

Lemma 2 *Consider the Riccati recursion*

$$\mathbf{P}_{k+1} = \mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T \tag{2.18}$$

If $\bar{\mathbf{R}} = E\{\mathbf{R}_{k+1}\}$ *and* $\bar{\mathbf{Q}} = E\{\mathbf{Q}_{k+1}\}$ *are the expected values of matrices* $\mathbf{R}_{k+1}$ *and* $\mathbf{Q}_{k+1}$*, respectively, then the solution to the following Riccati recursion*

$$\bar{\mathbf{P}}_{k+1} = \bar{\mathbf{P}}_k - \bar{\mathbf{P}}_k\mathbf{H}^T\left(\mathbf{H}\bar{\mathbf{P}}_k\mathbf{H}^T + \bar{\mathbf{R}}\right)^{-1}\mathbf{H}\bar{\mathbf{P}}_k + \mathbf{G}\bar{\mathbf{Q}}\mathbf{G}^T \tag{2.19}$$

with initial condition $\bar{\mathbf{P}}_0 = \mathbf{P}_0$*, satisfies* $\bar{\mathbf{P}}_k \succeq E\{\mathbf{P}_k\}$ *for all* $k \geq 0$*.*

Proof: See Appendix A.2. ∎

These lemmas allow us to derive constant-coefficient Riccati recursions, whose solutions are upper bounds to the worst case and the expected value of the uncertainty, respectively. Once these recursions are obtained, we subsequently evaluate their steady-state (i.e., asymptotic) solution. This solution provides us with a description of the long-term performance of the localization system, which can be of critical importance for applications. Finally, once the asymptotic solution has been obtained in closed form, we proceed to examine its properties, so as to identify the way in which key factors affect the estimates' accuracy. In

the remaining sections of this chapter, these steps are described in detail for the problems of CL, C-SLAM, and SLAM.

2.4 Cooperative localization

We first study the properties of the position uncertainty in CL [100, 107]. Consider a group of N robots moving in 2D that employ odometry and relative position measurements (e.g., provided by laser range finders) to cooperatively estimate their poses. In our formulation, we assume that each robot has access to measurements of its absolute orientation, and that an upper bound on the variance of these measurements can be determined *a priori*. This is the case, for example, when each robot is equipped with a heading sensor of limited accuracy (e.g., a compass [34, 49] or a sun sensor [48, 166]) that directly measures its orientation, or if the robots infer their orientation from measurements of the structure of the environment in their surroundings (e.g., from the direction of the walls when this is known *a priori*) [129]. Alternatively, absolute orientation measurements can be obtained by observing objects in the horizon [161], or equivalently, the vanishing points of sets of parallel lines [11].

The variance of the absolute orientation measurements that each robot receives defines an upper bound on each robot's orientation uncertainty. The availability of such a bound enables us to decouple the task of position estimation from that of orientation estimation, for the purpose of determining upper bounds on the performance of CL. Specifically, we formulate a state vector comprising of only the positions of the N robots, and the orientation estimates are used as inputs to the system, of which noise-corrupted observations are available. Clearly, the resulting EKF-based estimator is a suboptimal one, since the correlations that exist between the position and orientation estimates of the robots are discarded. Thus, by deriving an upper bound on the covariance of the estimates produced by this suboptimal "position-only" estimator, we simultaneously determine an upper bound on the covariance of the position estimates that would result from using a "full-state" EKF estimator [137].

We should note that, for most cases in practice, the condition for bounded orientation uncertainty is satisfied. If special care is not taken and the errors in the orientation estimates of the robots are allowed to grow unbounded, any EKF-based estimator of their poses will eventually diverge [92]. The significance of having small orientation errors for the

consistency of the EKF has also been demonstrated in [18, 70]. Thus, the requirement for bounded orientation errors is not an artificially imposed assumption; it is essentially a prerequisite for performing EKF-based localization. In fact, if we can determine the maximum tolerable value of the orientation variance, so as to ensure that the linearization errors are acceptably small, we can use this variance value in the derivations that follow.

Throughout this chapter, we consider that all robots move constantly in a random fashion (i.e., no specific formation is assumed, which is the case in Chapter 3). At every time-step, some (or all) robots record relative position measurements, and use this information to improve the position estimates for all members of the group. During each EKF update cycle, all exteroceptive measurements, as well as the current position estimates of the robots, must be available to the estimator. These can then be fused either in a distributed scheme [137], or at a central fusion center. Therefore, it is assumed that an appropriate communication network exists, enabling all robots to transmit all necessary information.

A key element in this analysis is the Relative Position Measurement Graph (RPMG), which is defined as a graph whose vertices represent robots in the group and its directed edges correspond to relative position measurements (Fig. 2.6). That is, if robot i measures the relative position of robot j, the RPMG contains a directed edge from vertex i to vertex j. In this work, we primarily consider the most challenging scenario, where the absolute positions of the robots cannot be measured or inferred. However, the case where global positioning information is available to at least one of the robots in the group, is also addressed in our formulation. In what follows, we present the EKF equations for the decoupled position-only estimator, and derive the Riccati recursion that will serve as the basis of our analysis.

2.4.1 EKF propagation

We first study the influx of uncertainty to the system, due to the noise in the odometric measurements of the robots. The discrete-time motion equations for the i-th robot of the team are

$$x_i(k+1) \;=\; x_i(k) + V_i(k)\delta t \cos(\phi_i(k)) \tag{2.20}$$

$$y_i(k+1) \quad = \quad y_i(k) + V_i(k)\delta t \sin(\phi_i(k)) \tag{2.21}$$

where $V_i(k)$ denotes the robot's translational velocity at time k and δt is the sampling period. The estimates of the robot's position are propagated using the measurements of the robot's velocity, $V_{m_i}(k)$, and the estimates of the robot's orientation, $\hat{\phi}_i(k)$:

$$\begin{aligned} \hat{x}_{i_{k+1|k}} &= \hat{x}_{i_{k|k}} + V_{m_i}(k)\delta t \cos(\hat{\phi}_i(k)) \\ \hat{y}_{i_{k+1|k}} &= \hat{y}_{i_{k|k}} + V_{m_i}(k)\delta t \sin(\hat{\phi}_i(k)) \end{aligned}$$

By linearizing (2.20) and (2.21), the error propagation equation for the robot position is derived:

$$\begin{aligned} &\begin{bmatrix} \widetilde{x}_{i_{k+1|k}} \\ \widetilde{y}_{i_{k+1|k}} \end{bmatrix} = \begin{bmatrix} \widetilde{x}_{i_{k|k}} \\ \widetilde{y}_{i_{k|k}} \end{bmatrix} + \begin{bmatrix} \delta t \cos(\hat{\phi}_i(k)) & -V_{m_i}(k)\delta t \sin(\hat{\phi}_i(k)) \\ \delta t \sin(\hat{\phi}_i(k)) & V_{m_i}(k)\delta t \cos(\hat{\phi}_i(k)) \end{bmatrix} \begin{bmatrix} w_{V_i}(k) \\ \widetilde{\phi}_i(k) \end{bmatrix} \\ &\Leftrightarrow \ \widetilde{\mathbf{p}}_{r_i\ k+1|k} = \widetilde{\mathbf{p}}_{r_i\ k|k} + \mathbf{G}_i(k)\mathbf{w}_i(k) \end{aligned} \tag{2.22}$$

where $\widetilde{\mathbf{p}}_{r_i} = [\widetilde{x}_i \ \ \widetilde{y}_i]^T$ is the error in the i-th robot's position estimate and $V_{m_i}(k) = V_i(k) - w_{V_i}(k)$ are the measurements of the translational velocity of the robot, contaminated by a white, zero-mean, Gaussian noise process with known variance $\sigma^2_{V_i}$. The errors in the orientation estimates, $\widetilde{\phi}_i(k) = \phi_i(k) - \hat{\phi}_i(k)$, are modeled by a white zero-mean Gaussian noise process, whose variance, $\sigma^2_{\phi_i} = E\{\widetilde{\phi}_i^{\,2}\}$, is bounded.

The covariance function of the system noise affecting the i-th robot is

$$\begin{aligned} \mathbf{Q}_i(k,\ell) &= E\{\mathbf{G}_i(k)\mathbf{w}_i(k)\mathbf{w}_i(\ell)^T\mathbf{G}_i(\ell)^T\} \\ &= \mathbf{C}(\hat{\phi}_i(k)) \begin{bmatrix} \delta t^2\sigma^2_{V_i} & 0 \\ 0 & \delta t^2\sigma^2_{\phi_i} V^2_{m_i}(k) \end{bmatrix} \mathbf{C}(\hat{\phi}_i(k))^T\delta_{k\ell} \\ &= \mathbf{C}(\hat{\phi}_i(k))\mathbf{Q}_{d_i}\mathbf{C}(\hat{\phi}_i(k))^T\delta_{k\ell} \end{aligned} \tag{2.23}$$

where $\delta_{k\ell}$ is the Kronecker delta function, and $\mathbf{C}(\hat{\phi}_i(k))$ is the 2×2 rotation matrix associated

with $\hat{\phi}_i(k)$:

$$\mathbf{C}(\hat{\phi}_i(k)) = \begin{bmatrix} \cos(\hat{\phi}_i(k)) & -\sin(\hat{\phi}_i(k)) \\ \sin(\hat{\phi}_i(k)) & \cos(\hat{\phi}_i(k)) \end{bmatrix} \tag{2.24}$$

The state vector for the entire system is defined as the stacked vector comprising the positions of the N robots, i.e.,

$$\mathbf{X}(k) = \begin{bmatrix} \mathbf{p}_{r_1}^T(k) & \mathbf{p}_{r_2}^T(k) & \cdots & \mathbf{p}_{r_N}^T(k) \end{bmatrix}^T \tag{2.25}$$

From (2.22) we see that the error propagation equation for the entire EKF state vector is:

$$\mathbf{X}_{k+1|k} = \mathbf{X}_{k|k} + \mathbf{w}(k) \tag{2.26}$$

where $\mathbf{w}(k)$ is a block vector with elements $\mathbf{G}_i(k)\mathbf{w}_i(k),\ i = 1, \ldots N$ (cf. (2.22)). Since the errors in the odometric measurements of different robots are uncorrelated, the covariance matrix of $\mathbf{w}(k)$ is given by

$$\mathbf{Q}_r(k) = \begin{bmatrix} \mathbf{Q}_1(k,k) & \cdots & \mathbf{0}_{2\times 2} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{2\times 2} & \cdots & \mathbf{Q}_N(k,k) \end{bmatrix} = \mathbf{Diag}\left(\mathbf{Q}_i(k,k)\right) \tag{2.27}$$

and therefore the covariance propagation equation is written as:

$$\mathbf{P}_{k+1|k} = \mathbf{P}_{k|k} + \mathbf{Q}_r(k) \tag{2.28}$$

2.4.2 Exteroceptive measurement models

Relative Position Measurements

At this point, we consider the exteroceptive measurements that the robots process to update their position estimates. The relative position measurement, $\mathbf{z}_{ij}$, between robot i and j at time-step $k+1$ is defined as

$$\mathbf{z}_{ij}(k+1) \quad = \quad \mathbf{C}^T\big(\phi_i(k+1)\big)\left(\mathbf{p}_{r_j}(k+1) - \mathbf{p}_{r_i}(k+1)\right) + \mathbf{n}_{z_{ij}}(k+1)$$

$$= \quad \mathbf{C}^T(\phi_i(k+1))\boldsymbol{\Delta}\mathbf{p}_{ij}(k+1) + \mathbf{n}_{z_{ij}}(k+1) \tag{2.29}$$

where $\mathbf{n}_{z_{ij}}(k+1)$ is a white, zero-mean, Gaussian noise process affecting the measurement.

By linearizing (2.29), the measurement residual is obtained:

$$\begin{aligned}
\mathbf{r}_{ij}(k+1) &= \mathbf{z}_{ij}(k+1) - \hat{\mathbf{z}}_{ij}(k+1) \\
&\simeq \mathbf{H}_{ij}(k+1)\widetilde{\mathbf{X}}_{k+1|k} + \boldsymbol{\Gamma}_{ij}(k+1)\mathbf{n}_{ij}(k+1)
\end{aligned}$$

where

$$\mathbf{H}_{ij}(k+1) = \mathbf{C}^T(\hat{\phi}_i(k+1))\mathbf{H}_{o_{ij}} \tag{2.30}$$

$$\mathbf{H}_{o_{ij}} = \begin{bmatrix} \mathbf{0}_{2\times 2} & \cdots & \underbrace{-\mathbf{I}_2}_{i} & \cdots & \underbrace{\mathbf{I}_2}_{j} & \cdots & \mathbf{0}_{2\times 2} \end{bmatrix} \tag{2.31}$$

$$\boldsymbol{\Gamma}_{ij}(k+1) = \begin{bmatrix} \mathbf{I}_2 & -\mathbf{C}^T(\hat{\phi}_i(k+1))\mathbf{J}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}(k+1) \end{bmatrix}, \quad \mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$$\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}(k+1) = \hat{\mathbf{p}}_{r_j\ k+1|k} - \hat{\mathbf{p}}_{r_i\ k+1|k}$$

$$\mathbf{n}_{ij}(k+1) = \begin{bmatrix} \mathbf{n}_{z_{ij}}(k+1) \\ \widetilde{\phi}_i(k+1) \end{bmatrix}$$

In Appendix A.3 it is shown that, if robot i performs M_i relative position measurements at each time-step, the covariance matrix of the errors in these measurements is given by:

$$\boldsymbol{R}_i(k+1) = \boldsymbol{\Xi}^T_{\hat{\phi}_i}(k+1)\boldsymbol{R}_{o_i}(k+1)\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1) \tag{2.32}$$

where $\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1) = \mathbf{I}_{M_i} \otimes \mathbf{C}(\hat{\phi}_i(k+1))$, and

$$\begin{aligned}
\boldsymbol{R}_{o_i}(k+1) &= \sigma^2_{\rho_i}\mathbf{I}_{2M_i} - \mathbf{D}_i(k+1)\,\mathrm{diag}\left(\frac{\sigma^2_{\rho_i}}{\hat{\rho}_{ij}(k+1)^2}\right)\mathbf{D}_i(k+1)^T + \sigma^2_{\theta_i}\mathbf{D}_i(k+1)\mathbf{D}_i(k+1)^T \\
&+ \sigma^2_{\phi_i}\mathbf{D}_i(k+1)\mathbf{1}_{M_i\times M_i}\mathbf{D}_i(k+1)^T
\end{aligned} \tag{2.33}$$

In this last expression, $\mathbf{D}_i(k+1) = \mathbf{Diag}\left(\mathbf{J}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}(k+1)\right)$ is the block diagonal matrix with diagonal elements $\mathbf{J}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}(k+1)$, $j \in \mathcal{N}_{M_i}$, where $\mathcal{N}_{M_i}$ is the set of the indices of the robots j observed by robot i. Additionally, σ_{ρ_i} and σ_{θ_i} are the standard deviations of range and

bearing measurements (which, combined, comprise the relative position measurement), and $\hat{\rho}_{ij}(k+1)$ is the estimated distance between robots i and j at time-step $k+1$.

The measurement matrix describing the relative position measurements obtained by robot i at each time-step is a matrix whose block rows are $\mathbf{H}_{ij}$, $j \in \mathcal{N}_{M_i}$, i.e.:

$$\boldsymbol{H}_i(k+1) = \boldsymbol{\Xi}^T_{\hat{\phi}_i}(k+1)\boldsymbol{H}_{o_i} \tag{2.34}$$

where $\boldsymbol{H}_{o_i}$ is a *constant* matrix with block rows $\mathbf{H}_{o_{ij}}$,$j \in \mathcal{N}_{M_i}$ (cf. (2.31)).

The measurement matrix for the entire system, $\mathbf{H}(k+1)$, which describes the relative position measurements recorded by all robots, is the block matrix with block rows $\boldsymbol{H}_i(k+1)$, i.e.,

$$\mathbf{H}(k+1) = \begin{bmatrix} \boldsymbol{\Xi}^T_{\hat{\phi}_1}(k+1)\boldsymbol{H}_{o_1} \\ \boldsymbol{\Xi}^T_{\hat{\phi}_2}(k+1)\boldsymbol{H}_{o_2} \\ \vdots \\ \boldsymbol{\Xi}^T_{\hat{\phi}_N}(k+1)\boldsymbol{H}_{o_N} \end{bmatrix} = \mathbf{Diag}\left(\boldsymbol{\Xi}^T_{\hat{\phi}_i}(k+1)\right) \begin{bmatrix} \boldsymbol{H}_{o_1} \\ \boldsymbol{H}_{o_2} \\ \vdots \\ \boldsymbol{H}_{o_N} \end{bmatrix} = \boldsymbol{\Xi}^T(k+1)\mathbf{H}_o \tag{2.35}$$

where $\boldsymbol{\Xi}(k+1) = \mathbf{Diag}\left(\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1)\right)$ is a block diagonal matrix with block elements $\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1)$, for $i = 1 \ldots N$, and $\mathbf{H}_o$ is a matrix with block rows $\boldsymbol{H}_{o_i}$, $i = 1 \ldots N$. Since the measurements performed by different robots are independent, the measurement covariance matrix for the entire system is given by

$$\mathbf{R}(k+1) = \mathbf{Diag}\left(\boldsymbol{R}_i(k+1)\right) = \mathbf{Diag}\left(\boldsymbol{\Xi}^T_{\hat{\phi}_i}\boldsymbol{R}_{o_i}(k+1)\boldsymbol{\Xi}_{\hat{\phi}_i}\right) = \boldsymbol{\Xi}^T(k+1)\mathbf{R}_o(k+1)\boldsymbol{\Xi}(k+1) \tag{2.36}$$

where $\mathbf{R}_o(k+1)$ is a block diagonal matrix with block elements $\boldsymbol{R}_{o_i}(k+1)$, $i = 1 \ldots N$.

Absolute position measurements

Up to this point, only relative position measurements have been considered. If any of the robots, e.g., robot ℓ, has access to absolute positioning information, such as GPS measure-

ments, the corresponding submatrix element of $\mathbf{H}(k)$ is:

$$\mathbf{H}_{a_\ell} = \begin{bmatrix} \mathbf{0}_{2\times 2} & \cdots & \underbrace{\mathbf{I}_2}_{\ell} & \cdots & \mathbf{0}_{2\times 2} \end{bmatrix} \tag{2.37}$$

while $\mathbf{R}_{a_\ell}$, the covariance of the absolute position measurement, is a constant provided by the specifications of the absolute positioning sensor. To account for the absolute position measurements, the matrix $\mathbf{H}(k+1)$ in (2.42) is augmented by simply appending the appropriate block rows $\boldsymbol{H}_{a_\ell}$, while $\mathbf{R}(k+1)$ is augmented by appending the matrices $\mathbf{R}_{a_\ell}$ on the diagonal. Specifically, we obtain:

$$\mathbf{H}(k+1) = \begin{bmatrix} \boldsymbol{\Xi}^T_{\hat{\phi}_1}(k+1)\boldsymbol{H}_{o_1} \\ \boldsymbol{\Xi}^T_{\hat{\phi}_2}(k+1)\boldsymbol{H}_{o_2} \\ \vdots \\ \boldsymbol{\Xi}^T_{\hat{\phi}_N}(k+1)\boldsymbol{H}_{o_N} \\ \mathbf{H}_{a_\ell} \end{bmatrix} = \mathbf{Diag}\left(\boldsymbol{\Xi}^T_{\hat{\phi}_i}(k+1), \mathbf{I}_2\right) \begin{bmatrix} \boldsymbol{H}_{o_1} \\ \boldsymbol{H}_{o_2} \\ \vdots \\ \boldsymbol{H}_{o_N} \\ \mathbf{H}_{a_\ell} \end{bmatrix} = \boldsymbol{\Xi}^T(k+1)\mathbf{H}_o \tag{2.38}$$

where now $\boldsymbol{\Xi}(k+1) = \mathbf{Diag}\left(\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1), \mathbf{I}_2\right)$ is a block diagonal matrix with block elements $\boldsymbol{\Xi}_{\hat{\phi}_i}(k+1), i = 1\ldots N$, and $\mathbf{I}_2$, while $\mathbf{H}_o$ is a matrix with block rows $\boldsymbol{H}_{o_i}$, $i = 1\ldots N$, and $\mathbf{H}_{a_\ell}$. Since the measurements performed by different robots are independent, the measurement covariance matrix for the entire system is given by

$$\mathbf{R}(k+1) = \boldsymbol{\Xi}^T(k+1)\mathbf{R}_o(k+1)\boldsymbol{\Xi}(k+1) \tag{2.39}$$

where

$$\mathbf{R}_o(k) = \begin{bmatrix} \mathbf{Diag}\left(\boldsymbol{R}_{o_i}(k)\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{Diag}\left(\mathbf{R}_{a_\ell}\right) \end{bmatrix} \tag{2.40}$$

We note that the case where more than one robot is recording absolute-position measurements can be treated in the same way, by further augmenting the above matrices.

2.4.3 The Riccati recursion

The covariance update equation of the EKF is written as:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}(k+1)^T \left(\mathbf{H}(k+1)\mathbf{P}_{k+1|k}\mathbf{H}(k+1)^T + \mathbf{R}(k+1)\right)^{-1} \mathbf{H}(k+1)\mathbf{P}_{k+1|k} \tag{2.41}$$

Substitution from (2.35) and (2.39) (or from (2.35) and (2.36), if no absolute-position measurements are available) and simple algebraic manipulation leads to the orientation-dependent terms $\mathbf{\Xi}(k+1)$ being cancelled out, owing to the property $\mathbf{\Xi}(k+1)\mathbf{\Xi}(k+1)^T = \mathbf{I}$. We thus obtain the expression:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}_o^T \left(\mathbf{H}_o\mathbf{P}_{k+1|k}\mathbf{H}_o^T + \mathbf{R}_o(k+1)\right)^{-1} \mathbf{H}_o\mathbf{P}_{k+1|k} \tag{2.42}$$

By combining this expression with the covariance propagation equation (cf. (2.28)), we obtain the Riccati recursion describing the time evolution of the position estimates' covariance during CL:

$$\mathbf{P}_{k+1} \;=\; \mathbf{P}_k - \mathbf{P}_k\mathbf{H}_o^T \left(\mathbf{H}_o\mathbf{P}_k\mathbf{H}_o^T + \mathbf{R}_o(k+1)\right)^{-1} \mathbf{H}_o\mathbf{P}_k + \mathbf{Q}_r(k+1) \tag{2.43}$$

where the substitutions $\mathbf{P}_k = \mathbf{P}_{k+1|k}$ and $\mathbf{P}_{k+1} = \mathbf{P}_{k+2|k+1}$ have been introduced to simplify the notation. The initial value of this recursion, $\mathbf{P}_0$, is equal to the initial covariance matrix of the team's position estimates.

2.4.4 Structure of the measurement matrix

It will be useful for our analysis to examine the structure of the matrix $\mathbf{H}_o$ more closely. From (2.31) we conclude that each of the block rows of $\mathbf{H}_o$ that describes a relative position measurement contains a term $\mathbf{I}_2$ corresponding to the measured robot, and a term $-\mathbf{I}_2$ corresponding to the measuring robot. As a result, the matrix $\mathbf{H}_o$ can be written as:

$$\mathbf{H}_o = \mathbf{A}_{\mathrm{RPMG}} \otimes \mathbf{I}_2$$

where $\mathbf{A}_{\text{RPMG}}$ is the *incidence matrix* of the RPMG describing the relative position measurements recorded at each time-step. If, in addition to relative positions, some of the robots in the team also record measurements of absolute position, this relationship still holds, and in this case the RPMG is a *grounded graph* [95].

This observation indicates the close connection between the structure of the RPMG and the equations describing the time evolution of the positioning uncertainty during CL. Moreover, we can now employ known properties of the incidence matrix, to derive useful properties of $\mathbf{H}_o$. For instance, when only relative position measurements are recorded, the incidence matrix of the RPMG is rank-deficient. The sum of the elements of each row of $\mathbf{A}_{\text{RPMG}}$ equals zero, which implies that:

$$\begin{aligned} \mathbf{A}_{\text{RPMG}}\mathbf{1}_{N\times 1} &= \mathbf{0}_{N\times 1} \Rightarrow \\ \mathbf{H}_o\left(\mathbf{1}_{N\times 1}\otimes\mathbf{I}_2\right) &= \mathbf{0}_{2N\times 2} \Rightarrow \\ \mathbf{H}(k)\left(\mathbf{1}_{N\times 1}\otimes\mathbf{I}_2\right) &= \mathbf{0}_{2N\times 2} \end{aligned} \tag{2.44}$$

Using this result, it is trivial to show that the system is not observable (intuitively, this means that any displacement of the entire robot team with respect to the origin of the global coordinate frame will not cause a change in the measurements). Additionally, it is clear that a basis of the unobservable subspace is formed by the columns of the matrix $\mathbf{V}_{\text{unobs}} = \mathbf{1}_{N\times 1}\otimes\mathbf{I}_2$.

2.4.5 Upper bound on the worst-case covariance

Having formulated the Riccati recursion in (2.43), we can now apply Lemma 1 to compute an upper bound on the robots' position uncertainty. By inspection, we see that the recursion (2.43) is in the form of (2.16), with the correspondences:

$$\mathbf{\Phi}_k \leftrightarrow \mathbf{I}_{2N}, \quad \mathbf{G} \leftrightarrow \mathbf{I}_{2N}, \quad \mathbf{H} \leftrightarrow \mathbf{H}_o, \quad \mathbf{R}_{k+1} \leftrightarrow \mathbf{R}_o(k+1), \quad \mathbf{Q}_{k+1} \leftrightarrow \mathbf{Q}_r(k+1) \tag{2.45}$$

In order to apply Lemma 1, we need to first compute appropriate upper bounds for the matrices $\mathbf{Q}_r(k+1)$ and $\mathbf{R}_o(k+1)$.

An upper bound for $\mathbf{Q}_r(k+1)$ is derived by noting that since $\mathbf{C}(\hat{\phi}_i)$ is an orthonormal ma-

trix, the eigenvalues of $\mathbf{Q}_i(k+1,k+1)$ are equal to $\delta t^2 \sigma_{V_i}^2$ and $\delta t^2 \sigma_{\phi_i}^2 V_{m_i}^2(k+1)$ (cf. (2.23)). Assuming that the maximum velocity of robot i is equal to $V_{\max}$, we denote

$$q_i = \max\left(\delta t^2 \sigma_{V_i}^2, \delta t^2 V_{\max}^2 \sigma_{\phi_i}^2\right) \tag{2.46}$$

This definition states that q_i is the maximum eigenvalue of $\mathbf{Q}_i(k+1,k+1)$, and therefore $\mathbf{Q}_i(k+1,k+1) \preceq q_i \mathbf{I}_2$, which implies that an upper bound for $\mathbf{Q}_r(k+1)$ is:

$$\mathbf{Q}_r(k+1) \preceq \mathbf{Diag}\left(q_i \mathbf{I}_2\right) = \mathbf{Q}_u \tag{2.47}$$

An upper bound for $\mathbf{R}_o(k+1)$ can be derived by considering the maximum distance, ρ_o, at which relative position measurements can be recorded by each robot. This distance can, for example, be determined by the maximum range of the robots' relative position sensors, or, by the size of the area in which the robots operate. In Appendix A.4 it is shown that:

$$\boldsymbol{R}_{o_i}(k+1) \preceq \left(\sigma_{\rho_i}^2 + M_i \sigma_{\phi_i}^2 \rho_o^2 + \sigma_{\theta_i}^2 \rho_o^2\right) \mathbf{I}_{2M_i} = r_i \mathbf{I}_{2M_i}$$

and thus an upper bound on $\mathbf{R}_o(k+1)$ is given by

$$\mathbf{R}_u = \begin{bmatrix} \mathbf{Diag}\left(r_i \mathbf{I}_{2M_i}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{Diag}\left(\mathbf{R}_{a_\ell}\right) \end{bmatrix}, \quad r_i = \sigma_{\rho_i}^2 + M_i \sigma_{\phi_i}^2 \rho_o^2 + \sigma_{\theta_i}^2 \rho_o^2 \tag{2.48}$$

We note at this point that the upper bounds derived in the preceding expressions are valid only for the particular range-and-bearing sensor model. However, the approach is valid for any sensor model, as long as it is possible to determine appropriate upper bounds on the measurement and system-noise covariance matrices. For example, a holonomic kinematic model could be employed instead of the non-holonomic model in (2.20)-(2.21), and the more accurate method of evaluating the covariance of the relative position measurements of Lerro and Bar-Shalom [83] could be employed in (A.10).

Following Lemma 1, and using the above results, we obtain a constant-coefficient Riccati recursion, whose solution is an upper bound on the covariance of the position estimates in

CL:

$$\mathbf{P}^u_{k+1} = \mathbf{P}^u_k - \mathbf{P}^u_k \mathbf{H}_o^T \left(\mathbf{H}_o \mathbf{P}^u_k \mathbf{H}_o^T + \mathbf{R}_u\right)^{-1} \mathbf{H}_o \mathbf{P}^u_k + \mathbf{Q}_u \tag{2.49}$$

Intuitively, we see that this recursion describes the time evolution of the covariance of a deduced Linear Time Invariant (LTI) system, whose measurements' covariance is larger or equal to the covariance of the measurements in the actual nonlinear time-varying system. Moreover, all the parameters of this deduced system (and therefore, of the Riccati (2.49)) depend on known parameters of the actual robot team. Specifically, $\mathbf{H}_o$ describes the structure of the RPMG (cf. Section 2.4.4), $\mathbf{Q}_u$ depends on the accuracy of the robots' velocity and orientation measurements, while $\mathbf{R}_u$ depends on the accuracy and range of the robots' relative position sensors, and the structure of the RPMG.

Numerical evaluation of the solution to the recursion in (2.49) yields an upper bound on the maximum possible uncertainty of the position estimates in CL at any time instant after the deployment of the robot team. However, significant insight into the properties of the covariance matrix can be gained by evaluating the solution of (2.49) at steady state, i.e., after sufficient time has elapsed from the onset of the localization task. In order to evaluate the steady-state solution for $\mathbf{P}^u_k$, we first re-write the Riccati in a form more amenable to manipulation. We first apply the matrix inversion lemma (cf. Appendix D) to obtain

$$\mathbf{P}^u_{k+1} = \mathbf{P}^u_k \left(\mathbf{I}_{2N} + \mathbf{H}_o^T \mathbf{R}_u^{-1} \mathbf{H}_o \mathbf{P}^u_k\right)^{-1} + \mathbf{Q}_u \tag{2.50}$$

The derivations are simplified by defining the *normalized* covariance matrix as

$$\mathbf{P}_{n_k} = \mathbf{Q}_u^{-1/2} \mathbf{P}^u_k \mathbf{Q}_u^{-1/2}$$

We now obtain

$$\mathbf{P}_{n_{k+1}} = \mathbf{P}_{n_k} \left(\mathbf{I}_{2N} + \mathbf{C}_u \mathbf{P}_{n_k}\right)^{-1} + \mathbf{I}_{2N} \tag{2.51}$$

where

$$\mathbf{C}_u = \mathbf{Q}_u^{1/2} \mathbf{H}_o^T \mathbf{R}_u^{-1} \mathbf{H}_o \mathbf{Q}_u^{1/2}$$

Note that the only parameter in the above Riccati recursion is the matrix $\mathbf{C}_u$, which contains all the parameters that characterize the localization performance of the robot team. In Appendix A.5 the properties of this matrix are studied, and it is shown that it is closely related to the *Laplacian* matrix of the RPMG. The eigenvalues of this matrix depend on the type and number of exteroceptive measurements recorded by the robot team, and determine the properties of the upper bound on the steady-state positioning uncertainty. Specifically, in Appendix A.5 it is shown that when at least one robot has access to absolute positioning information, matrix $\mathbf{C}_u$ is nonsingular. In contrast, when the robots of the team record relative position measurements only, this matrix is singular, and has two eigenvalues equal to zero. We hereafter present the asymptotic uncertainty bounds for these two cases:

Observable System

If at least one of the robots receives absolute position measurements the system is observable and the covariance of the robots' position estimates remains bounded at steady state [137]. An upper bound for the steady-state covariance of CL in this case is determined by the asymptotic solution of the Riccati recursion in (2.51). This derivation is presented in Appendix A.6, and the final result is stated as a lemma:

Lemma 3 *The steady-state covariance of the position estimates for a team of robots performing CL, when at least one robot has access to absolute positioning information, is upper bounded by the matrix*

$$\mathbf{P}_{ss}^{u} \;=\; \mathbf{Q}_u^{1/2}\mathbf{U}\,\mathrm{diag}\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right)\mathbf{U}^T\mathbf{Q}_u^{1/2} \tag{2.52}$$

where we have denoted the singular value decomposition of $\mathbf{C}_u$ *as* $\mathbf{C}_u = \mathbf{U}\,\mathrm{diag}(\lambda_i)\mathbf{U}^T$.

At this point we should note that the upper bound on the steady-state uncertainty depends on the topology of the RPMG (affecting $\mathbf{C}_u$) and the accuracy of the proprioceptive and exteroceptive sensors of the robots, represented by $\mathbf{Q}_u$ and $\mathbf{R}_u$, which are "embedded"

in $\mathbf{C}_u$. However, the steady-state uncertainty is independent of the initial covariance of the robots, which comes as no surprise, since the system is observable.

Unobservable System

If none of the robots receives absolute position measurements, the system is unobservable. As a result, the steady-state uncertainty for the robots' position estimates will be a monotonically increasing function of time. The upper bound on the asymptotic covariance of the position estimates is described by the following lemma, whose proof is presented in Appendix A.6:

Lemma 4 *When none of the robots of the team has access to absolute position measurements, the asymptotic positioning uncertainty of CL is bounded above by:*

$$\mathbf{P}_{ss}^{u}(k) = k \cdot q_T \mathbf{1}_{N\times N} \otimes \mathbf{I}_2 + \mathbf{Q}_u^{1/2}\mathbf{U} \begin{bmatrix} \mathrm{diag}_{2N-2}\left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{1}{\lambda_i}}\right) & \mathbf{0}_{(2N-2)\times 2} \\ \mathbf{0}_{2\times(2N-2)} & \mathbf{0}_{2\times 2} \end{bmatrix} \mathbf{U}^T\mathbf{Q}_u^{1/2} + q_T \mathbf{1}_{N\times N} \otimes \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \tag{2.53}$$

where $\lambda_i,\ i = 1 \ldots 2N-2$ *are the non-zero singular values of* $\mathbf{C}_u$*, and* q_T *is defined as*

$$\frac{1}{q_T} = \sum_{i=1}^{N} \frac{1}{q_i} \tag{2.54}$$

and the parameters α*,* β*,* γ*,* δ *are defined as follows. Let*

$$\mathbf{W} = q_T \mathbf{Q}_u^{-1}(\mathbf{I}_{2N} + \mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\,\mathrm{diag}\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4} + \lambda_i}\right)\mathbf{U}^T\mathbf{Q}_u^{-1/2})^{-1}\mathbf{P}_0\mathbf{Q}_u^{-1}$$

Then $\alpha = \sum_{i,j \text{ odd}} w_{ij}$ *(*$\delta = \sum_{i,j \text{ even}} w_{ij}$*) is the sum of all elements of* $\mathbf{W} = [w_{ij}]$ *with two odd (even) indices and* $\gamma = \sum_{i \text{ odd,j even}} w_{ij}$ *is the sum of all elements of* $\mathbf{W} = [w_{ij}]$ *with an odd row index and an even column index. Due to symmetry,* $\beta = \gamma$*.*

Several observations can be made with respect to the above result. We note that the upper bound comprises three terms, the first of which contributes with a constant rate of uncertainty increase, equal to $q_T \delta t^{-1}$. The second term in (2.53) is a constant term, whose value depends on the topology of the RPMG and the accuracy of the sensors on the robots. Finally, the third term in (2.53) is a constant term that describes the effect of the initial uncertainty on the asymptotic values of the covariance. It also depends on the noise characteristics of the sensors of the robots, as well as the RPMG topology. The fact that the steady-state bound depends on the initial uncertainty is a consequence of the fact that the system is not observable, and therefore initial errors in the estimates for the robots' positions cannot be fully compensated for.

It is clear that the most important term in (2.53) is the one that corresponds to a constant rate of uncertainty increase. After sufficient time, this term will always dominate the remaining ones, and will largely determine the worst-case positioning performance of the team. A striking observation is that q_T, the rate of increase of the maximum uncertainty, is *independent* of both the topology of the RPMG and of the precision of the robots' relative position measurements. This quantity depends solely on the number of robots in the team, and the accuracy of the robots' dead reckoning capabilities. An intuitive interpretation of this result is that the primary factor determining the rate of uncertainty increase is the rate at which uncertainty is injected in the unobservable subspace of the system. Since the number, or the accuracy, of the relative position measurements does not alter this subspace, we should expect no change in the rate of uncertainty increase, as a result of changes in the information contributed by the exteroceptive measurements. Additional properties of the uncertainty increase rate of CL are discussed in the next section, where bounds for the expected covariance matrix are derived.

2.4.6 Upper bound on the expected covariance

The results of the preceding section enable us to determine the guaranteed accuracy of CL for a team of robots with a given set of sensors, and a specified RPMG topology. The bounds determined in (2.52) and (2.53) hold for any scenario of the robots' motion, as long as the maximum distance between any pair of them remains smaller than ρ_o. However,

it is the case for many practical scenarios, that a better characterization of the robots' trajectories is known in advance. For example, we may be able to model the pose of the robots by a known probability distribution function (pdf) in their operational area. In this case, the covariance matrices $\mathbf{Q}_r$ and $\boldsymbol{R}_{o_i}$ (cf. (2.27) and (2.33)) are functions of random variables, whose mean value can be determined. This additional knowledge, in the form of a prior distribution for the robots' poses, is used to derive an upper bound on the *expected* covariance of the position estimates, by application of Lemma 2.

The first step for applying Lemma 2 is to derive the expected values of $\mathbf{Q}_r(k+1)$ and $\boldsymbol{R}_{o_i}(k+1)$, where expectation is computed with respect to the robot poses. The average value of $\mathbf{Q}_r(k+1)$ is easily computed by averaging over all values of orientation of the robots. Assuming a uniform distribution of the robots' orientation, from (2.23) we obtain:

$$E\{\mathbf{Q}_i(k+1,k+1)\} = \delta t^2 \frac{\sigma_{V_i}^2 + \sigma_{\phi_i}^2 V_i^2}{2} \mathbf{I}_2 = \bar{q}_i \mathbf{I}_2 \tag{2.55}$$

and thus

$$\bar{\mathbf{Q}} = E\{\mathbf{Q}(k+1)\} = \mathbf{Diag}\left(\bar{q}_i \mathbf{I}_2\right), \qquad \bar{q}_i = \delta t^2 \frac{\sigma_{V_i}^2 + \sigma_{\phi_i}^2 V_i^2}{2} \tag{2.56}$$

In order to evaluate the expected value of $\mathbf{R}_o(k+1)$, we assume that the positions of the robots are modeled by a uniform[1] pdf, inside a rectangular area of side a. Using the definition of $\boldsymbol{R}_{o_i}(k+1)$ in (2.33), it is shown in Appendix A.7 that:

$$\bar{\mathbf{R}}_i = E\{\boldsymbol{R}_{o_i}\} = \left(\sigma_{\rho_i}^2 \frac{1}{2} + \sigma_{\theta_i}^2 \frac{a^2}{6} + \sigma_{\phi_i}^2 \frac{a^2}{12}\right) \mathbf{I}_{2M_i} + \sigma_{\phi_i}^2 \frac{a^2}{12} \left(\mathbf{1}_{M_i \times M_i} \otimes \mathbf{I}_2\right) \tag{2.57}$$

and thus

$$\bar{\mathbf{R}} = \begin{bmatrix} \mathbf{Diag}\left(\bar{\mathbf{R}}_i\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{Diag}\left(\mathbf{R}_{a_i}\right) \end{bmatrix} \tag{2.58}$$

Using these results, upper bounds on the expected steady-state covariance of the position

[1]The uniform distribution was employed in the calculation of $\bar{\mathbf{R}}$, since it was deemed an appropriate model for the positions of the robots in the experiments presented in Section 2.7. However, the analysis holds for any given pdf.

estimates in CL, for both the observable and unobservable case, can be derived. The solutions of the Riccati recursion in (2.19) for the two cases are completely analogous to those presented in Lemmas 3 and 4, with the sole difference that the quantities $\mathbf{Q}_u$ and $\mathbf{R}_u$ are replaced by $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$, respectively (and therefore the matrix $\mathbf{C}_u$ is also replaced by $\bar{\mathbf{C}} = \bar{\mathbf{Q}}^{1/2}\mathbf{H}_o^T\bar{\mathbf{R}}^{-1}\mathbf{H}_o\bar{\mathbf{Q}}^{1/2}$).

Some interesting remarks can be made about the uncertainty increase rate in a robot team that has no access to absolute position information. The upper bound on the expected rate of increase is equal to $\bar{q}_T\delta t^{-1}$, where

$$\frac{1}{\bar{q}_T} = \sum_{i=1}^{N} \frac{1}{\bar{q}_i} \tag{2.59}$$

We once again underline the fact that the maximum expected rate of uncertainty increase is independent of the initial uncertainty $\mathbf{P}_0$, the accuracy of the relative position measurements, and the topology of the RPMG. Moreover, we can compare this value with the rate at which uncertainty increases when each robot localizes independently, using DR. In that case, the covariance matrix for all robots' estimates evolves in time according to (2.28), and therefore the average rate of increase in uncertainty for robot i is

$$\begin{aligned} E\Big\{\frac{1}{\delta t}\left(\mathbf{P}_{i_{k+1}} - \mathbf{P}_{i_k}\right)\Big\} &= E\Big\{\frac{1}{\delta t}\mathbf{Q}_i(k,k)\Big\} = \frac{\bar{q}_i}{\delta t}\mathbf{I}_2 \Rightarrow \\ E\Big\{\mathbf{P}_{i_{k+1}}\Big\} &= E\Big\{\mathbf{P}_{i_k}\Big\} + \bar{q}_i\mathbf{I}_2 \end{aligned} \tag{2.60}$$

From the definition of $\bar{q}_T$ (cf. (2.59)), it becomes clear that it will be smaller than the smallest of the $\bar{q}_i$'s (notice that the definition of $\bar{q}_T$ is analogous to the expression for the total resistance of resistors in parallel). This implies that it suffices to equip only one robot in the team with proprioceptive sensors of high accuracy, in order to achieve a desired rate of uncertainty increase. All the robots of the group will experience a reduction in the rate at which their uncertainty increases and this improvement is more significant for robots with sensors of poor quality. Moreover, the maximum expected rate of uncertainty increase is identical for all robots of the team, regardless of the accuracy of each robot's odometry.

Corollary 5 *The maximum expected rate of positioning uncertainty increase of all the ro-*

bots of a heterogeneous team performing CL is the same, equal to $\bar{q}_T \delta t^{-1}$*, where*

$$\frac{1}{\bar{q}_T} = \sum_{i=1}^{N} \frac{1}{\bar{q}_i} \geq \max\left(\frac{1}{\bar{q}_i}\right) \Rightarrow \bar{q}_T \leq \min \bar{q}_i \tag{2.61}$$

This rate is smaller than the rate of uncertainty increase of the robot with the best DR performance, if it were to localize independently.

In the following section we study some important properties of the derived upper bounds.

2.4.7 RPMG reconfigurations

In the preceding analysis, it is assumed that the topology of the graph describing the relative position measurements between robots does not change. However, this may be difficult to implement in a realistic scenario. For example, due to the robots' motion or because of obstacles in the environment, some robots may not be able to measure their relative positions. Additionally, robot teams often need to allocate computational and communication resources to mission-specific tasks and this may force them to reduce the number of measurements they process for localization purposes. Consequently, it is of considerable interest to study the effects of changes in the topology of the RPMG on the localization accuracy of the team.

Consider the following scenario: At the initial stage of the deployment of a robot team, the RPMG has a dense topology $\mathcal{T}_A$, e.g., the complete graph shown in Fig. 2.6(a), and retains this topology until some time instant t_1, when it assumes a sparser topology $\mathcal{T}_B$, e.g., the ring graph shown in Fig. 2.6(b). This sparse topology may even be an *empty graph*, i.e., the case in which the robots localize independently, based only on DR. Subsequent topology changes are assumed to occur at time instants $t_i, i = 1 \ldots n-1$, and finally, at time instant t_n, the RPMG returns to its initial, dense topology, $\mathcal{T}_A$. Assuming that the time intervals $[t_{i-1}, t_i]$ are of sufficient duration for the transient phenomena in the time evolution of uncertainty to subside, the following lemma applies.

Lemma 6 *Once the RPMG resumes its initial topology after a sequence of reconfigurations, the maximum positioning uncertainty of the robots at steady state is identical to the one the*

robot team would have had if no RPMG reconfigurations had taken place.

Proof: See Appendix A.8. ∎

This implies that during time intervals when the RPMG topology is a sparse one, the "additional uncertainty" is introduced in directions of the state space that belong to the observable subspace. Thus, when the topology resumes its initial dense form, this additional uncertainty vanishes.

This is a significant result due to its practical implications. Consider the scenario where the robots of a team, during a phase of their mission, are forced to receive and process a small number of measurements, or even resort to mere DR, due to communication or sensor failures, or because CPU and bandwidth resources are required by other tasks of higher priority. During this interval, a reduced amount of positioning information is available to the robots (sparse RPMG topology) and as a result the performance of CL will temporarily deteriorate. However, once the initial, dense RPMG topology is restored, the team's positioning performance will have sustained no degradation. Furthermore, Lemma 6 indicates that a dense topology for the RPMG during the initial phase of the deployment of a robot team has a long-term effect on the localization performance of the team. Specifically, if during the initial deployment, the robots leverage their communication and computational resources to support a dense RPMG, this will improve their positioning accuracy at the beginning of CL. Later on, and as the robots focus on mission-specific and other time-critical tasks, they will have to rely on sparser RPMGs as resources dictate. However, when at a subsequent time instant the RPMG resumes its initial, dense topology, the above lemma states that the guaranteed accuracy will be identical to the one that would result if the dense RPMG topology was retained throughout the run of the robots.

2.5 Cooperative SLAM

The previous section presented our analysis of the positioning accuracy during CL, where a group of robots update their estimates using relative position and (possibly) absolute-position measurements. We now address the case in which the robots localize while simultaneously estimating the positions of M static landmarks in the environment. This is the

case of C-SLAM [104,108]. The formulation of the problem bears similarities to that of CL: We consider N robots moving in 2D, each equipped with an odometric sensor, allowing it to measure its translational velocity, and an orientation sensor, allowing it to measure its global orientation. Moreover, some (or all) the robots are capable of measuring the relative positions of other robots or landmarks. The RPMG is once again employed for describing the measurements occurring at every time instant in the system, with the difference that now some of the nodes of this graph are landmarks, instead of robots (cf. Fig.2.7(a)).

Our approach for determining upper bounds on the localization uncertainty is analogous to that followed in the case of CL. Specifically, we formulate a suboptimal, position-only EKF estimator for the system of robots and landmarks, and subsequently compute the guaranteed accuracy of the position estimates produced by this estimator, to obtain the desired bounds.

2.5.1 The Riccati recursion

The state vector for the C-SLAM system at time-step k is:

$$\mathbf{X}(k) = \begin{bmatrix} \mathbf{p}_{r_1}^T(k) & \cdots & \mathbf{p}_{r_N}^T(k) & \mathbf{p}_{L_1}^T & \cdots & \mathbf{p}_{L_M}^T \end{bmatrix}^T \tag{2.62}$$

where $\mathbf{p}_{r_i}(k)$, $i = 1, \ldots, N$ are the positions of the N robots, and $\mathbf{p}_{L_i}$, $i = 1, \ldots, M$ are the positions of the M landmarks. The state propagation equation, for each of the N robots, is identical to that presented in Section 2.4.1. On the other hand, the landmarks are modeled as static points in 2D space, and therefore the state propagation equations for them are

$$\mathbf{p}_{L_i}(k+1) = \mathbf{p}_{L_i}(k), \quad \text{for} \quad i = 1 \ldots M$$

Similarly, the estimates for the landmark positions are propagated using the relations

$$\hat{\mathbf{p}}_{L_i k+1|k} = \hat{\mathbf{p}}_{L_i k|k}, \quad \text{for} \quad i = 1 \ldots M$$

while the errors are propagated by

$$\widetilde{\mathbf{p}}_{L_i k+1|k} = \widetilde{\mathbf{p}}_{L_i k|k}, \quad \text{for} \quad i = 1 \ldots M$$

Hence, the error-state propagation equation is:

$$\widetilde{\mathbf{X}}_{k+1|k} = \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{G}\mathbf{w}(k) \tag{2.63}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_{2N} \\ \mathbf{0}_{2M\times 2N} \end{bmatrix}$$

and $\mathbf{w}(k)$ is a noise vector with covariance matrix $\mathbf{Q}_r(k)$ (cf. (2.27)). The equation for propagating the covariance matrix of the state error is written as:

$$\mathbf{P}_{k+1|k} = \mathbf{P}_{k|k} + \mathbf{G}\mathbf{Q}_r(k)\mathbf{G}^T \tag{2.64}$$

At every time-step, the robots perform robot-to-robot and robot-to-landmark relative position measurements. The relative position measurement between robots r_i and r_m is given by:

$$\mathbf{z}_{r_i r_m} = \mathbf{C}^T(\phi_i)\left(\mathbf{p}_{r_m} - \mathbf{p}_{r_i}\right) + \mathbf{n}_{z_{r_i r_m}} \tag{2.65}$$

where r_i (r_m) is the observing (observed) robot, and $\mathbf{n}_{z_{r_i r_m}}$ is the noise affecting this measurement. Similarly, the measurement of the relative position between the robot r_i and landmark L_n is given by:

$$\mathbf{z}_{r_i L_n} = \mathbf{C}^T(\phi_i)\left(\mathbf{p}_{L_n} - \mathbf{p}_{r_i}\right) + \mathbf{n}_{z_{r_i L_n}} \tag{2.66}$$

Both of these equations adhere to the form of the measurement model encountered in the case of CL (cf. (2.29)). As a result, the derivation of the covariance update equation is analogous to that presented in Section 2.4.2, and the resulting expression is:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}_o^T\left(\mathbf{H}_o\mathbf{P}_{k+1|k}\mathbf{H}_o^T + \mathbf{R}_o(k+1)\right)^{-1}\mathbf{H}_o\mathbf{P}_{k+1|k} \tag{2.67}$$

where the matrix $\mathbf{H}_o$ is once again closely related to the structure of the RPMG:

$$\mathbf{H}_o = \mathbf{A}_{\text{RPMG}} \otimes \mathbf{I}_2 \tag{2.68}$$

while $\mathbf{R}_o(k+1)$ is a block-diagonal matrix with block elements $\boldsymbol{R}_{o_i}(k+1)$, $i = 1 \dots N$ (cf. (2.33)).

By combining the covariance propagation and update equations, the Riccati recursion describing the time evolution of the position uncertainty is obtained:

$$\mathbf{P}_{k+1} = \mathbf{P}_k - \mathbf{P}_k \mathbf{H}_o^T \left(\mathbf{H}_o \mathbf{P}_k \mathbf{H}_o^T + \mathbf{R}_o(k+1) \right)^{-1} \mathbf{H}_o \mathbf{P}_k + \mathbf{G} \mathbf{Q}_r(k+1) \mathbf{G}^T \quad (2.69)$$

The initial value of this recursion is assumed to be equal to

$$\mathbf{P}_0 = \begin{bmatrix} \mathbf{P}_{rr_0} & \mathbf{0}_{2N \times 2M} \\ \mathbf{0}_{2M \times 2N} & \mathbf{P}_{LL_0} \end{bmatrix} \quad (2.70)$$

i.e., we assume that the position estimates for the robots and the map features are initially uncorrelated, which is the case at the onset of a mapping task within an unknown area.

We note that the Riccati recursion in (2.69) is very similar to that obtained in the case of CL. However, it should be pointed out that the covariance matrix in this case describes the position uncertainty of both the robots and landmarks, and is of dimensions $(2N + 2M) \times (2N + 2M)$, while in the case of CL, the covariance matrix is of dimensions $2N \times 2N$, since it pertains to the robots' positions only. Moreover, it should be pointed out that during propagation, in the case of CL the uncertainty of all states increases (since all robots are moving), while in the case of C-SLAM the uncertainty of the landmarks remains unchanged during propagation. As a result of these differences, the properties of the covariance in C-SLAM are quite different than those in CL, as shown in the next section.

2.5.2 Upper bound on the worst-case covariance

Application of Lemma 1 enables us to determine the guaranteed accuracy of localization. In particular, the solution of the following constant-coefficient Riccati is an upper bound to the covariance of the C-SLAM position estimates:

$$\mathbf{P}_{k+1}^u = \mathbf{P}_k^u - \mathbf{P}_k^u \mathbf{H}_o^T \left(\mathbf{H}_o \mathbf{P}_k^u \mathbf{H}_o^T + \mathbf{R}_u \right)^{-1} \mathbf{H}_o \mathbf{P}_k^u + \mathbf{G} \mathbf{Q}_u \mathbf{G}^T \quad (2.71)$$

where $\mathbf{Q}_u$ and $\mathbf{R}_u$ are given in (2.47) and (2.48), respectively.

While the above recursion provides us with the guaranteed accuracy for any time instant after the robots' deployment, it is interesting to evaluate the performance of C-SLAM at steady state, i.e., when the covariance of the map has converged to a constant value [41]. To this end, it is necessary to compute the solution to the recursion in (2.71) after sufficient time, i.e., as $k \to \infty$. This computation is simplified by employing the following lemma, adapted from [58]:

Lemma 7 *Suppose* $\mathbf{P}_{k+1}^{u(0)}$ *is the solution to the discrete-time Riccati recursion in (2.71) with initial value* $\mathbf{P}_0^u = \mathbf{0}_{\kappa\times\kappa}$*, where* $\kappa = 2M + 2N$*. Then the solution to the same Riccati recursion but with an arbitrary initial condition,* $\mathbf{P}_0$*, is determined by the identity*

$$\mathbf{P}_{k+1}^{u} - \mathbf{P}_{k+1}^{u(0)} = \mathbf{\Psi}(0, k+1)\left(\mathbf{I}_\kappa + \mathbf{P}_0\mathbf{J}_{k+1}\right)^{-1}\mathbf{P}_0\mathbf{\Psi}^T(0, k+1)$$

where

$$\mathbf{\Psi}(0, k+1) = \left(\mathbf{I}_\kappa - \mathbf{K}_p\mathbf{H}_o\right)^{k+1}\left(\mathbf{I}_\kappa + \mathbf{P}\mathbf{J}_{k+1}\right)$$

In these expressions, $\mathbf{P}$ *is any solution to the Discrete Algebraic Riccati Equation (DARE):*

$$\mathbf{P} = \mathbf{P} - \mathbf{P}\mathbf{H}_o^T(\mathbf{H}_o\mathbf{P}\mathbf{H}_o^T + \mathbf{R}_u)^{-1}\mathbf{H}_o\mathbf{P} + \mathbf{G}\mathbf{Q}_u\mathbf{G}^T,$$

and $\mathbf{K}_p = \mathbf{P}\mathbf{H}_o^T\left(\mathbf{H}_o\mathbf{P}\mathbf{H}_o^T + \mathbf{R}_u\right)^{-1}$*. Finally,* $\mathbf{J}_{k+1}$ *denotes the solution to the* dual *Riccati recursion:*

$$\mathbf{J}_{k+1} = \mathbf{J}_k - \mathbf{J}_k\mathbf{G}(\mathbf{G}^T\mathbf{J}_k\mathbf{G} + \mathbf{Q}_u^{-1})^{-1}\mathbf{G}^T\mathbf{J}_k + \mathbf{H}_o^T\mathbf{R}_u^{-1}\mathbf{H}_o$$

with zero initial condition, $\mathbf{J}_0 = \mathbf{0}_{\kappa\times\kappa}$*.*

Lemma 7 simplifies the evaluation of the steady-state value of $\mathbf{P}_{k+1}^u$, since the solution to the Riccati recursion with zero initial condition can be easily derived. We note that (2.71) describes the time evolution of the covariance for a deduced C-SLAM scenario, in which the robots' kinematic equations and measurement equations are time invariant. Zero initial co-

variance of the landmarks' position estimates corresponds to a perfectly known map. In this case the robots essentially perform cooperative localization, while the robot-to-landmark measurements are equivalent to absolute position measurements. Thus, the resulting system is observable, and the steady-state solution to the Riccati equation is of the form:

$$\mathbf{P}_{\infty}^{u(0)} = \begin{bmatrix} \mathbf{P}_{rr_\infty}^{u} & \mathbf{0}_{2N\times 2M} \\ \mathbf{0}_{2M\times 2N} & \mathbf{0}_{2M\times 2M} \end{bmatrix} \tag{2.72}$$

where the term $\mathbf{P}_{rr_\infty}^{u}$ is given by [102]:

$$\mathbf{P}_{rr_\infty}^{u} = \mathbf{Q}_u^{1/2}\mathbf{U}\,\mathrm{diag}\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right)\mathbf{U}^T\mathbf{Q}_u^{1/2}$$

In the last expression, the quantities $\mathbf{U}$ and λ_i, $i = 1\ldots 2N$ are defined as the matrix of eigenvectors and the eigenvalues of $\mathbf{C}_u = \mathbf{Q}_u^{1/2}\mathbf{Y}_r\mathbf{Q}_u^{1/2}$ respectively, where

$$\mathbf{Y}_r \;=\; [\mathbf{I}_{2N}\;\; \mathbf{0}_{2N\times 2M}]\,\mathbf{H}_o^T\mathbf{R}_u^{-1}\mathbf{H}_o \begin{bmatrix} \mathbf{I}_{2N} \\ \mathbf{0}_{2M\times 2N} \end{bmatrix}$$

It is interesting to note that $\mathbf{H}_o^T\mathbf{R}_u^{-1}\mathbf{H}_o$ represents the information matrix associated with the measurements in the deduced linear time-invariant system, and $\mathbf{Y}_r$ is the submatrix expressing the information about the robots' positions.

The rest of the derivations for the upper bound on the steady-state covariance matrix involve only algebraic manipulations, and are not included here, since they offer only little intuition about the properties of the problem. The interested reader is referred to [102] for the details of the intermediate steps. The final result is stated in the following lemma:

Lemma 8 *The worst-case asymptotic covariance matrix in C-SLAM is bounded above by the matrix*

$$\mathbf{P}_{\infty}^{u} \;=\; \mathbf{P}_{\infty}^{u(0)} + \mathbf{1}_{(N+M)\times(N+M)} \otimes \mathbf{\Theta}^{-1} \tag{2.73}$$

where $\mathbf{P}_{\infty}^{u(0)}$ *is defined in (2.72) and*

$$\mathbf{\Theta} = (\mathbf{1}_{1\times M} \otimes \mathbf{I}_2)\, \mathbf{P}_{LL_0}^{-1}\, (\mathbf{1}_{M\times 1} \otimes \mathbf{I}_2) + (\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\, \left(\mathbf{J}_{rr_\infty}^{-1} + \mathbf{P}_{rr_0}\right)^{-1} (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2) \quad (2.74)$$

with

$$\mathbf{J}_{rr_\infty} = \mathbf{Q}_u^{-1/2}\mathbf{U}\, \mathrm{diag}\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4} + \lambda_i}\right) \mathbf{U}^T\mathbf{Q}_u^{-1/2} \quad (2.75)$$

Note that the first term in (2.73) depends only on the RPMG and the accuracy of the robots' sensors, while the second term also encapsulates the effect of the initial uncertainty. Additionally, it is worth mentioning that only the second term affects the accuracy of the computed map, while both terms determine the localization accuracy of the robots.

A case of particular interest in C-SLAM is that of a robot team building a map of an area for which no prior knowledge exists. We model this scenario by setting $\mathbf{P}_{LL_0} = \mu\mathbf{I}_{2M}$, with $\mu \to \infty$, which yields the following simplified expression for matrix $\mathbf{\Theta}$:

$$\mathbf{\Theta} = (\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\, \left(\mathbf{J}_{rr_\infty}^{-1} + \mathbf{P}_{rr_0}\right)^{-1} (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)$$

If additionally, the initial position of all robots is perfectly known, which is often the case in many mapping applications, the previous equation further simplifies to

$$\mathbf{\Theta} = (\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\, \mathbf{J}_{rr_\infty}\, (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)$$

Clearly, the topology of the RPMG has a significant impact on the positioning accuracy of the robots and the quality of the map. As one would expect, for a given set of robots and landmarks, the λ_i's (cf. (2.75)) receive their lowest values when the RPMG corresponds to a dense graph, and higher values for sparser graphs [22]. Intuitively, the best C-SLAM results are obtained for groups of robots that can detect all landmarks and all other robots at every time instant. Finally, (2.74) indicates that the eigenvalues of the RPMG also affect the way the initial uncertainty is diffused via the C-SLAM process.

2.5.3 Upper bound on the expected covariance

If additional information about the spatial distribution of the robots and landmarks is available, then application of Lemma 2 enables us to compute an upper bound on the average position uncertainty during C-SLAM. The process is completely analogous to the derivation of an upper bound on the worst-case uncertainty, presented in the previous section. We only state the final result:

Lemma 9 *The expected steady-state covariance of the position estimates in C-SLAM, when the spatial density of landmarks is described by a known pdf, is bounded above by the matrix*

$$\bar{\mathbf{P}}_{\infty} = \begin{bmatrix} \bar{\mathbf{P}}_{rr_{\infty}} & \mathbf{0}_{2N\times 2M} \\ \mathbf{0}_{2M\times 2N} & \mathbf{0}_{2M\times 2M} \end{bmatrix} + \mathbf{1}_{(N+M)\times(N+M)} \otimes \bar{\mathbf{\Theta}}^{-1}$$

with

$$\bar{\mathbf{P}}_{rr_{\infty}} = \bar{\mathbf{Q}}^{1/2}\bar{\mathbf{U}}\,\mathrm{diag}\left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{1}{\bar{\lambda}_i}}\right)\bar{\mathbf{U}}^T\bar{\mathbf{Q}}^{1/2}$$

and

$$\bar{\mathbf{\Theta}} \;=\; (\mathbf{1}_{1\times M}\otimes\mathbf{I}_2)\,\mathbf{P}_{LL_0}^{-1}\,(\mathbf{1}_{M\times 1}\otimes\mathbf{I}_2) + (\mathbf{1}_{1\times N}\otimes\mathbf{I}_2)\left(\bar{\mathbf{J}}_{rr_{\infty}}^{-1} + \mathbf{P}_{rr_0}\right)^{-1}(\mathbf{1}_{N\times 1}\otimes\mathbf{I}_2)$$

The quantity $\bar{\mathbf{J}}_{rr_{\infty}}$ *appearing in this last expression is:*

$$\bar{\mathbf{J}}_{rr_{\infty}} = \bar{\mathbf{Q}}^{-1/2}\bar{\mathbf{U}}\,\mathrm{diag}\left(\frac{\bar{\lambda}_i}{2} + \sqrt{\frac{\bar{\lambda}_i^2}{4} + \bar{\lambda}_i}\right)\bar{\mathbf{U}}^T\bar{\mathbf{Q}}^{-1/2} \tag{2.76}$$

where $\bar{\mathbf{U}}$ *and* $\bar{\lambda}_i$*,* $i = 1\ldots 2N$ *are defined based on the singular value decomposition of the matrix* $\bar{\mathbf{Y}}_r$*:*

$$\begin{aligned} \bar{\mathbf{Y}}_r &= [\mathbf{I}_{2N} \;\; \mathbf{0}_{2N\times 2M}]\,\mathbf{H}_o^T\bar{\mathbf{R}}^{-1}\mathbf{H}_o \begin{bmatrix} \mathbf{I}_{2N} \\ \mathbf{0}_{2M\times 2N} \end{bmatrix} \\ &= \bar{\mathbf{U}}\,\mathrm{diag}(\bar{\lambda}_i)\bar{\mathbf{U}}^T \end{aligned}$$

The matrices $\bar{\mathbf{R}}$ and $\bar{\mathbf{Q}}$ appearing in the equations of the above lemma are defined, similarly to the case of CL, as:

$$\bar{\mathbf{R}} = E\{\mathbf{R}_o(k)\} = \mathbf{Diag}(\bar{\mathbf{R}}_i), \quad \text{and} \quad \bar{\mathbf{Q}} = E\{\mathbf{Q}_r(k)\} = \mathbf{Diag}(\bar{q}_i\mathbf{I}_2) \tag{2.77}$$

2.6 Single-robot SLAM

The results of the preceding section, which provide performance guarantees for C-SLAM, encompass, as a special case, the situation where a single robot localizes using SLAM [101]. This is a very important special case, which has been at the focus of robotics research for more than a decade. The results of Lemmas 8 and 9 enable us to characterize the performance of SLAM with a single robot, when an upper bound on the orientation errors' variance can be determined *a priori*. Such a bound was obtained in the preceding analysis, by considering the accuracy of the robot's orientation sensor. However, it is often the case that global orientation measurements are impossible to obtain or unreliable (e.g., compasses often suffer from electrical interference in urban environments). For this reason, it is important to relax the requirement that an absolute orientation sensor is available.

In what follows, we present an analysis of the performance of single-robot SLAM, and study the estimation accuracy attainable without absolute orientation measurements [105]. When the variance of the orientation errors of the robot cannot be determined in advance, the decoupling of the position and orientation estimators, which was instrumental in the analysis for CL and C-SLAM, is no longer valid. If, following the "standard" EKF-SLAM approach [149], we formulate a state vector comprising the robot's position and orientation, as well as the global position of the M landmarks, the resulting Riccati recursion has the form:

$$\mathbf{P}_{k+1} = \mathbf{\Phi}(k)\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T(k)\left(\mathbf{H}(k)\mathbf{P}_k\mathbf{H}^T(k) + \mathbf{R}(k+1)\right)^{-1}\mathbf{H}(k)\mathbf{P}_k\right)\mathbf{\Phi}^T(k) + \mathbf{Q}(k+1) \tag{2.78}$$

where $\mathbf{P}_{k+1}$ is a $(3+2M)\times(3+2M)$ covariance matrix. In this formulation, the state-transition matrix, $\mathbf{\Phi}(k)$, is not equal to the identity matrix, as was the case for the position-only estimators for CL and C-SLAM (cf. 2.28 and (2.64)). Moreover, the measurement

Jacobian matrix $\mathbf{H}(k)$ is time varying, and a suitable factorization into a constant and a time-varying matrix (cf. (2.35)) cannot be derived. Due to the particular structure of the matrices $\mathbf{\Phi}(k)$ and $\mathbf{H}(k)$, the recursion of (2.78) is not amenable to an analysis similar to the one presented in the preceding sections.

In order to circumvent this problem, we introduce a novel formulation of SLAM, which we term the *dual-map filter*. Specifically, instead of employing a state vector comprising the pose of the robot and the global landmark coordinates, we formulate a state vector that consists of both the *global* and the *relative*[2] coordinates of the landmarks. The basic idea here is that the robot pose is not explicitly included in the state of the EKF, but it is implicitly defined from the transformation between the global and the robot-relative map. This formulation of SLAM is equivalent to the standard one, and therefore, by studying the covariance properties of the dual-map filter, we can derive conclusions for the covariance in the standard EKF-based SLAM. The details of this analysis are presented in what follows.

2.6.1 The dual-map filter

The state vector of the dual-map filter is defined as

$$\mathbf{X}(k) = \begin{bmatrix} {}^{R}\mathbf{X}(k)^T & {}^{G}\mathbf{X}^T \end{bmatrix}^T$$

where ${}^{R}\mathbf{X}$ and ${}^{G}\mathbf{X}$ are $2M \times 1$ vectors, comprising the positions of M landmarks in the robot and and global frames, respectively. We will refer to these as the *relative map* and *global map*, respectively. It is worth noting that the computational complexity of this formulation is higher than that of the "standard" extended Kalman filter (EKF), whose state vector contains the robot pose and the global landmark coordinates. However, the sole purpose of employing such a formulation of SLAM is to determine analytical upper bounds for the covariance of the state estimates. As will be made clear in the following, in the dual-map EKF all available measurements are used, and apart from linearization, no other approximations are made. Therefore, the covariance of the absolute map computed

[2]The term relative coordinates refers to the coordinates of the landmarks expressed with respect to a frame attached on the robot. This frame moves along with the robot, and thus the relative coordinates of the landmarks change in time, even though their global coordinates remain constant. This is different than the notion of the relative map employed, for example, in [118].

with this filter will be identical (except for small linearization inaccuracies) to the covariance that is computed with the traditional EKF SLAM algorithm [149].

Initialization

Here, we assume that the landmark positions are unknown prior to the first observation, and the robot has perfect knowledge of its initial pose, which is the most common setting for SLAM. Following standard practice, we select the global coordinate frame to coincide with the initial robot frame. Thus, immediately after the first set of robot-to-landmark measurements (at time-step 0), the absolute and relative maps coincide, i.e., ${}^{G}\mathbf{X} = {}^{R}\mathbf{X}_0$. Moreover, the uncertainty of both the relative and the absolute map is equal to the covariance matrix of the exteroceptive measurement used for initialization, and the two maps are fully correlated. The initial covariance matrix of the dual-map filter is thus given by

$$\mathbf{P}_{0|0} = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_0 \\ \mathbf{R}_0 & \mathbf{R}_0 \end{bmatrix} \tag{2.79}$$

where $\mathbf{R}_0$ is the covariance matrix of the robot-to-landmark measurement used for initialization.

Propagation

As the robot moves, the relative-map coordinates of all landmarks change, while the global ones remain constant. Thus, during the EKF propagation phase, the relative-map estimates and the filter covariance matrix must be updated. If we denote the position of the i-th landmark with respect to the robot at time-step k by ${}^{R_k}\mathbf{p}_i$, $i = 1 \dots M$, we obtain the following propagation equation:

$${}^{R_{k+1}}\mathbf{p}_i = {}^{R_{k+1}}\mathbf{p}_{R_k} + \mathbf{C}(-\omega_k \delta t)\,{}^{R_k}\mathbf{p}_i \tag{2.80}$$

where the rotation matrix expressing the rotation of the robot frame between time-steps $k+1$ and k is:

$$\mathbf{C}(-\omega_k \delta t) = \begin{bmatrix} \cos(\omega_k \delta t) & \sin(\omega_k \delta t) \\ -\sin(\omega_k \delta t) & \cos(\omega_k \delta t) \end{bmatrix} \tag{2.81}$$

and ${}^{R_{k+1}}\mathbf{p}_{R_k}$ is the position of the robot at time-step k, expressed with respect to the robot frame at time-step $k+1$:

$${}^{R_{k+1}}\mathbf{p}_{R_k} = -\mathbf{C}(-\omega_k \delta t)\,{}^{R_k}\mathbf{p}_{R_{k+1}} = -V_k \delta t \mathbf{C}(-\omega_k \delta t)\mathbf{e}_1 \tag{2.82}$$

In the preceding expressions, V_k and ω_k are the translational and rotational velocity of the robot at time-step k, respectively, δt is the sampling interval, and $\mathbf{e}_1 = [1 \;\; 0]^T$.

Using the measurements of the robot's translational and rotational velocities, V_{m_k} and ω_{m_k}, respectively, the position estimate of the i-th landmark is propagated according to:

$$\begin{aligned} {}^{R_{k+1}}\widehat{\mathbf{p}}_i &= {}^{R_{k+1}}\widehat{\mathbf{p}}_{R_k} + \mathbf{C}(-\omega_{m_k}\delta t)\,{}^{R_k}\widehat{\mathbf{p}}_i \\ &= \mathbf{C}(-\omega_{m_k}\delta t)\left(-V_{m_k}\delta t \mathbf{e}_1 + {}^{R_k}\widehat{\mathbf{p}}_i\right) \end{aligned} \tag{2.83}$$

By linearizing (2.80), we obtain the error-propagation equation for the relative position of the i-th landmark:

$$\begin{aligned} {}^{R_{k+1}}\widetilde{\mathbf{p}}_i &= \mathbf{C}(-\omega_{m_k}\delta t)\,{}^{R_k}\widetilde{\mathbf{p}}_i + \delta t \mathbf{C}(-\omega_{m_k}\delta t)\mathbf{e}_1 \widetilde{V}_k - \delta t \mathbf{J}\,{}^{R_{k+1}}\widehat{\mathbf{p}}_i \widetilde{\omega}_k \\ &= \mathbf{C}(-\omega_{m_k}\delta t)\,{}^{R_k}\widetilde{\mathbf{p}}_i + \underbrace{\delta t \begin{bmatrix} \mathbf{C}(-\omega_{m_k}\delta t)\mathbf{e}_1 & -\mathbf{J}\,{}^{R_{k+1}}\widehat{\mathbf{p}}_i \end{bmatrix}}_{{}^{R}\mathbf{G}_{i_k}} \underbrace{\begin{bmatrix} \widetilde{V}_k \\ \widetilde{\omega}_k \end{bmatrix}}_{\mathbf{n}_{\text{od}}} \end{aligned} \tag{2.84}$$

The vector $\mathbf{n}_{\text{od}}$ is the noise of the robot's odometry measurements, assumed to be zero-mean, white and Gaussian, with covariance matrix $\mathbf{Q} = \text{diag}(\sigma_v^2, \sigma_\omega^2)$. Based on (2.84), and taking into account the fact that the global landmark coordinates do not change during propagation, the linearized error-state propagation equation for the dual-map filter is

derived as:

$$\begin{aligned}\widetilde{\mathbf{X}}_{k+1|k} &= \underbrace{\begin{bmatrix} {}^{R}\mathbf{\Phi}_k & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix}}_{\mathbf{\Phi}_k} \widetilde{\mathbf{X}}_{k|k} + \underbrace{\begin{bmatrix} \mathbf{I}_{2M} \\ \mathbf{0}_{2M\times 2M} \end{bmatrix}}_{\mathbf{G}} {}^{R}\mathbf{G}_k \mathbf{n}_{\text{od}} \\ &= \mathbf{\Phi}_k \widetilde{\mathbf{X}}_{k|k} + \mathbf{G}\, {}^{R}\mathbf{G}_k \mathbf{n}_{\text{od}} \end{aligned} \tag{2.85}$$

where

$${}^{R}\mathbf{\Phi}_k = \mathbf{I}_M \otimes \mathbf{C}(-\omega_{m_k}\delta t) \tag{2.86}$$

and ${}^{R}\mathbf{G}_k$ is a $2M \times 2$ block matrix, whose i-th element is

$${}^{R}\mathbf{G}_{i_k} = \delta t \begin{bmatrix} -\mathbf{C}(-\omega_{m_k}\delta t)\mathbf{e}_1 & -\mathbf{J}^{R_{k+1}}\widehat{\mathbf{p}}_i \end{bmatrix} \tag{2.87}$$

The covariance propagation equation for the dual-map filter is

$$\mathbf{P}_{k+1|k} = \mathbf{\Phi}_k \mathbf{P}_{k|k} \mathbf{\Phi}_k^T + \mathbf{G}\, {}^{R}\mathbf{Q}_k \mathbf{G}^T \tag{2.88}$$

where we have denoted

$${}^{R}\mathbf{Q}_k = {}^{R}\mathbf{G}_k \mathbf{Q}\, {}^{R}\mathbf{G}_k^T \tag{2.89}$$

Update

At every time-step, the robot performs a direct observation of the relative positions of all landmarks, and therefore the measurement vector at each time-step is described by

$$\mathbf{z}(k) = \mathbf{H}\mathbf{X}(k) + \mathbf{n}(k), \quad \text{with} \quad \mathbf{H} = \begin{bmatrix} \mathbf{I}_{2M} & \mathbf{0}_{2M\times 2M} \end{bmatrix} \tag{2.90}$$

where $\mathbf{n}(k)$ is a Gaussian, zero-mean, white noise vector. The measurements of different landmarks are independent, and therefore the covariance matrix of $\mathbf{n}(k)$ will generally be a

time-varying, block-diagonal matrix:

$$\mathbf{R}_k \;=\; \mathbf{Diag}(\mathbf{R}_{i_k}) \tag{2.91}$$

where $\mathbf{R}_{i_k}$ is the 2×2 covariance matrix of the measurement of the i-th landmark. Using these definitions, we can write the covariance update equation of the EKF as:

$$\mathbf{P}_{k+1|k+1} \;=\; \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}^T \left(\mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1} \mathbf{H}\mathbf{P}_{k+1|k} \tag{2.92}$$

It is interesting to note that in the dual-map filter formulation, the measurement matrix $\mathbf{H}$ is constant, and this result is exact (i.e., it does not involve linearization). This fact, which facilitates the analytical study of the time evolution of the covariance matrix, was our main motivation for introducing the dual-map filter. We also note that, even though the measurement equation does not directly involve the absolute position estimates of the landmarks, the correlations that exist between the relative- and absolute-map estimates ensure that the absolute map is appropriately updated every time a new measurement is processed.

The Riccati recursion

By combining the covariance propagation and update equations (cf. (2.88) and (2.92)), we form the Riccati recursion that describes the time evolution of the state covariance matrix. This is given by:

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_k \left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T \left(\mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1} \mathbf{H}\mathbf{P}_k\right) \mathbf{\Phi}_k^T + \mathbf{G}\,{}^R\mathbf{Q}_k\mathbf{G}^T \tag{2.93}$$

where we have introduced the substitutions $\mathbf{P}_k = \mathbf{P}_{k|k-1}$ and $\mathbf{P}_{k+1} = \mathbf{P}_{k+1|k}$ to simplify the notation. In the next section, we show how an upper bound on the asymptotic covariance can be computed, by applying Lemma 1.

2.6.2 Upper bounds on the asymptotic map covariance

In order to apply Lemma 1, we first need to compute appropriate upper bounds for the matrices ${}^R\mathbf{Q}_k$, $\mathbf{R}_k$, and $\mathbf{P}_0$. We start by rewriting the system noise covariance matrix ${}^R\mathbf{Q}_k$, as (cf. (2.85), (2.87), and (2.89)):

$$ {}^R\mathbf{Q}_k = \underbrace{\delta t^2 \sigma_v^2 \cdot {}^R\mathbf{\Phi}_k \mathbf{v}\mathbf{v}^T ({}^R\mathbf{\Phi}_k)^T}_{\mathbf{Q}_{v_k}} + \underbrace{\delta t^2 \sigma_\omega^2 \cdot (\mathbf{I}_M \otimes \mathbf{J}) {}^R\mathbf{X}(k) {}^R\mathbf{X}(k)^T (\mathbf{I}_M \otimes \mathbf{J})^T}_{\mathbf{Q}_{\omega_k}} \tag{2.94} $$

where

$$ \mathbf{v} = \begin{bmatrix} 1 & 0 & 1 & 0 & \cdots \end{bmatrix}^T = \mathbf{1}_{M\times 1} \otimes \mathbf{e}_1 $$

We now consider each of the components of ${}^R\mathbf{Q}_k$ independently. For $\mathbf{Q}_{v_k}$ we obtain:

$$ \mathbf{Q}_{v_k} = \delta t^2 \sigma_v^2 \cdot {}^R\mathbf{\Phi}_k \mathbf{v}\mathbf{v}^T ({}^R\mathbf{\Phi}_k)^T \tag{2.95} $$

$$ \preceq \delta t^2 \sigma_v^2 \cdot {}^R\mathbf{\Phi}_k \left(\mathbf{v}\mathbf{v}^T + \mathbf{u}\mathbf{u}^T\right) ({}^R\mathbf{\Phi}_k)^T \tag{2.96} $$

where

$$ \mathbf{u} = \begin{bmatrix} 0 & 1 & 0 & 1 & \cdots \end{bmatrix}^T = \mathbf{1}_{M\times 1} \otimes \mathbf{e}_2 $$

We now note that

$$ \mathbf{v}\mathbf{v}^T + \mathbf{u}\mathbf{u}^T = \mathbf{1}_{M\times M} \otimes \mathbf{I}_2 \tag{2.97} $$

and thus

$$ \mathbf{Q}_{v_k} \preceq \delta t^2 \sigma_v^2 \cdot {}^R\mathbf{\Phi}_k \left(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2\right) ({}^R\mathbf{\Phi}_k)^T \tag{2.98} $$

$$ = \delta t^2 \sigma_v^2 (\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) \tag{2.99} $$

For the term $\mathbf{Q}_{\omega_k}$ we obtain:

$$ \text{trace}(\mathbf{Q}_{\omega_k}) = \sigma_\omega^2 \delta t^2 \,\text{trace}\left((\mathbf{I}_M \otimes \mathbf{J}) {}^R\mathbf{X}(k) {}^R\mathbf{X}(k)^T (\mathbf{I}_M \otimes \mathbf{J})^T\right) \tag{2.100} $$

$$ = \sigma_\omega^2 \delta t^2 \,\text{trace}\left({}^R\mathbf{X}(k)^T (\mathbf{I}_M \otimes \mathbf{J})^T (\mathbf{I}_M \otimes \mathbf{J}) {}^R\mathbf{X}(k)\right) \tag{2.101} $$

$$ = \sigma_\omega^2 \delta t^2 \,\text{trace}\left({}^R\mathbf{X}(k)^T \, {}^R\mathbf{X}(k)\right) \tag{2.102} $$

$$= \sigma_\omega^2 \delta t^2 \sum_{i=1}^{M} \rho_i^2 \tag{2.103}$$

where ρ_i is the distance of the i-th landmark to the robot. Thus, if ρ_o is the maximum possible distance between the robot and any landmark (determined, for example, by the robot's maximum sensing range), we obtain $\text{trace}\,(\mathbf{Q}_{\omega_k}) \;\leq\; M\rho_o^2\sigma_\omega^2\delta t^2$ and therefore

$$\mathbf{Q}_{\omega_k} \preceq M\rho_o^2\sigma_\omega^2\delta t^2 \mathbf{I}_{2M} \tag{2.104}$$

By combining this result with those of (2.94) and (2.99), we obtain:

$${}^R\mathbf{Q}_k \preceq \delta t^2\sigma_v^2(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + M\rho_o^2\sigma_\omega^2\delta t^2\mathbf{I}_{2M} = q_1(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + q_2\mathbf{I}_{2M} \tag{2.105}$$

And thus an upper bound for ${}^R\mathbf{Q}_k$ is given by the matrix

$$\mathbf{Q}_u = \underbrace{\delta t^2\sigma_v^2}_{q_1}(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + \underbrace{M\rho_o^2\sigma_\omega^2\delta t^2}_{q_2}\mathbf{I}_{2M} \tag{2.106}$$

$$= q_1(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + q_2\mathbf{I}_{2M} \tag{2.107}$$

An upper bound on the measurement covariance matrix, $\mathbf{R}_k$, can be derived by considering the characteristics of the particular sensor used for the relative position measurements. If the covariance matrix of the measurement of each individual landmark can be bounded above by $\mathbf{R}_{i_k} \preceq r\mathbf{I}_2$, then we obtain

$$\mathbf{R}_k \preceq r\mathbf{I}_{2M} = \mathbf{R}_u$$

Regarding the initial value of the recursion in (2.93), it is easy to see that the following matrix satisfies the condition $\mathbf{P}_0^u \succeq \mathbf{P}_0$:

$$\mathbf{P}_0^u \;=\; \begin{bmatrix} \mathbf{Q}_u + r\mathbf{I}_{2M} & r\mathbf{I}_{2M} \\ r\mathbf{I}_{2M} & r\mathbf{I}_{2M} \end{bmatrix} \tag{2.108}$$

Using the above results, and following Lemma 1, we can formulate the following Riccati

recursion:

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_k \left(\mathbf{P}_k - \mathbf{P}_k \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_{k+1|k} \mathbf{H}^T + \mathbf{R}_{k+1} \right)^{-1} \mathbf{H} \mathbf{P}_k \right) \mathbf{\Phi}_k^T + \mathbf{G}\, \mathbf{Q}_u \mathbf{G}^T \tag{2.109}$$

whose solution is an upper bound for the covariance of the state estimates in the dual-map filter. A difficulty in solving for the steady-state value of this recursion is that the matrix $\mathbf{\Phi}_k$ is time-varying. Considering, however, the special structure of the matrices that appear in this recursion, the following lemma can be proven:

Lemma 10 *Let the solution, $\mathbf{P}_k^u$, to the recursion in* (2.109) *be partitioned into $2M \times 2M$ blocks as*

$$\mathbf{P}_k^u = \begin{bmatrix} {}^R\mathbf{P}_k^u & \mathbf{P}_{RG_k} \\ \mathbf{P}_{RG_k}^T & {}^G\mathbf{P}_k^u \end{bmatrix} \tag{2.110}$$

Additionally, let $\mathbf{P}'_k$ be the solution to the recursion

$$\mathbf{P}'_{k+1} = \mathbf{P}'_k - \mathbf{P}'_k \mathbf{H}^T \left(\mathbf{H} \mathbf{P}'_k \mathbf{H}^T + \mathbf{R}_u \right)^{-1} \mathbf{H} \mathbf{P}'_k + \mathbf{G} \mathbf{Q}_u \mathbf{G}^T \tag{2.111}$$

with initial condition $\mathbf{P}'_0 = \mathbf{P}_0^u$, and let $\mathbf{P}'_k$ be partitioned as

$$\mathbf{P}'_k = \begin{bmatrix} {}^R\mathbf{P}'_k & \mathbf{P}'_{RG_k} \\ \mathbf{P}'^T_{RG_k} & {}^G\mathbf{P}'_k \end{bmatrix} \tag{2.112}$$

Then, for any $k \geq 0$, and for the transition matrix $\mathbf{\Phi}_k$ defined in (2.85)*, the following relations hold:*

$${}^R\mathbf{P}'_k = {}^R\mathbf{P}_k^u, \quad {}^G\mathbf{P}'_k = {}^G\mathbf{P}_k^u, \quad \text{and} \quad \mathbf{P}_{RG_k} = {}^R\mathbf{C}_k \mathbf{P}'_{RG_k} \tag{2.113}$$

where ${}^R\mathbf{C}_k = {}^R\mathbf{\Phi}_k \cdot {}^R\mathbf{\Phi}_{k-1} \cdots {}^R\mathbf{\Phi}_0$.

Proof: See Appendix A.10 ∎

The above lemma demonstrates that to derive an upper bound on the steady-state covariance of the absolute and relative maps in SLAM, it suffices to determine the steady-

state solution of the Riccati in (2.111). This recursion is simpler than that of (2.109), since it is a constant-coefficient Riccati recursion. It should be pointed out that (2.111) provides us with an upper bound for the covariance of the dual-map filter after every EKF *propagation* step. To derive a bound for the covariance immediately after the *update* step of the EKF, we account for the fact that during propagation, the absolute map covariance remains unchanged, while the uncertainty of the relative map is increased according to (2.88).

Application of Lemma 10, and evaluation of the asymptotic solution of (2.111) yields the following upper bound on the steady-state covariance matrix of both the relative and the absolute map after the EKF update step (cf. Appendix A.11):

$$ {}^{R}\mathbf{P}^{u}_{\infty} = {}^{G}\mathbf{P}^{u}_{\infty} = b_1(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + b_2\mathbf{I}_{2M} \tag{2.114} $$

where

$$ b_1 = -\frac{q_1}{2} + \frac{1}{M}\sqrt{\frac{(Mq_1+q_2)^2}{4} + (Mq_1+q_2)r} - \frac{1}{M}\sqrt{\frac{q_2^2}{4} + q_2 r} \tag{2.115} $$

and

$$ b_2 = -\frac{q_2}{2} + \sqrt{\frac{q_2^2}{4} + q_2 r} \tag{2.116} $$

This result provides bounds for the accuracy of the map in SLAM, which are evaluated in closed form, and depend on the accuracy of the robot's sensors, as well as on the size of the mapped area. Interestingly, the bounds on both the relative and absolute map are equal. However, it should be clear that the actual covariance matrices of the two map representations are not identical at steady state (in fact, the covariance of the relative map will fluctuate at steady state, due to the infusion of uncertainty during the robot's motion). In the next section, we show how these results can be used for obtaining bounds on the covariance of the robot's pose estimates in SLAM.

2.6.3 The accuracy of pose estimation in SLAM

Estimation of the robot pose from the dual map

Although the robot pose is not explicitly contained in the state vector of the dual-map filter, an estimate for this pose is implicitly defined from the estimates of the relative map, ${}^{R}\mathbf{X}(k)$, and the absolute map, ${}^{G}\mathbf{X}$. Specifically, the relation between the representation of the i-th landmark in the global frame, ${}^{G}\mathbf{p}_i$, and in the robot frame at time-step k, ${}^{R_k}\mathbf{p}_i$, is given by:

$$ {}^{G}\mathbf{p}_i \;=\; {}^{G}\mathbf{p}_{R_k} + \mathbf{C}(\phi_k)\,{}^{R_k}\mathbf{p}_i \tag{2.117} $$

where ${}^{G}\mathbf{p}_{R_k}$ and ϕ_k are the position and orientation of the robot with respect to the global frame at time-step k, respectively. Thus, given the augmented state vector at time-step k, $\mathbf{X}(k) = [{}^{R}\mathbf{X}(k)^T \;\; {}^{G}\mathbf{X}^T]^T$, and its covariance, $\mathbf{P}_k$, we are able to determine the robot pose,

$$ \boldsymbol{\theta}_k = \begin{bmatrix} {}^{G}\mathbf{p}_{R_k}^T & \phi_k \end{bmatrix}^T $$

and its covariance, $\mathbf{P}_{\boldsymbol{\theta}\boldsymbol{\theta}}$, by solving the least-squares minimization problem:

$$ \min_{\boldsymbol{\theta}_k} \boldsymbol{\varepsilon}_k^T \mathbf{W}_k^{-1} \boldsymbol{\varepsilon}_k \tag{2.118} $$

where $\boldsymbol{\varepsilon}_k$ is the vector of errors that we seek to minimize, i.e., the $2M \times 1$ vector whose i-th block is equal to

$$ \boldsymbol{\varepsilon}_{k_i} \;=\; {}^{G}\mathbf{p}_{R_k} + \mathbf{C}(\phi_k)\,{}^{R_k}\mathbf{p}_i - {}^{G}\mathbf{p}_i \tag{2.119} $$

and $\mathbf{W}_k$ is the covariance matrix of the vector $\boldsymbol{\varepsilon}_k$. Employing linearization of (2.119), we obtain

$$ \mathbf{W}_k = \mathbf{H}_{X_k} \mathbf{P}_k \mathbf{H}_{X_k}^T \tag{2.120} $$

where

$$\mathbf{H}_{X_k} = \begin{bmatrix} \mathbf{I}_M \otimes \mathbf{C}(\phi_k) & -\mathbf{I}_{2M} \end{bmatrix} \tag{2.121}$$

is the Jacobian of the error vector $\boldsymbol{\varepsilon}_k$ with respect to the state vector $\mathbf{X}_k$. The covariance matrix of the least-squares estimate for $\boldsymbol{\theta}_k$ is:

$$\begin{aligned} \mathbf{P}_{\boldsymbol{\theta}\boldsymbol{\theta}} &= \left(\mathbf{H}_{\boldsymbol{\theta}_k}^T \mathbf{W}_k^{-1} \mathbf{H}_{\boldsymbol{\theta}_k}\right)^{-1} \\ &= \left(\mathbf{H}_{\boldsymbol{\theta}_k}^T \left(\mathbf{H}_{X_k} \mathbf{P}_k \mathbf{H}_{X_k}^T\right)^{-1} \mathbf{H}_{\boldsymbol{\theta}_k}\right)^{-1} \end{aligned} \tag{2.122}$$

where $\mathbf{H}_{\boldsymbol{\theta}_k}$ is the Jacobian matrix of the error vector $\boldsymbol{\varepsilon}_k$ with respect to $\boldsymbol{\theta}_k$. This is a $2M \times 3$ block matrix, whose i-th block element is equal to

$$\mathbf{H}_i = \begin{bmatrix} \mathbf{I}_2 & \breve{\mathbf{p}}_{i_k} \end{bmatrix}, \quad \text{with} \quad \breve{\mathbf{p}}_{i_k} = \mathbf{J}\mathbf{C}(\phi_k)^{R_k}\mathbf{p}_i \tag{2.123}$$

We point out that the solution of the least-squares problem in (2.118) and the covariance of this solution, given by (2.122), yield the same results for the robot's pose as the "standard" EKF formulation for SLAM, when at least 2 landmarks are available. This is because in both cases, all the available measurements are used, and no approximations are made (apart from linearization). Thus, we can use the expression of (2.122), to study the properties of the robot's pose covariance in EKF-based SLAM.

2.6.4 Upper bounds on the robot-pose covariance

We now focus on deriving upper bounds on the steady-state value of the matrix $\mathbf{P}_{\boldsymbol{\theta}\boldsymbol{\theta}}$. Note that since $\mathbf{P}_k \preceq \mathbf{P}_k^u$, an upper bound on the covariance of the robot pose at time-step k is given by (cf. (2.122)):

$$\mathbf{P}_{\boldsymbol{\theta}\boldsymbol{\theta}_k}^u = \left(\mathbf{H}_{\boldsymbol{\theta}_k}^T \left(\mathbf{H}_{X_k} \mathbf{P}_k^u \mathbf{H}_{X_k}^T\right)^{-1} \mathbf{H}_{\boldsymbol{\theta}_k}\right)^{-1} = \begin{bmatrix} \mathbf{P}_{pp_k} & \mathbf{P}_{p\phi_k} \\ \mathbf{P}_{p\phi_k}^T & P_{\phi\phi_k} \end{bmatrix} \tag{2.124}$$

Substitution of the asymptotic results from (2.114), and of the values of the Jacobians $\mathbf{H}_{\boldsymbol{\theta}_k}$ and $\mathbf{H}_{X_k}$ from (2.121) and (2.123), enables us to determine upper bounds on the asymptotic

uncertainty of the robot pose during SLAM. Specifically, the upper bound on the steady-state orientation variance is computed from the above expression as (cf. Appendix A.12):

$$P_{\phi\phi_\infty} = \frac{4Mb_2}{\sum_{i=1}^{M}\sum_{j=1}^{M}\rho_{ij}^2} \tag{2.125}$$

where ρ_{ij} is the distance between landmarks i and j (and $\rho_{ii} = 0$). Thus, if the pairwise distances of the landmarks are known, an upper bound on the robot's orientation variance is determined by the preceding expression. Alternatively, if some properties of the placement of the landmarks in space are known, then using this expression, we can determine bounds that are independent of the actual landmark positions. For example, if the minimum allowable distance between any two landmarks is equal to $\rho_{LL_{\min}}$, then

$$P_{\phi\phi_\infty} \leq \frac{4b_2}{(M-1)\rho_{LL_{\min}}^2} \tag{2.126}$$

For the upper bound on the covariance matrix of the robot's position estimates at steady state, we obtain from (2.124) (cf. Appendix A.12):

$$\mathbf{P}_{pp_\infty} = \frac{2b_2 + 2Mb_1}{M}\mathbf{I}_2 + \underbrace{\frac{2b_2\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)}{M^2\left(\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\right)}}_{\mathbf{T}_2}$$

To derive an upper bound for $\mathbf{P}_{pp_\infty}$, we examine the trace of the second term, $\mathbf{T}_2$, in the last expression. In Appendix A.12 it is shown that:

$$\text{trace}(\mathbf{T}_2) \;=\; 4b_2\frac{\sum_{i=1}^{M}\left({}^R\mathbf{p}_i^T\,{}^R\mathbf{p}_i\right)}{\sum_{i=1}^{M}\sum_{j=1}^{M}\rho_{ij}^2} - \frac{2b_2}{M} \tag{2.127}$$

and thus, by combining the preceding two equations we obtain:

$$\mathbf{P}_{pp_\infty} \preceq \frac{2b_2 + 2Mb_1}{M}\mathbf{I}_2 + \text{trace}(\mathbf{T}_2)\mathbf{I}_2 \tag{2.128}$$

$$= 2b_1\mathbf{I}_2 + \frac{1}{M}P_{\phi\phi}\sum_{i=1}^{M}\rho_i^2\mathbf{I}_2 \tag{2.129}$$

Finally, given that the maximum distance between the robot and any landmark is ρ_o, the

covariance of the robot's position estimate is bounded above by

$$\mathbf{P}_{pp_\infty} \preceq \left(2b_1 + \rho_o^2 P_{\phi\phi}\right) \mathbf{I}_2 \tag{2.130}$$

This result, along with those of (2.125)-(2.126), which determine upper bounds on the robot's orientation uncertainty, and that of (2.114), which yields the upper bounds on the covariance of the map, are the key results of the SLAM performance analysis. They enable us to compute the guaranteed accuracy of the state estimates in SLAM, as an analytical function of the accuracy of the robot's sensors, and the properties of the landmarks' spatial configuration. Hence, these expressions can be employed to determine whether a candidate robot system design satisfies the accuracy requirements of a given SLAM application, without the need for simulations or experimentation. In the following section, we present results from real-world experiments, which demonstrate the validity of the preceding theoretical analysis.

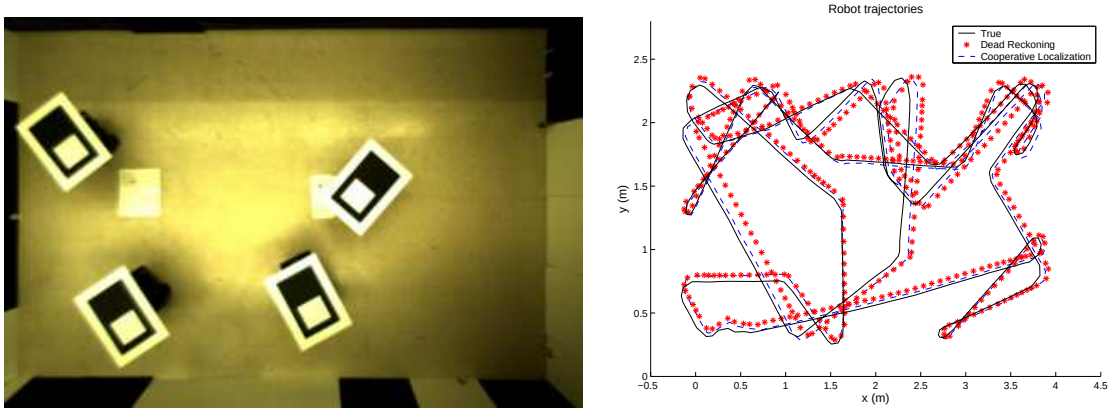
2.7 Experimental and simulation results

2.7.1 Cooperative localization

Real-World Experiments

A series of experiments were conducted for validating the theoretical analysis of the accuracy in CL, presented in Section 2.4. Our experimental setup is shown in Fig. 2.1(a). A team of four Pioneer I robots moves in a rectangular area, within which the positions of the robots are being tracked by an overhead camera. For this purpose, rectangular targets are mounted on top of the robots and the vision system is calibrated in order to provide measurements of the pose of the robots in a global coordinate frame. The standard deviation of the noise in these measurements is approximately 0.5^o for orientation and 1 cm, along each axis, for position. The robots were commanded to move at a constant velocity of 0.1 m/sec, while avoiding collision with the boundaries of the arena as well as with their teammates.

Although four identical robots were used, calibration of their odometric sensors showed that the accuracy of the wheel encoder measurements is not identical for all robots. Specifi-



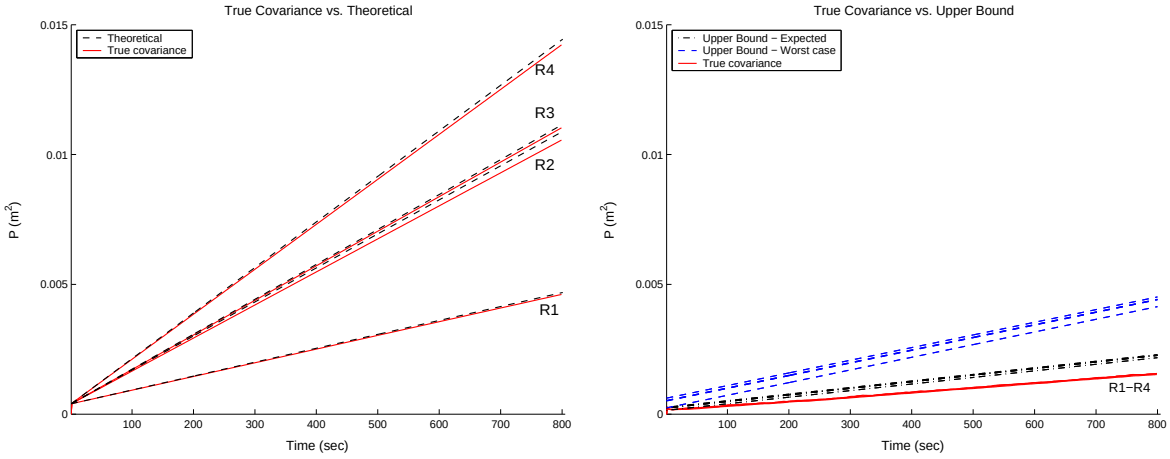
(a) (b)

Figure 2.1: (a) Calibrated image of robots with targets mounted on top of them. (b) True and estimated trajectories for robot 1. For presentation clarity only part of the trajectory, corresponding to the first 450 sec, is plotted. The size of the arena is approximately 2.5 × 4.5 m.

cally, the measurement errors are well-modeled as Gaussian, zero-mean, white-noise processes and the standard deviation of the velocity measurements ranges from $\sigma_{V_{\min}} = 0.038V$, for the most accurate odometer, to $\sigma_{V_{\max}} = 0.069V$, for the robot with the highest noise levels. Similarly, the standard deviations of the rotational velocity measurements have values between $\sigma_{\omega_{\min}} = 0.0078$ rad/sec and $\sigma_{\omega_{\max}} = 0.02$ rad/sec for the four robots. Although this had not been planned for, we observe that as a result of the variability of the sensor characteristics, attributed to manufacturing imperfections, the experiments involve a heterogeneous robot team.

Each of the robots is equipped with a laser range finder, that is used for measuring absolute orientation. This is done by exploiting the perpendicularity of the surfaces surrounding the arena and employing a simple line-fitting technique. The standard deviation of the errors in the orientation measurements is approximately 0.5° for all robots.

Relative position measurements are produced synthetically, using the differences in the positions of the robots recorded by the overhead camera, expressed in the measuring robot's coordinate frame, with the addition of noise. This facilitates the study of the effects of varying the accuracy of the relative position measurements, and allows for control of the topology of the RPMG. For the experimental results shown in this section, a complete



(a) Dead Reckoning (DR)

(b) Cooperative Localization (CL)

Figure 2.2: Time evolution of the true covariance of the position estimates (solid bounding lines), and theoretically computed values (dashed black lines).

RPMG topology is formed and the relative position measurements (distance and bearing) are corrupted by zero-mean white Gaussian noise processes with standard deviation $\sigma_\rho = 0.05$ m and $\sigma_\theta = 0.0349$ rad.

In Fig. 2.1(b), the true trajectory (solid line) for one of the robots, as measured by the overhead camera, is compared to the trajectory estimated using DR (asterisks) and CL (dashed line). The significant improvement in positioning performance, resulting from the use of relative position information, is apparent and is demonstrated more clearly in Fig. 2.2, where the time evolution of the covariance is shown. Fig. 2.2(a) corresponds to the case in which the four robots localize independently and compares the true[3] covariance of the robots' position estimates (solid lines) to the expected covariance values computed by (2.60) (dashed lines). On the other hand, Fig. 2.2(b) corresponds to the CL case and presents the true covariance (solid lines) as well as the theoretically derived upper bound for the expected covariance (dash-dotted lines) and the upper bound for the worst-case covariance (dashed lines). It is evident that the derived upper bound is indeed larger than the actual covariance of the position estimates. Moreover, we note that despite the fact that we deal with a heterogeneous team, the positioning uncertainty increases at the *same*

[3]By "true" we refer to the covariance estimated by the EKF employed for position estimation.

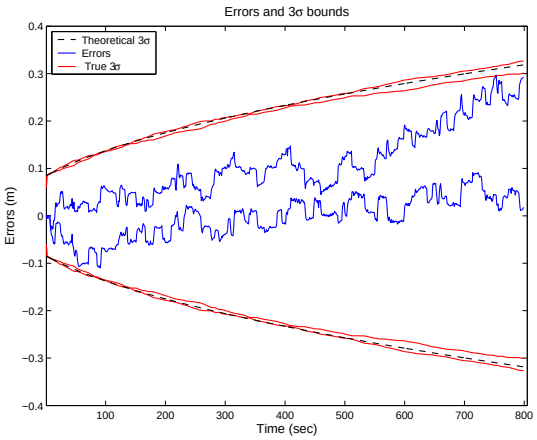
rate for all robots. This rate is significantly smaller compared to that of the robot with the most accurate sensors localizing when relying on DR (Robot 2 as shown in Fig 2.2(a)). This observation agrees with the theoretical result of Corollary 5.

In Figs. 2.3-2.4, the errors in the position estimates of the robots are plotted, and compared against the $\pm 3\sigma$ values of the position estimates' covariance. The solid lines represent the $\pm 3\sigma$ values associated with the true covariance, while the dashed ones represent the $\pm 3\sigma$ values computed using (2.60) for the case of DR, and the upper bound on the expected covariance for the case of CL. In these plots, the substantial improvement in positioning accuracy, achieved when the robots are recording and processing relative position measurements, is illustrated. However, the most important conclusion drawn from these figures is that the derived analytical expressions can be employed in order to accurately predict the localization performance of a robot team. The $\pm 3\sigma$ enveloping lines, evaluated using the derived analytical expressions, define a confidence region that closely describes the magnitude of the position errors. This justifies the use of the covariance matrix as a performance metric, and demonstrates that for a robot team with known sensor noise characteristics, it is possible to predict the positioning accuracy, without resorting to simulations or experimentation.

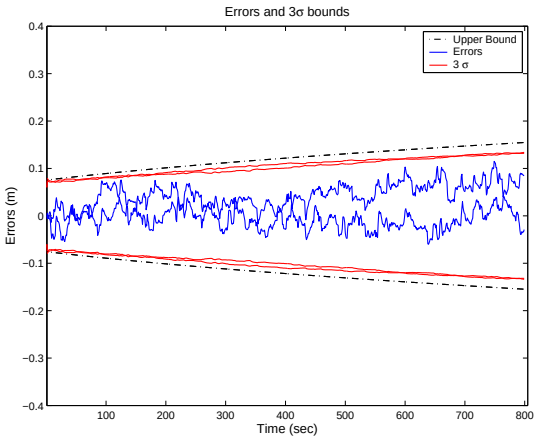
Simulation Results

We now present simulation results that demonstrate the effect of RPMG reconfigurations and corroborate the corresponding theoretical analysis. In order to isolate the effects of different RPMG topologies, a homogeneous team comprised of 9 robots is considered in these simulations. The robots are restricted to move in an area of radius $r = 20$ m, and their velocity is assumed to be constant, equal to $V_i = 0.25$ m/sec. The orientation of the robots, while they move, changes randomly using samples drawn from a uniform distribution of width 20^{o}.

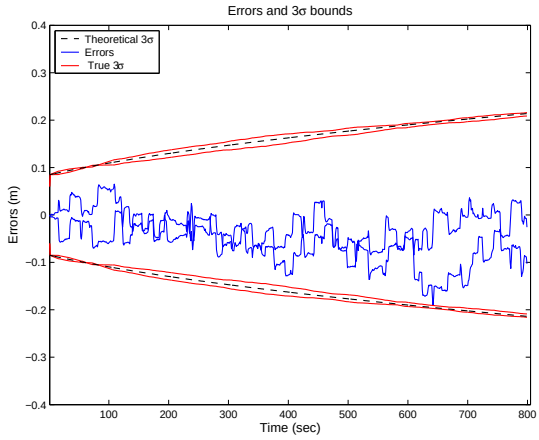
The parameters of the noise that corrupts the proprioceptive measurements of the simulated robots are identical to those measured on a iRobot PackBot robot ($\sigma_V = 0.0125$ m/sec, $\sigma_\omega = 0.0384$ rad/sec). The absolute orientation of each robot was measured by a simulated compass with $\sigma_\phi = 0.0524$ rad. The robot tracker sensor returned range and bearing mea-



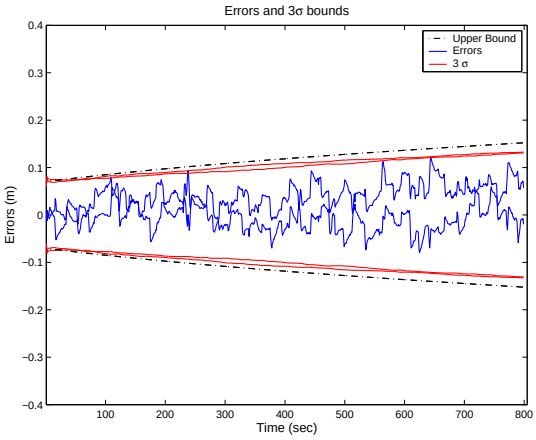
(a) Robot 1 - DR



(b) Robot 1 - CL

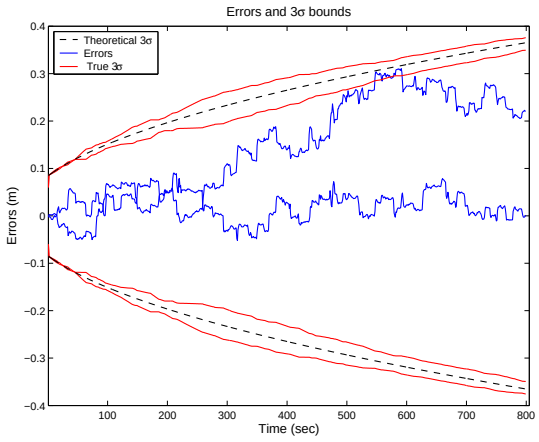


(c) Robot 2 - DR

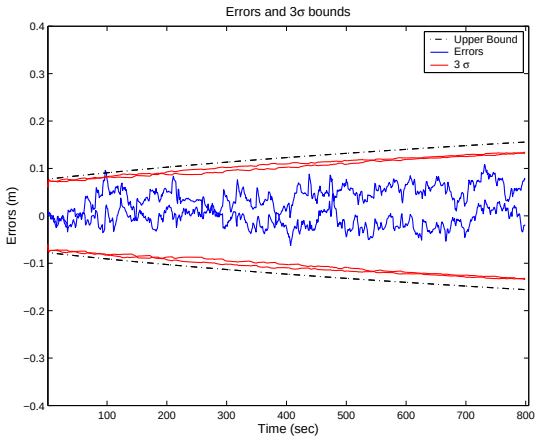


(d) Robot 2 - CL

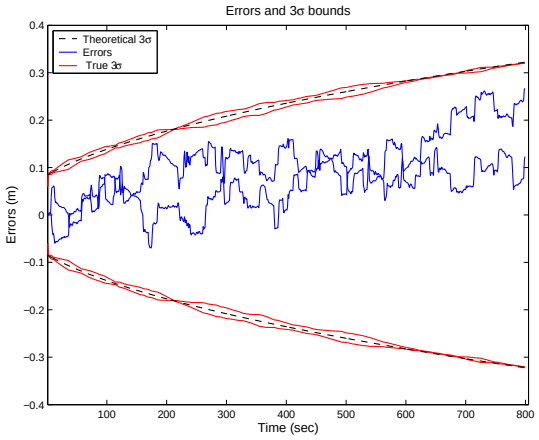
Figure 2.3: Left column: errors (solid blue lines) in the position estimates for the first two robots when they perform DR. Right column: position errors during CL for the same robots. The solid bounding lines represent the $\pm 3\sigma$ values of the actual covariance, computed by the EKF, while the dash-dotted bounding lines represent the $\pm 3\sigma$ values computed employing the theoretical upper bound for the expected covariance.



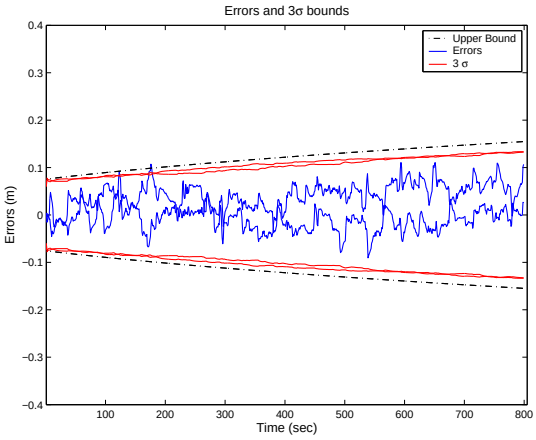
(a) Robot 3 - DR



(b) Robot 3 - CL



(c) Robot 4 - DR



(d) Robot 4 - CL

Figure 2.4: Left column: errors (solid blue lines) in the position estimates for the third and fourth robots when they perform DR. Right column: position errors during CL for the same robots. The solid bounding lines represent the $\pm 3\sigma$ values of the actual covariance, computed by the EKF, while the dash-dotted bounding lines represent the $\pm 3\sigma$ values computed employing the theoretical upper bound for the expected covariance.

surements corrupted by zero-mean white noise with $\sigma_\rho = 0.01$ m and $\sigma_\theta = 0.0349$ rad. The above values are compatible with noise parameters observed in laboratory experiments [134]. All measurements were available at 1 Hz.

In Fig. 2.5, the time evolution of the positioning uncertainty of the robot team is shown. Initially, up to $t = 200$ sec, the robots do not record any relative position measurements and propagate their position estimates using DR. At $t = 200$ sec, the robots start receiving relative position measurements and the topology of the RPMG becomes a complete one (Fig. 2.6(a)). The significant improvement in the rate of uncertainty increase, achieved by using relative positioning information, is demonstrated in this transition. At $t = 400$ sec, the RPMG assumes a ring topology (Fig. 2.6(b)). We note that the uncertainty undergoes a transient phase, during which it increases at a higher rate and then, once steady state is reached, the rate of increase is identical to the rate associated with the complete graph. This corroborates the result of (2.53) and shows that the dominant factor in determining the rate of localization uncertainty increase is the quality of the proprioceptive sensors of the robots.

At $t = 600$ sec, a supposed failure of the communication network occurs, and in the time interval between 600 sec and 800 sec only two robots are able to measure their relative positions (Fig. 2.6(c)). This case can be viewed as a degenerate one, where the 7 robots localize based solely on DR, while the other 2 robots form a smaller team. As expected, the rate of uncertainty increase is higher when the team consists of only 2 robots, instead of 9, but lower when compared to DR.

At $t = 800$ sec, the RPMG resumes the complete graph topology (Fig. 2.6(a)). It is evident that the uncertainty in the position estimates during the time intervals $(200,\ 400)$ sec and $(800,\ 1000)$ sec is described by the *same* linear function of time. This occurs despite the prior two reconfigurations of the topology of the RPMG, which occurred at $t = 400$ sec and $t = 600$ sec. This result is in accordance with the theoretical analysis of Section 2.4.7.

At $t = 1000$ sec, the RPMG assumes a non-canonical topology (Fig. 2.6(d)). This scenario is perhaps the most important one for real applications, since robots will usually measure the relative positions of neighboring robots that are within their field of view. We observe that the positioning uncertainty increases at a rate identical to that of Phases I

and II of the graph's topology, as predicted by our theoretical analysis. It is also apparent that the uncertainty for each robot converges to a set of lines with the same slope (rate of uncertainty increase), but with different constant offset. This is due to the effect of the different degree of connectivity in the RPMG of each robot. Connection-rich robots have direct access to positioning information from more robots and thus attain lower positioning uncertainty.

Finally, at $t = 1200$ sec, one of the robots starts receiving GPS measurements, while the RPMG retains the topology of Fig. 2.6(d) The GPS measurements are corrupted by noise with a standard deviation of $\sigma_{GPS} = 0.05$ m along each axis. It is evident that the availability of absolute position measurements to any robot drastically reduces the localization uncertainty for all the robots in the group. Furthermore, the system becomes observable and the uncertainty is bounded for all the members of the team. As in the previous phase, the uncertainty for the position of each robot converges to a value (constant in this case) that depends on its degree of connectivity.

2.7.2 Cooperative SLAM

Simulation Results

We now focus on the case of Cooperative SLAM. A series of numerical experiments were conducted, in order to validate the analysis presented in Section 2.5. The four simulated robots move in an arena of dimensions 10 m $\times$ 10 m, where point landmarks are located. The velocity of the robots is kept constant at $V = 0.25$ m/s, while their orientation changes randomly, using samples drawn from a uniform distribution. To simplify the presentation, a homogeneous robot team is assumed. The standard deviation of the velocity measurement noise is $\sigma_V = 0.05V$, and the standard deviation of the errors in the orientation estimates is $\sigma_\phi = 2^{\text{o}}$, for all robots. Similarly, the values selected for the standard deviations of the exteroceptive measurements of the robots are $\sigma_\theta = 2^{\text{o}}$, for the bearing measurements, and $\sigma_\rho = 0.05$ m, for the range measurements. For the results presented in this section, the RPMG shown in Fig. 2.7(a) is used. For this experiment, it is assumed that initially the robots have perfect knowledge of their positions, while the landmark positions are unknown.

In order to demonstrate the validity of the bound on the worst-case covariance of C-

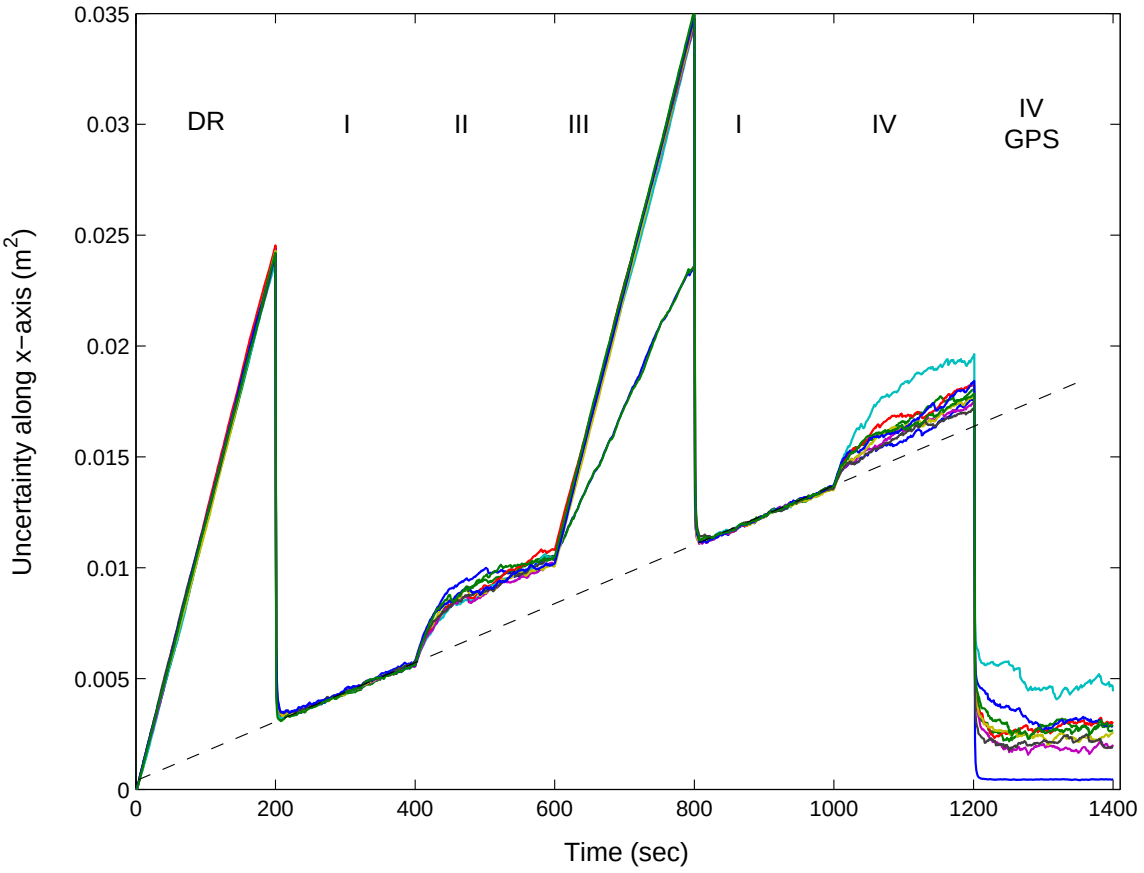
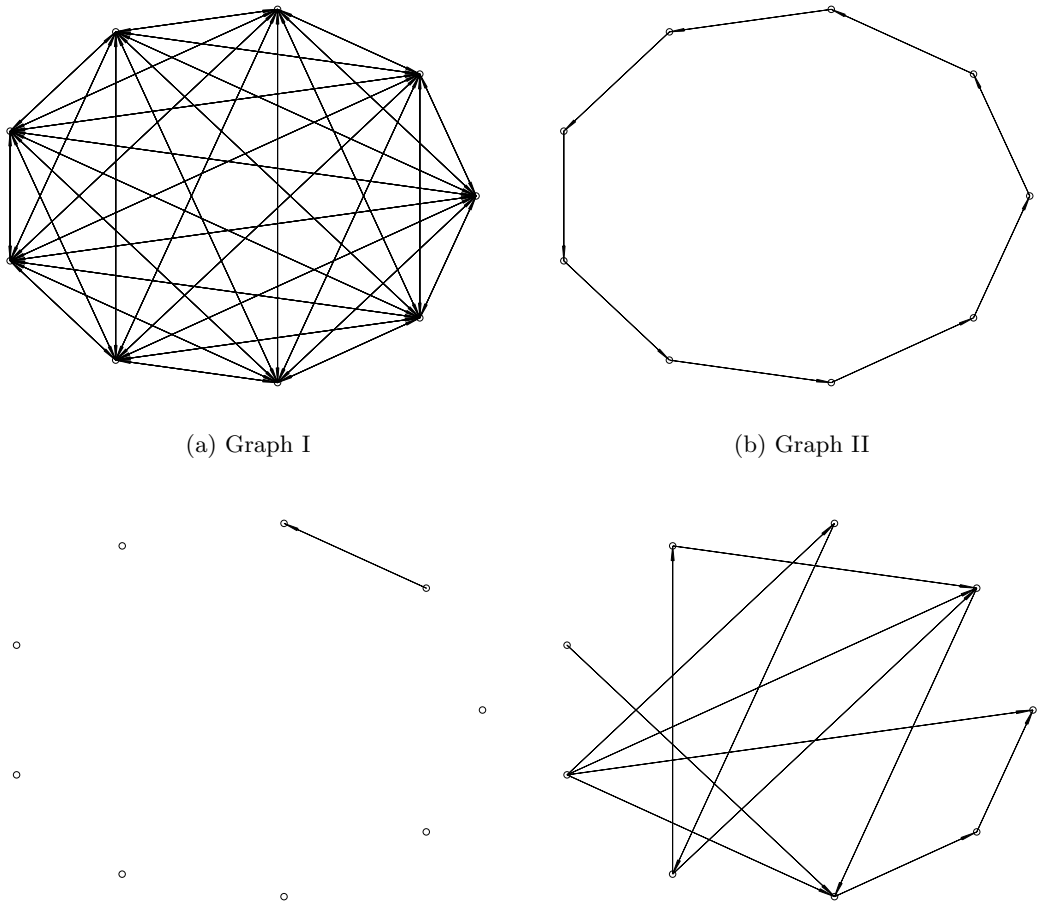


Figure 2.5: Uncertainty evolution during CL for a RPMG with changing topology. The thinner dashed line has been superimposed on the figure to facilitate the comparison between the values of the covariances for different topologies of the RMPG.



(a) Graph I

(b) Graph II

(c) Graph III

(d) Graph IV

Figure 2.6: The four different relative position measurement graph topologies considered in the CL simulations. Each arrow represents a relative position measurement, with the robot (node) where the arrow starts being the observing robot.

SLAM, provided in Lemma 8, a particularly adverse scenario for the placement of the landmarks is considered. Specifically, all the landmarks form a cluster at one corner of the arena, while the robots begin their exploration at the opposite corner (Fig. 2.7(b)). In this case the exteroceptive measurements provide only a small amount of positioning information during the crucial first few updates. In Fig. 2.7(c), the time evolution of the covariance of the position estimates for the robots and landmarks is shown and compared to the theoretically-derived steady-state performance bound. Clearly, the upper bound is indeed larger than the steady-state covariance of the landmarks and robots. It is also

worth noting that the covariance of the position estimates converges to the same value for all landmarks, while the accuracy of the position estimates varies between robots. These differences result from the non-symmetric topology of the RPMG, which causes each robot to have access to positioning information of different quality.

Although the bound of Lemma 8 accounts for the worst-case accuracy of C-SLAM, it does not yield a close description of the accuracy when the map features are more evenly distributed in space. In such cases, employing Lemma 9 results in a tighter bound on the average positioning uncertainty, as demonstrated in Fig. 2.8. In this plot, the average values (over 50 runs) of the covariance in C-SLAM, are compared against the theoretically derived bounds on the expected uncertainty. For each run of the algorithm, the locations of the landmarks, as well as the initial positions for the robots, were selected using samples from a uniform distribution. Note that the scale of the axes in Fig. 2.8 has been changed compared to Fig. 2.7(c), to preserve clarity. Mere comparison of the values for the covariance of the robots' and landmarks' position estimates with the corresponding bounds demonstrates that when available information about the distribution of the landmarks is exploited, i.e., by employing the expressions from Lemma 9, a better characterization of the expected accuracy of the position estimates is achieved.

Real-World Experiments

In the simulation results presented above, the RPMG remained unchanged throughout the duration of each experiment. However, due to occlusions and data association failures, this is usually not the case in practice. In order to demonstrate the validity of the theoretical analysis in a more realistic setting, we have also conducted real-world experiments. A team comprising two Pioneer 3 robots, each equipped with two opposite-facing SICK LMS200 laser scanners to provide a 360^{o} field of view, was employed (cf. Fig. 2.9). The robots move randomly at a constant velocity of 0.1 m/sec, while performing C-SLAM in an area with approximate dimensions of 10 m$\times$4 m. The estimated trajectories of the robots are shown in Fig. 2.10(a). A sample laser scan, acquired by robot 1, is superimposed on the same figure (after being transformed to the global frame), to illustrate the geometry of the area where the robots operate.

Since the indoor environment where the experiment takes place is rectilinear, absolute orientation measurements were obtained by employing line-fitting on the laser scans. The upper bound for the standard deviation of these measurements is $\sigma_\phi = 1^o$. Moreover, the laser scans are processed for detecting corners (which are used as landmarks), and for obtaining relative position measurements. The RPMG for this experiment is shown in Fig. 2.10(b). We note that 8 landmarks were reliably detected; of these, 6 are observed by robot 1, and 4 are observed by robot 2. Moreover, robot 1 measures the relative position of robot 2 at every time-step. To avoid introducing correlations in the measurements, each laser scan point is used at most once: for computing either the robot-to-robot, or robot-to-landmark, or absolute orientation measurements. The robots are equipped with wheel encoders that provide velocity measurements, with standard deviation $\sigma_V = 5 \times 10^{-3}$ m/sec.

In Figs. 2.11(a) and 2.11(b), the time evolution of the diagonal elements of the covariance matrix of the position estimates is shown, and compared to the upper bounds on the worst-case and average covariance, respectively. We note at this point that, primarily due to the existence of occlusions, the robot-to-landmark measurements described in the RPMG shown in Fig. 2.10(b) were not possible at every time instant. As a result, the RPMG did not remain constant for the entire duration of the experiment. On the average, the measurements described by the edges of the RPMG were successfully recorded 85% of the time. We observe that despite the changing topology of the RPMG in this case, the theoretical bounds still provide an accurate characterization of the positioning accuracy of C-SLAM.

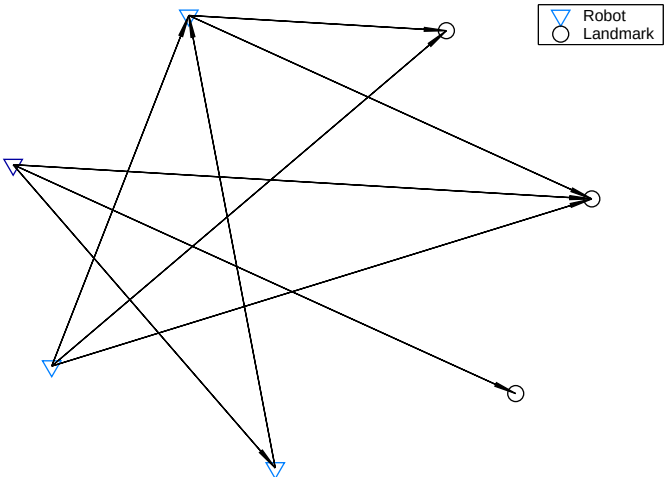
2.7.3 SLAM

We now turn our attention to single-robot SLAM, and demonstrate the application of the analysis in Section 2.6 to real-world experiments. For the experiments presented in this section, the same experimental setup as in the case of C-SLAM is employed, but with only a single robot. Moreover, this robot does not utilize absolute orientation measurements. In Figs. 2.12-2.14, the standard deviation of the estimation errors (solid lines), computed by the filter, is compared to the standard deviation determined by the theoretically derived bounds (dashed lines). These three figures present the results for the landmark positions,

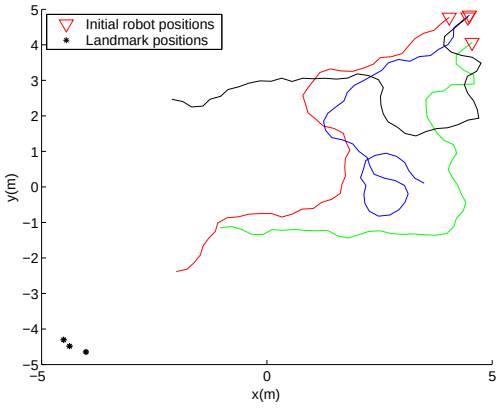
the robot position, and the robot orientation, respectively. For the robot orientation, the bound in (2.126) is employed. The results shown here validate the theoretical analysis, and show that the derived bounds can be used to predict the guaranteed localization accuracy of SLAM. We should point out that in this particular case, where the robot moves randomly in space, the actual standard deviations are approximately 2-3 times smaller than the corresponding upper bounds. If the robot's trajectory was such that the robot-to-landmark distances were always close to their maximum values, the bounds would have been significantly tighter.

2.8 Summary

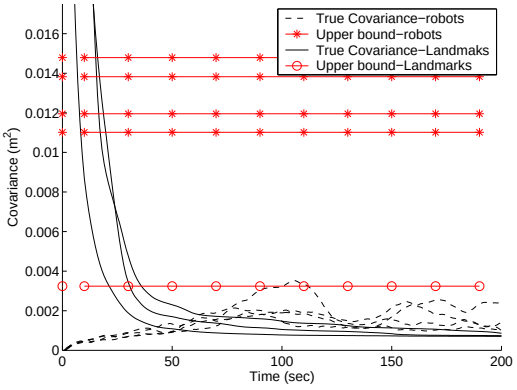
In this chapter, we have presented a study of the localization accuracy during cooperative localization, cooperative SLAM, and single-robot SLAM. We have derived closed-form expressions that allow us to predict the localization accuracy as a function of the relevant system parameters. The main results of our analysis are upper bounds on the worst-case uncertainty, as well as on the average uncertainty of the state estimates. These bounds are functions of the number of robots and landmarks, the accuracy of all available sensor measurements, the structure of the sensing graph, and the spatial distribution of the robots and landmarks in the operational area. By analytically studying the properties of the derived expressions, we have been able to identify key properties of the time evolution of the uncertainty. These properties, which have also been verified in numerical and real-world experiments, can help develop intuition about the interaction of the various factors affecting the accuracy of localization. In addition to their theoretical significance, the results of this chapter are also useful from a practical standpoint: by enabling a designer to predict the localization capabilities of a robot system, the derived relations reduce the dependence on time-consuming simulations and experimentation. This has the potential to not only significantly speed-up the design process, but also provide theoretical guarantees for the performance of a given system.



(a)



(b)



(c)

Figure 2.7: (a) The RPMG used for the numerical experiments (b) The initial positions and part of the trajectories of the robots for an adverse C-SLAM scenario. (c) Comparison of the actual covariance of the position estimates against the worst-case performance bound, for the scenario in (b). The plotted lines correspond to the mean of the covariance along the two coordinate axes.

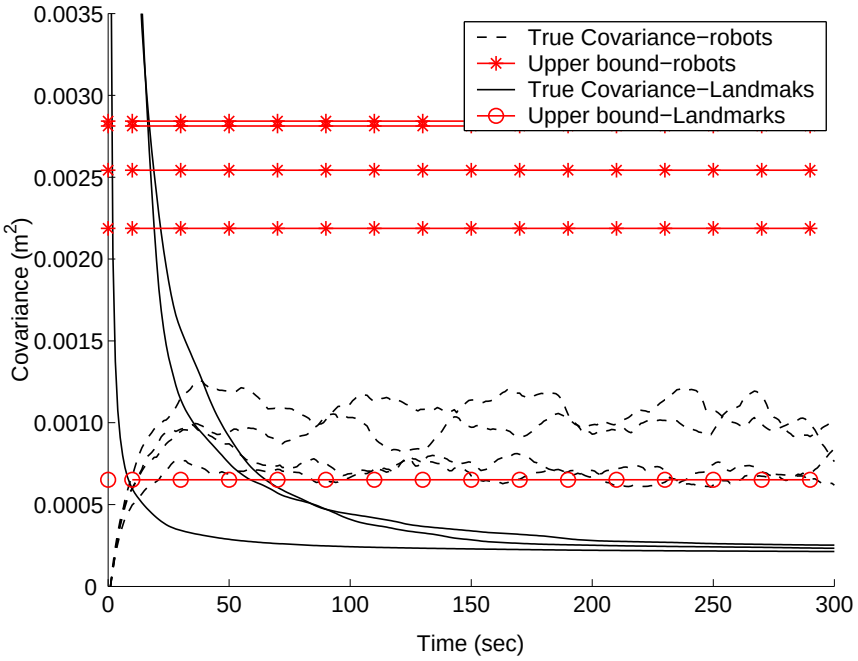
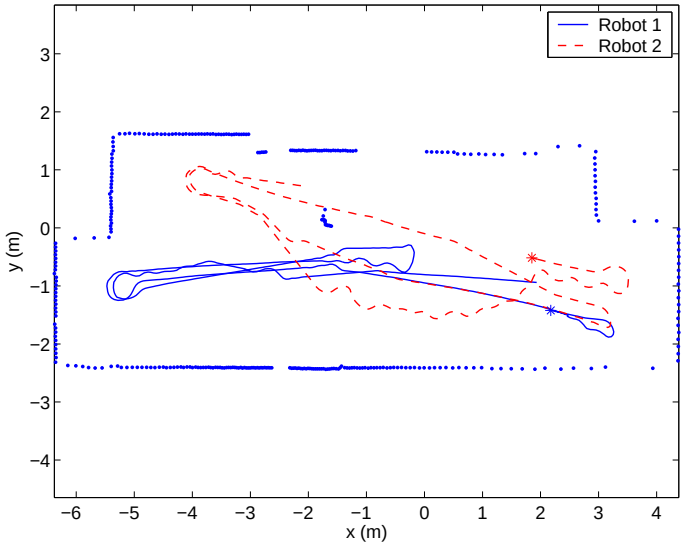
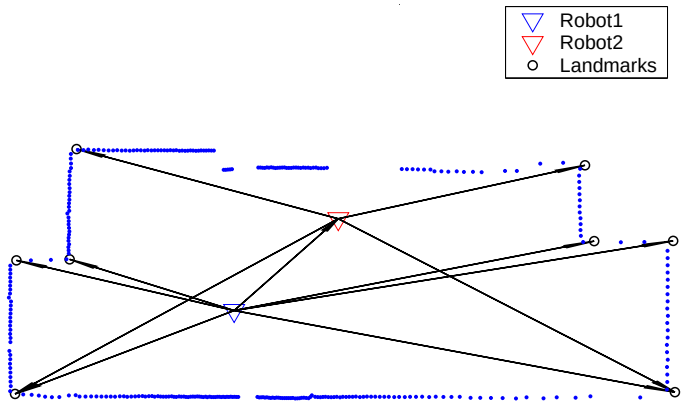


Figure 2.8: Comparison of the average true covariance of the position estimates against the corresponding upper bound. Landmark positions and the initial robot positions are selected using samples from a uniform distribution. Averages over 50 runs of C-SLAM are computed.

Figure 2.9: The Pioneer III robots used in the experiments. Two laser range-finders are installed on each robot, to provide a 360^o field of view.

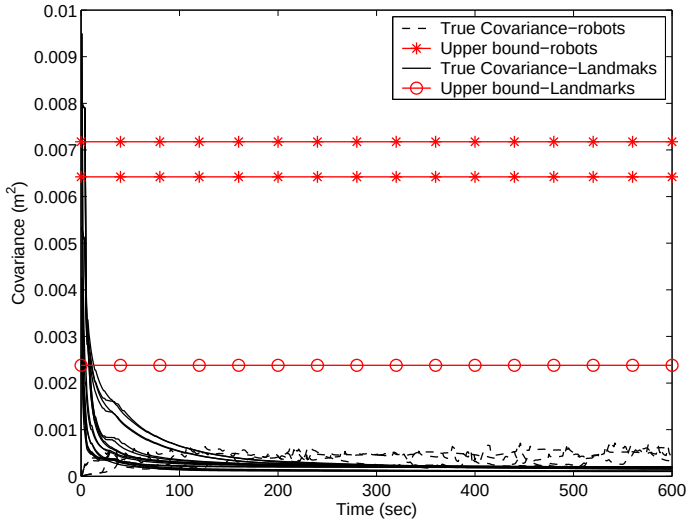


(a)

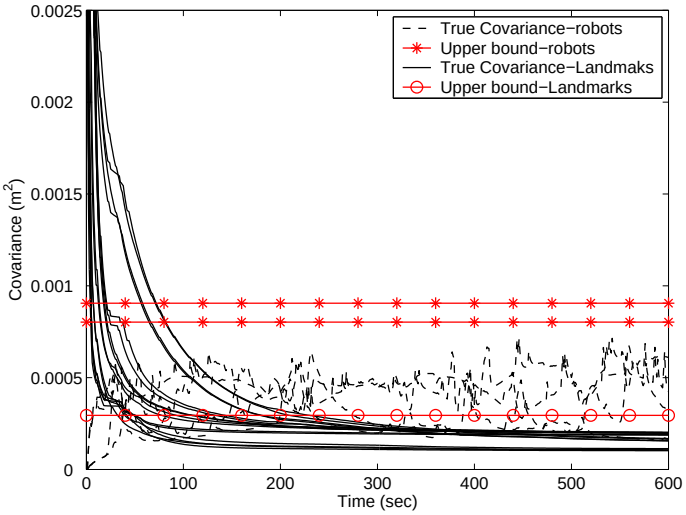


(b)

Figure 2.10: (a) The estimated trajectories of the robots, plotted along with a sample laser scan of the area where the experiment was conducted. The initial positions of the robots are indicated by asterisks (b) The RPMG used in the C-SLAM experiment.

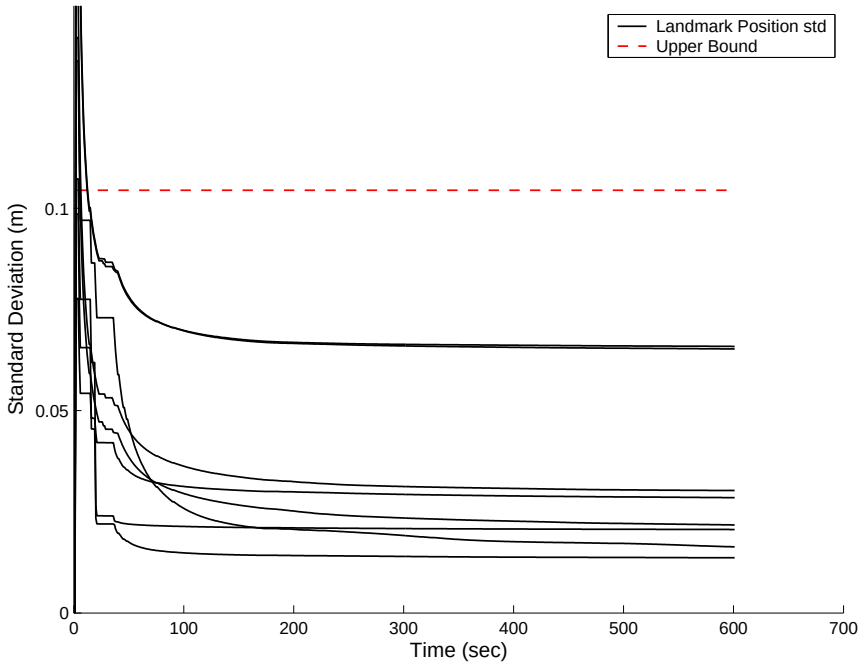


(a)



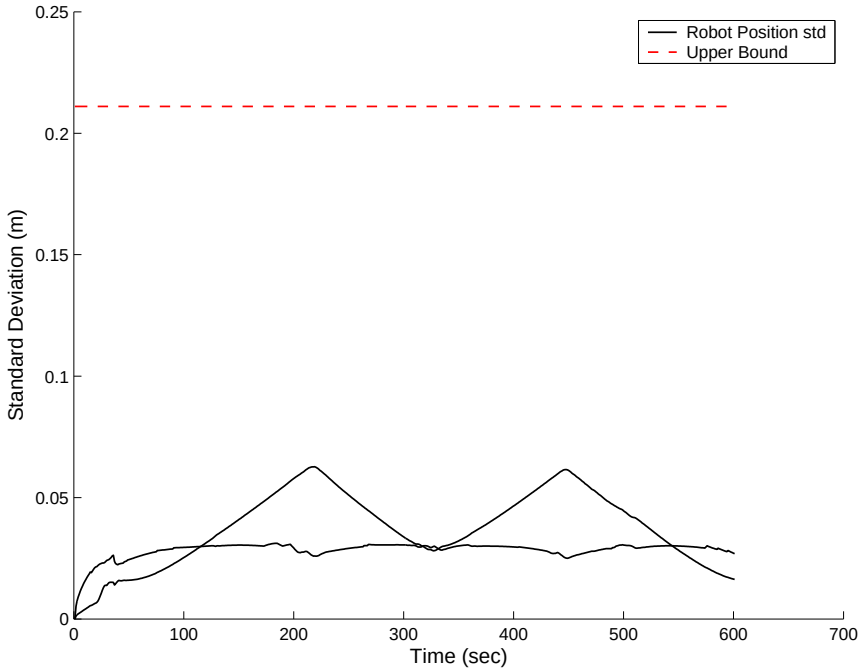
(b)

Figure 2.11: The time evolution of the diagonal elements of the covariance matrix during the experiments vs. the theoretical upper bounds. In plot (a) the true values are compared to the worst-case upper bounds, while in (b) they are compared to the upper bounds on the average uncertainty. The dashed lines correspond to the robot coordinates' covariance, the solid lines to the landmark coordinates' covariance, the lines with asterisks to the robots' covariance bounds, and the lines with circles to the landmarks' covariance bounds.



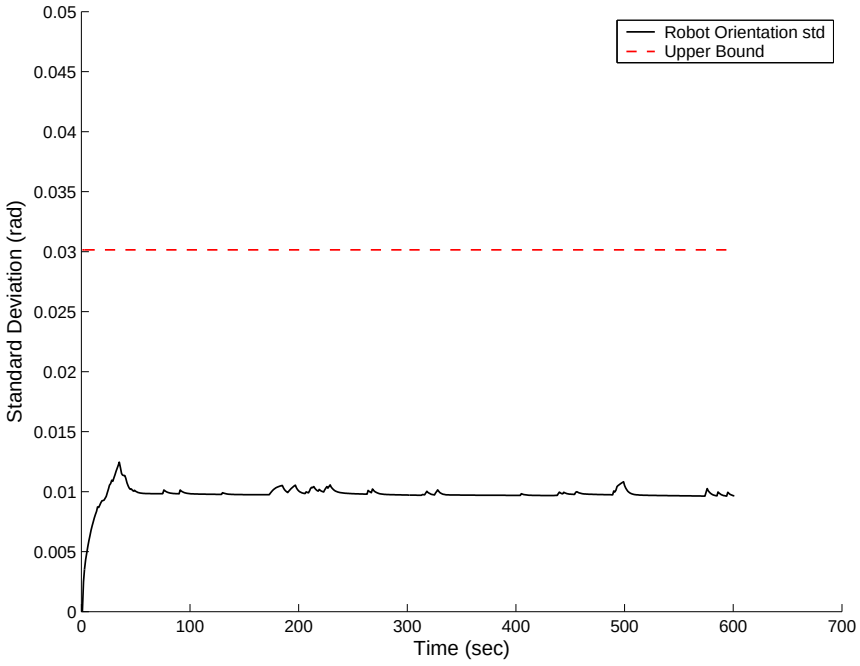
(a)

Figure 2.12: The landmarks' position standard deviation and corresponding upper bound during the SLAM experiment.



(a)

Figure 2.13: The robot's position standard deviation and corresponding upper bound during the SLAM experiment.



(a)

Figure 2.14: The robot's orientation standard deviation and corresponding upper bound during the SLAM experiment.

Chapter 3

Optimal CL in Robot Formations

3.1 Introduction

The study of the estimation accuracy in CL, presented in Section 2.4, considered the case where the robots of the team follow arbitrary trajectories within an area. We showed that in that case, the time-varying nature of the Riccati recursion prevents us from obtaining a closed-form description of the asymptotic covariance. For this reason, we resorted to deriving upper bounds, to obtain a characterization of the attainable accuracy as a function of relevant system parameters. However, there exists a number of applications where the robots of a team are required to move in a pre-specified geometric configuration, to accomplish a certain task. Such *robot formations* are encountered in several tasks (e.g., object moving [167], surveillance [8], platooning for efficient transportation systems [15], formation flying [165], and spacecraft formations [2]). In this chapter, we show that, because the relative poses of the robots in a formation are known in advance, a much closer description of the localization accuracy of a robot formation can be obtained. Moreover, we show how this can be employed in order to optimize the accuracy of localization [103, 106].

3.2 Overview of the approach

We study the problem of determining sensing strategies that maximize localization accuracy, given constraints on the available computational and communication resources. Roumeliotis and Bekey [137], have shown that proprioceptive measurements from the robots' odometry

sensors can be processed locally by each robot to propagate its own pose estimates. However, every time an exteroceptive measurement is received by any of the robots in the formation, all robots must communicate their current pose estimates. Additionally, the measuring robot must transmit its new measurement in order for the EKF update to be performed. Therefore, every exteroceptive measurement that is processed incurs a penalty in terms of use of both communication bandwidth and CPU time, as well as in terms of power consumption. In a realistic scenario, the robots of a team will need to allocate computational and communication resources to mission-specific tasks, and this may force them to reduce the number of measurements they process for localization purposes. Moreover, the finite battery life of robots imposes constraints on the amount of power that can be used for tracking their position. The limitations on the available resources may thus prohibit the robots from registering, transmitting, and processing all measurements available at every time instant.

It is clear that whether or not an exteroceptive measurement should be processed in an EKF update, is determined by a tradeoff between the value of the localization information it carries, and the cost of processing it. In this chapter, we assume that the robots process each of the available measurements at a constant frequency, and we seek the *optimal measurement frequencies*, in order to attain the highest possible positioning accuracy. The key element in our analysis is the derivation of an *equivalent continuous-time system model*, whose noise parameters are functionally related to the frequency of the measurements. This enables us to express the covariance matrix of the pose errors (and thus the localization accuracy) as a function of the sensors' frequencies. Moreover, using this description of accuracy, we formulate an optimization problem for determining the optimal sensing strategy. An important result that we prove is that this is a *convex optimization problem* and therefore it is possible to compute a globally optimal solution, using efficient algorithms.

3.3 Prior work

In [26, 27, 152], localization algorithms for recovering the relative poses between the robots in a formation, using omnidirectional cameras as the primary sensors, are described. The authors propose suboptimal estimation algorithms for achieving efficient implementations.

These are derived by either considering that each robot localizes using only relative position measurements to a "leader" robot in the team, or by decoupling the problems of orientation and position estimation. In presenting these methods, the trade-offs that exist between localization accuracy and the overhead for communicating and processing relative position measurements are pointed out by the authors. However, no analysis is conducted to reveal the effect of the varying available resources on positioning uncertainty, and no optimal sensing strategies are proposed.

The impact of the geometry of a robot formation on localization accuracy has been addressed in previous work. Specifically, the case of a static formation is studied in the work of Zhang et al. [171]. The authors consider formations of robots that receive absolute position measurements, as well as robot-to-robot measurements (i.e., relative range, bearing, or orientation). A study of the structure of the measurement equations shows that the information matrix corresponding to the exteroceptive measurements is a function of the relative positions of the robots, and a gradient-based optimization technique is employed to determine local maxima of the trace of this matrix. However, due to the non-concavity of the objective function, the selected optimization method does not guarantee global optimality of the solution.

The effects of formation geometry in the case of *moving* robots are studied in [60]. In that work, an evolutionary optimization algorithm is proposed for determining the optimal relative positions of the robots in a moving formation. The optimality criterion is the steady-state position uncertainty of the team, and it is shown that genetic algorithm-based minimization is an appropriate tool for this problem, due to the existence of multiple local minima in the objective function. In [77], a robot team comprised of one master and two slave robots is considered, and a portable landmarks-based technique is adopted for localization, i.e., at each time instant at least one robot remains stationary. The authors propose a method for determining the optimal relative positions between the robots, and identify configurations that yield the maximum possible localization accuracy at the end of a straight-line path.

We note that in all aforementioned approaches the constraints imposed by the available computational and communication resources are not taken into consideration. Our work is

more closely related to work in the sensor networks community that aims at determining the optimal scheduling of measurements received by a *static* set of sensors, in order to attain the best possible estimation results. Representative examples of this line of research can be found in [6, 23, 94], while a similar analysis, in the context of designing observers for dynamical systems, is presented in [9, 81, 143, 147]. The defining assumption in all these cases is that the observation model switches sequentially between modes determined by the candidate subsets of sensors, a finite number of times during a certain time interval. This problem amounts to determining the optimal measurement ordering, so as to maximize the achieved estimation accuracy and/or minimize the consumed energy [6]. For this problem, tree-search algorithms (e.g., [23, 94]), as well as optimization methods in the continuous domain (e.g., [9, 81, 143, 147]), have been proposed.

The main limitation of these approaches, which consider a *finite* time-horizon (or equivalently a finite number of measurements), is that the complexity of determining the optimal sensing strategy increases, often exponentially (e.g., in tree-search algorithms) as the time-span of the sensor operation increases. In contrast, in our work we consider the frequencies of the measurements as the design variables, and we are interested in the steady-state estimation accuracy. The benefit of this formulation is that the optimal strategy has to be determined only once, potentially off-line, for a given spatial configuration of the sensors, and the computational cost of determining the optimal solution is independent of the time duration of the sensors' operation.

A formulation of the scheduling problem that also considers the infinite time-horizon problem has been presented in [54, 55]. In that work, a probability density function is employed to describe the time instants at which each measurement is performed. An upper bound on the expected steady-state covariance of the target's position estimate is then computed as a function of the pdf's parameters. Employing a numerical optimization routine, it is possible to minimize this upper bound, and the resulting pdf is used as the optimal sensing strategy. Despite its mathematical elegance, this approach only aims at optimizing an upper bound. Since no means of determining the looseness of the bound are available, we cannot have any guarantee of optimality, or a measure of suboptimality, when this method is used.

Our work differs from the aforementioned approaches, in that we consider a team of robots that *move* while maintaining their formation, and localize in a global coordinate frame. The steady-state covariance matrix of the robots' localization is expressed as a function of the frequencies of all the exteroceptive measurements, and we seek to select the optimal frequencies, in order to attain the best possible positioning accuracy for the team. The constraints imposed by the available resources are taken into account, and their effect on the attainable localization accuracy is examined.

3.4 The asymptotic uncertainty of CL in robot formations

In this section, we present our approach for characterizing the asymptotic uncertainty during CL in robot formations. We consider a team of N robots that move in formation, employing a suitable control strategy in order to maintain a constant heading and constant relative positions among them. The spatial configuration of the robots is assumed to be given, defined, for example, by the application at hand. All robots are equipped with proprioceptive sensors (such as wheel encoders) that measure their translational and rotational velocities at every time-step. Additionally, some (or all) of the robots are equipped with exteroceptive sensors that enable them to measure quantities such as: (i) distance between two robots, (ii) relative bearing between two robots, (iii) relative orientation between two robots, (iv) absolute position of a robot, and (v) absolute orientation of a robot. The measurements received from all the sensors are processed using an EKF, in order to estimate the pose of the robots with respect to a global frame of reference.

Clearly, due to cost, reliability, or other design considerations, it may not be desirable for all robots to be equipped with identical sensors. This potential heterogeneity of the team is incorporated naturally in our approach, under the restriction that at least one robot has access, at least intermittently, to absolute position information, such as those provided by a GPS receiver or from observing previously mapped features. This constraint is imposed because our goal is to minimize the steady-state localization uncertainty of the robots in a global coordinate frame (cf. Section 3.5). As shown in the previous chapter, when no absolute position information is available to a robot team, the system is unobservable, and at steady state, the uncertainty of the robots continuously increases. The assumption for the

availability of absolute positioning information could be removed if we studied a scenario in which only relative localization was sought. For that scenario, robot-to-robot measurements would (under certain observability conditions) be sufficient, in order to attain a bounded steady-state error covariance, and our approach would be applicable.

Since the processing, communication, and power resources allocated for localization are inevitably limited, it may not be possible to process all available exteroceptive measurements at every time instant. We here assume that measurements can be processed at a maximum total rate of f_{total} throughout the robot team, and our goal is to determine the frequency at which *each individual sensor* should be utilized, in order to attain the highest possible localization accuracy. Before describing the details of our method for obtaining the optimal measurement frequencies, we now present the system and measurement models used for pose estimation.

3.4.1 EKF propagation

Consider N non-holonomic robots moving in 2D. The discrete-time kinematic equations for the i-th robot are:

$$x_i(k+1) = x_i(k) + V_i(k)\delta t \cos(\phi_i(k)) \tag{3.1}$$

$$y_i(k+1) = y_i(k) + V_i(k)\delta t \sin(\phi_i(k)) \tag{3.2}$$

$$\phi_i(k+1) = \phi_i(k) + \omega_i(k)\delta t, \quad i = 1 \dots N \tag{3.3}$$

where $V_i(k)$ and $\omega_i(k)$ denote the translational and rotational velocity of the i-th robot at time-step k, respectively, and δt is the odometry sampling period. Note that, in contrast to the position-only estimator employed in the analysis of Section 2.4, we here use the full-state EKF estimator, where the state comprises all robots' positions *and* orientations. The pose estimates of robot i are propagated using the measurements from its proprioceptive sensors:

$$\hat{x}_{i_{k+1|k}} = \hat{x}_{i_{k|k}} + V_{m_i}(k)\delta t \cos(\hat{\phi}_{i_{k|k}}) \tag{3.4}$$

$$\hat{y}_{i_{k+1|k}} = \hat{y}_{i_{k|k}} + V_{m_i}(k)\delta t \sin(\hat{\phi}_{i_{k|k}}) \tag{3.5}$$

$$\hat{\phi}_{i_{k+1|k}} = \hat{\phi}_{i_{k|k}} + \omega_{m_i}(k)\delta t, \quad i = 1 \dots N \tag{3.6}$$

where $V_{m_i}(k)$ and $\omega_{m_i}(k)$ are the measurements of the robot's translational and rotational velocity, respectively. By linearizing (3.1) - (3.3) the error propagation equation for the robot's pose is derived:

$$\begin{bmatrix} \widetilde{x}_{i_{k+1|k}} \\ \widetilde{y}_{i_{k+1|k}} \\ \widetilde{\phi}_{i_{k+1|k}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -V_{m_i}(k)\delta t \sin(\hat{\phi}_{i_{k|k}}) \\ 0 & 1 & V_{m_i}(k)\delta t \cos(\hat{\phi}_{i_{k|k}}) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \widetilde{x}_{i_{k|k}} \\ \widetilde{y}_{i_{k|k}} \\ \widetilde{\phi}_{i_{k|k}} \end{bmatrix} + \begin{bmatrix} \delta t \cos(\hat{\phi}_{i_{k|k}}) & 0 \\ \delta t \sin(\hat{\phi}_{i_{k|k}}) & 0 \\ 0 & \delta t \end{bmatrix} \begin{bmatrix} w_{V_i}(k) \\ w_{\omega_i}(k) \end{bmatrix}$$

$$\Leftrightarrow \quad \widetilde{\mathbf{X}}_{i_{k+1|k}} = \mathbf{\Phi}_i(k)\widetilde{\mathbf{X}}_{i_{k|k}} + \mathbf{G}_i(k)\mathbf{w}_i(k) \tag{3.7}$$

where $w_{V_i}(k)$ and $w_{\omega_i}(k)$ are white, zero-mean, Gaussian and uncorrelated noise sequences of variance $\sigma^2_{V_i}$ and $\sigma^2_{\omega_i}$ affecting the translational and rotational velocity measurements, respectively.

Considering that the robot team moves in a predefined formation, all robots are required to head towards the same direction, and move with the same velocity, both of which are known constants. Assuming that a motion controller is used in order to minimize the deviations from the desired formation, and that the accuracy of the velocity measurements and orientation estimates is sufficiently high, we can replace the quantities $V_{m_i}(k)$, $\omega_{m_i}(k)$, and $\hat{\phi}_{i_{k|k}}$ in the above expressions by their respective predefined values, V_o, ω_o and ϕ_o. Thus the time-varying matrices $\mathbf{\Phi}_i(k)$ and $\mathbf{G}_i(k)$ can be approximated by the *constant* matrices:

$$\mathbf{\Phi}_i(k) \simeq \begin{bmatrix} 1 & 0 & -V_o\delta t \sin(\phi_o) \\ 0 & 1 & V_o\delta t \cos(\phi_o) \\ 0 & 0 & 1 \end{bmatrix} = \mathbf{\Phi}_{o_i} \tag{3.8}$$

and

$$\mathbf{G}_i(k) \simeq \begin{bmatrix} \delta t \cos(\phi_o) & 0 \\ \delta t \sin(\phi_o) & 0 \\ 0 & \delta t \end{bmatrix} = \mathbf{G}_{o_i} \tag{3.9}$$

With this approximation, the error-state covariance propagation equation for the i-th robot

can be written as:

$$\mathbf{P}_{i_{k+1|k+1}} = \mathbf{\Phi}_{o_i} \mathbf{P}_{i_{k+1|k}} \mathbf{\Phi}_{o_i}^T + \mathbf{G}_{o_i} \mathbf{Q}_i \mathbf{G}_{o_i}^T \tag{3.10}$$

where $\mathbf{Q}_i = \text{diag}(\sigma_{V_i}^2, \sigma_{\omega_i}^2)$.

At this point, a comment regarding our selection of the state propagation model is due. In the preceding expressions, a simple non-holonomic kinematic model for the robots' motion is employed, because it is appropriate for the robots used in our experiments (cf. Section 3.6). However, any other motion model could be employed in our analysis, such as that of skid-steered vehicles [5], that of four-wheeled vehicles [15], or a more accurate kinematic model that assumes constant rotational velocity during an integration step [156]. If a different motion model is used, the structure of the Jacobians (cf. (3.8) and (3.9)) will be different, but the approach for determining the optimal measurement frequencies remains unchanged.

The state vector for the entire robot team, $\mathbf{X}$, is defined as the $3N \times 1$ vector comprising the poses of all the robots, $\mathbf{X}_i = [x_i \quad y_i \quad \phi_i]^T,\ i = 1 \ldots N$. Therefore, the covariance propagation equation can be written as:

$$\mathbf{P}_{k+1|k} = \mathbf{\Phi}_o \mathbf{P}_{k|k} \mathbf{\Phi}_o^T + \mathbf{Q}_o \tag{3.11}$$

where $\mathbf{\Phi}_o = \mathbf{Diag}(\mathbf{\Phi}_{o_i})$ and $\mathbf{Q}_o = \mathbf{Diag}(\mathbf{G}_{o_i} \mathbf{Q}_i \mathbf{G}_{o_i}^T)$ are $3N \times 3N$ block-diagonal matrices.

3.4.2 EKF update

The robots of the team employ the measurements recorded by their exteroceptive sensors for EKF updates. In general, these measurements are described by nonlinear equations (cf. (2.7)), and thus linearization is employed (cf. (2.9)). The Jacobians of the measurement models (cf. (2.10)) are generally time varying, due to the dependence on the state estimates. However, for many practical observation models in robotics, the Jacobians are only functions of the robots' orientations and the relative poses between robots, both of which are, in the case of formation motion, approximately constant. We can thus employ the *constant*

approximation

$$\mathbf{H}_i(k) = \nabla_{\mathbf{X}(k)}\mathbf{h}(\mathbf{X}(k))\Big|_{\hat{\mathbf{X}}_{k|k-1}} \simeq \nabla_{\mathbf{X}(k)}\mathbf{h}(\mathbf{X}(k))\Big|_{\mathbf{X}_o(k)} = \mathbf{H}_{i_o} \tag{3.12}$$

where $\mathbf{X}_o(k)$ is the desired state of the formation at time-step k. To demonstrate this based on concrete examples, we hereafter consider five types of exteroceptive measurements:

Relative range measurements

If robot i is equipped with a sensor capable of measuring the distance of other robots with respect to itself, such as a laser scanner, then the distance measurement between robots i and j is:

$$z_{\rho_{ij}}(k) = \sqrt{\Delta x_{ij}(k)^2 + \Delta y_{ij}(k)^2} + n_{\rho_{ij}}(k) \tag{3.13}$$

where $\Delta x_{ij}(k) = x_j(k) - x_i(k)$, $\Delta y_{ij}(k) = y_j(k) - y_i(k)$, and $n_{\rho_{ij}}(k)$ is a white, zero-mean, Gaussian noise process, whose standard deviation, σ_{ρ_i}, is determined by the characteristics of the sensor. By linearizing (3.13), the measurement residual equation is determined:

$$\begin{aligned} r_{\rho_{ij}}(k) &\simeq \mathbf{H}_{\rho_{ij}}(k)\widetilde{\mathbf{X}}_{k|k-1} + n_{\rho_{ij}}(k) \\ &= \begin{bmatrix} 0 & .. & \mathbf{H}_{\rho_i}(k) & .. & \mathbf{H}_{\rho_j}(k) & .. & 0 \end{bmatrix} \widetilde{\mathbf{X}}_{k|k-1} + n_{\rho_{ij}} \end{aligned} \tag{3.14}$$

where $\mathbf{H}_{\rho_{ij}}(k)$ is a $1 \times 3N$ matrix, whose i-th and j-th block elements are, respectively:

$$\mathbf{H}_{\rho_j}(k) \;=\; -\mathbf{H}_{\rho_i}(k) \;=\; \begin{bmatrix} \frac{\widehat{\Delta x}_{ij}(k)}{\hat{\rho}_{ij}(k)} & \frac{\widehat{\Delta y}_{ij}(k)}{\hat{\rho}_{ij}(k)} & 0 \end{bmatrix} \tag{3.15}$$

In the preceding expression, $\widehat{\Delta x}_{ij}(k)$, $\widehat{\Delta y}_{ij}(k)$ and $\hat{\rho}_{ij}(k)$ represent the estimated differences in the x and y coordinates, and the estimated distance between robots i and j, respectively. By replacing the estimates with the values corresponding to the desired formation of the robots, we can derive the following approximations:

$$\mathbf{H}_{\rho_i}(k) \;\simeq\; \begin{bmatrix} \frac{-\Delta x_{ij_o}}{\rho_{ij_o}} & \frac{-\Delta y_{ij_o}}{\rho_{ij_o}} & 0 \end{bmatrix} \;=\; \mathbf{H}_{\rho_{i_o}} \tag{3.16}$$

$$\mathbf{H}_{\rho_j}(k) \;\simeq\; \begin{bmatrix} \frac{\Delta x_{ij_o}}{\rho_{ij_o}} & \frac{\Delta y_{ij_o}}{\rho_{ij_o}} & 0 \end{bmatrix} \;=\; \mathbf{H}_{\rho_{j_o}} \tag{3.17}$$

For practical reasons, it may not be possible for all robots to measure relative distances to all other robots in the team. For example, some robots may not be equipped with range sensors, or certain measurements may be impossible due to occlusions in the formation. In order to describe the set of all possible measurements we define the set

$$\mathcal{H}_\rho = \{\mathbf{H}_{\rho_{ij}} | \text{ robot } i \text{ can measure range to robot } j\}$$

Relative bearing measurements

Assuming robot i measures the relative bearing towards robot j, the corresponding measurement equation is:

$$z_{\theta_{ij}}(k) = \text{Atan2}(\Delta y_{ij}(k), \Delta x_{ij}(k)) - \phi_i(k) + n_{\theta_{ij}}(k) \tag{3.18}$$

where $n_{\theta_{ij}}(k)$ is a white, zero-mean, Gaussian noise process, with standard deviation σ_{θ_i}. Linearization yields the following measurement residual equation:

$$\begin{aligned} r_{\theta_{ij}}(k) &\simeq \mathbf{H}_{\theta_{ij}}(k)\widetilde{\mathbf{X}}_{k|k-1} + n_{\theta_{ij}}(k) \\ &\simeq \mathbf{H}_{\theta_{ijo}}\widetilde{\mathbf{X}}_{k|k-1} + n_{\theta_{ij}}(k) \\ &= \begin{bmatrix} 0 & .. & \mathbf{H}_{\theta_{i_o}} & .. & \mathbf{H}_{\theta_{j_o}} & .. & 0 \end{bmatrix} \widetilde{\mathbf{X}}_{k|k-1} + n_{\theta_{ij}} \end{aligned} \tag{3.19}$$

where we have once again approximated the time-varying position estimates with their constant, desired values. Note that $\mathbf{H}_{\theta_{ij_o}}$ is a $1 \times 3N$ matrix, whose i-th and j-th block elements are, respectively:

$$\mathbf{H}_{\theta_{i_o}} = \begin{bmatrix} \frac{\Delta y_{ij_o}}{\rho^2_{ij_o}} & \frac{-\Delta x_{ij_o}}{\rho^2_{ij_o}} & -1 \end{bmatrix} \tag{3.20}$$

$$\mathbf{H}_{\theta_{j_o}} = \begin{bmatrix} \frac{-\Delta y_{ij_o}}{\rho^2_{ij_o}} & \frac{\Delta x_{ij_o}}{\rho^2_{ij_o}} & 0 \end{bmatrix} \tag{3.21}$$

Similarly to the case of range measurements, we describe all possible bearing measurements with the set

$$\mathcal{H}_\theta = \{\mathbf{H}_{\theta_{ij}} | \text{ robot } i \text{ can measure bearing to robot } j\}$$

Relative orientation measurements

If robot i measures the relative orientation of robot j, the corresponding measurement equation is:

$$z_{\Delta\phi_{ij}}(k) \quad = \quad \phi_j(k) - \phi_i(k) + n_{\Delta\phi_{ij}}(k) \tag{3.22}$$

where $n_{\Delta\phi_{ij}}(k)$ is a white, zero-mean, Gaussian noise process, with standard deviation $\sigma_{\Delta\phi_i}$. The residual equation is:

$$\begin{aligned} r_{\Delta\phi_{ij}}(k) &= \mathbf{H}_{\Delta\phi_{ij}}\widetilde{\mathbf{X}}_{k|k-1} + n_{\Delta\phi_{ij}}(k) \\ &= \begin{bmatrix} .. & \underbrace{[0\ 0\ -1]}_{i\text{th block}} & .. & \underbrace{[0\ 0\ 1]}_{j\text{th block}} & .. \end{bmatrix} \widetilde{\mathbf{X}}_{k|k-1} + n_{\Delta\phi_{ij}}(k) \end{aligned}$$

All possible relative orientation measurements are described by the set

$$\mathcal{H}_{\Delta\phi} = \{\mathbf{H}_{\Delta\phi_{ij}} | \text{ robot } i \text{ can measure rel. ori. of robot } j\}$$

Absolute orientation measurements

If the i-th robot of the team is equipped with an absolute orientation sensor, such as a compass, the corresponding measurement equation is:

$$z_{\phi_i}(k) = \phi_i(k) + n_{\phi_i}(k) \tag{3.23}$$

where n_{ϕ_i} is a white, zero-mean, Gaussian noise process, with standard deviation σ_{ϕ_i}. In this case the residual is:

$$\begin{aligned} r_{\phi_i}(k) &= \mathbf{H}_{\phi_i}\widetilde{\mathbf{X}}_{k|k-1} + n_{\phi_i}(k) \\ &= \begin{bmatrix} 0 & .. & \underbrace{[0\ \ 0\ \ 1]}_{i\text{th block}} & .. & 0 \end{bmatrix} \widetilde{\mathbf{X}}_{k|k-1} + n_{\phi_i}(k) \end{aligned} \tag{3.24}$$

All possible absolute orientation measurements are described by the set

$$\mathcal{H}_{\phi} = \{\mathbf{H}_{\phi_i} | \text{ robot } i \text{ can measure absolute orientation}\}$$

Absolute position measurements

In this work, the robots localize with respect to a global coordinate frame. Therefore, in order for the position errors to remain bounded for all times, it is necessary that at least one of the robots has access to absolute position measurements. The measurement equation for the i-th robot is

$$\mathbf{z}_{p_i}(k) = \begin{bmatrix} x_i(k) \\ y_i(k) \end{bmatrix} + \mathbf{n}_{p_i}(k) \tag{3.25}$$

where $\mathbf{n}_{p_i}(k)$ is a 2×1 white, zero-mean, Gaussian noise process, with covariance matrix $\mathbf{R}_{p_i}$. The measurement error equation for this type of measurement is

$$\begin{aligned} \mathbf{r}_{p_i}(k) &= \mathbf{H}_{p_i} \widetilde{\mathbf{X}}_{k|k-1} + \mathbf{n}_{p_i}(k) \\ &= \begin{bmatrix} \mathbf{0}_{2\times 3} & \cdots & \underbrace{\begin{bmatrix} \mathbf{I}_2 & \mathbf{0}_{2\times 1} \end{bmatrix}}_{i\text{th block}} & \cdots & \mathbf{0}_{2\times 3} \end{bmatrix} \widetilde{\mathbf{X}}_{k|k-1} + \mathbf{n}_{p_i}(k) \end{aligned}$$

where $\mathbf{H}_{p_i}$ is a $2 \times 3N$ matrix.

In order to describe all possible absolute position measurements we define the set

$$\mathcal{H}_p = \{\mathbf{H}_{p_i} | \text{ robot } i \text{ can measure absolute position}\}$$

3.4.3 Discrete-time evolution of the covariance

The time evolution of the covariance of the EKF state estimates is described by the Riccati recursion:

$$\mathbf{P}_{k+2|k+1} = \mathbf{\Phi}_o \left(\mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k} \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_{k+1|k} \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1} \mathbf{H}_k \mathbf{P}_{k+1|k} \right) \mathbf{\Phi}_o^T + \mathbf{Q}_o \tag{3.26}$$

where $\mathbf{H}_k$ is the measurement matrix for the system at time-step k, and $\mathbf{R}_k$ is the corresponding measurement-noise covariance matrix. These matrices will be block matrices containing the Jacobians and covariance matrices of all the measurements occurring at time-step k. It is important to observe that, when only a subset of sensor measurements, often varying, can be processed at each time instant, $\mathbf{H}_k$ and $\mathbf{R}_k$ will not remain constant,

and will possibly vary even in size at each time-step. Specifically, if at time-step k a total of m_k measurements are performed, $\mathbf{H}_k$ will comprise m_k block rows belonging in the set $\mathcal{H} = \mathcal{H}_\rho \bigcup \mathcal{H}_\theta \bigcup \mathcal{H}_{\Delta\phi} \bigcup \mathcal{H}_\phi \bigcup \mathcal{H}_p$, and $\mathbf{R}_k$ will be a block-diagonal matrix whose elements can be defined accordingly.

Because the system is observable, after undergoing an initial transient phase the covariance matrix will enter a steady state, where its elements will fluctuate around some mean value (cf. Fig. 3.1). Had we been able to provide a description of this mean value as a function of the measurement frequencies, then we would have a means of directly relating the localization performance of the system to these frequencies. However, the Riccati recursion (3.26) has time-varying coefficients, and thus such an analysis cannot be conducted based on it. To address this problem, we utilize a transition from the discrete-time system model to a continuous-time one, as described in the following.

3.4.4 The Riccati differential equation

Intuitively, the rate at which a given sensor is providing measurements determines the amount of localization information this sensor contributes per unit of time. If we view this as a continuous information flow, then the frequency of the measurements determines the magnitude of the flow. This key idea allows us to express the steady-state localization accuracy of the robots as an analytical function of the measurement frequencies, by employing a transition to the continuous-time domain.

In particular, in [91] it is shown that given a discrete-time Riccati recursion, we can derive a continuous-time Riccati differential equation that is equivalent, in the sense that the state estimates' accuracy in both cases is the same. Specifically, if state observations whose covariance is $\mathbf{R}_d$ are performed with frequency f in the discrete-time description, then the equivalent continuous-time measurements' covariance function is $E\{\mathbf{n}_c(t)\mathbf{n}_c^T(\tau)\} = \mathbf{R}_c\delta(t-\tau)$, where $\mathbf{n}_c(\cdot)$ is a white Gaussian noise process, $\delta(\cdot)$ denotes the Dirac delta function, and[1] $\mathbf{R}_c = f^{-1}\mathbf{R}_d$. We observe that the covariance matrix of the continuous-time model is scaled by the inverse measurement frequency, to ensure a constant information influx. By a similar argument, we can derive the appropriate value of the system noise

[1]The subscripts c and d denote continuous- and discrete-time quantities, respectively.

covariance matrix.

We now employ the idea of deriving an equivalent continuous-time Riccati, in order to formulate a *constant coefficient* differential equation for the covariance of the pose estimates in the robot team. Specifically, since each of the measurements in the set $\mathcal{H}$ occurs at a constant frequency (generally different for each measurement), we can formulate a continuous-time model, where all the measurements occur continuously, and the covariance of each measurement is scaled by the inverse of its frequency. In the continuous-time formulation, the measurement matrix $\mathbf{H}_c$ will be a constant matrix comprising all the block rows in the set $\mathcal{H}$. The covariance matrix of the measurements, $\mathbf{R}_c$, will be a (block) diagonal matrix, with elements the weighted covariances of the discrete-time measurements. For example, if robot i receives absolute orientation measurements with covariance σ_{ϕ_i} at a rate of f_{ϕ_i}, then the continuous-time covariance function corresponding to this measurement is $R_{\phi_{i_c}}\delta(t-\tau)$, where

$$R_{\phi_{i_c}} = \sigma^2_{\phi_{i_c}} = \frac{\sigma^2_{\phi_{i_d}}}{f_{\phi_i}} = \frac{1}{f_{\phi_i}} R_{\phi_{i_d}} \tag{3.27}$$

We can now use the Riccati differential equation in order to describe the time evolution of the covariance of the robots' pose estimates. We note that the state transition matrix for the system in continuous time is equal to $\mathbf{F}_c = \mathbf{Diag}(\mathbf{F}_o)$, where

$$\mathbf{F}_o = \begin{bmatrix} 0 & 0 & -V_o \sin(\phi_o) \\ 0 & 0 & V_o \cos(\phi_o) \\ 0 & 0 & 0 \end{bmatrix} \tag{3.28}$$

while the matrix describing the influx of uncertainty in the continuous-time system is equal to $\mathbf{Q}_c = \mathbf{Diag}(\mathbf{G}_{o_c}\mathbf{Q}_{i_c}\mathbf{G}^T_{o_c})$ with

$$\mathbf{G}_{o_c} = \begin{bmatrix} \cos(\phi_o) & 0 \\ \sin(\phi_o) & 0 \\ 0 & 1 \end{bmatrix} \tag{3.29}$$

and $\mathbf{Q}_{i_c} = f^{-1}_{o_i} \operatorname{diag}(\sigma^2_{V_i}, \sigma^2_{\omega_i})$. In this last expression, f_{o_i} denotes the rate at which robot i

samples its proprioceptive sensors. Using the previous relations, the Riccati differential equation is written as

$$\dot{\mathbf{P}}(t) \;=\; \mathbf{F}_c\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}_c^T + \mathbf{Q}_c - \mathbf{P}(t)\mathbf{C}\mathbf{P}(t) \tag{3.30}$$

where we have denoted

$$\mathbf{C} \;=\; \mathbf{H}_c^T\mathbf{R}_c^{-1}\mathbf{H}_c \tag{3.31}$$

The first two terms in (3.30) describe the effect of the dynamics of the system on the state covariance matrix, the term $\mathbf{Q}_c$ accounts for the increase in uncertainty due to the existence of system noise, while the term $\mathbf{P}(t)\mathbf{C}\mathbf{P}(t)$ describes the influx of localization information due to the exteroceptive measurements. If we denote by M the total number of available exteroceptive measurements (i.e., the number of elements in $\mathcal{H}$), by f_i the frequency of the i-th measurement in $\mathcal{H}$, by $\mathbf{H}_i$ the corresponding measurement matrix, and by $\mathbf{R}_{d_i}$ the associated covariance matrix, then $\mathbf{C}$ can be rewritten as

$$\mathbf{C} \;=\; \sum_{i=1}^{M} f_i\mathbf{H}_i^T\mathbf{R}_{d_i}^{-1}\mathbf{H}_i \;=\; \sum_{i=1}^{M} f_i\mathbf{C}_i \tag{3.32}$$

We can therefore see that the elements of $\mathbf{C}$ are linear combinations of the measurement frequencies. This is an important observation, because it allows us to express the problem of determining the optimal measurement frequencies as a convex optimization problem, as shown in the next section.

We note that the Riccati equation in (3.30) is a constant-coefficient differential equation, and its steady-state solution, $\mathbf{P}_{ss}$, can be found by solving the Algebraic Riccati Equation (ARE)

$$\mathbf{F}_c\mathbf{P}_{ss} + \mathbf{P}_{ss}\mathbf{F}_c^T + \mathbf{Q}_c - \mathbf{P}_{ss}\mathbf{C}\mathbf{P}_{ss} = \mathbf{0} \tag{3.33}$$

The solution is a function of the matrix coefficients of the ARE [14], and therefore the steady-state covariance of the pose estimates for the robots of the formation is a *function of the measurement frequencies*, which appear in $\mathbf{C}$. To be more precise, $\mathbf{P}_{ss}$ is the steady-

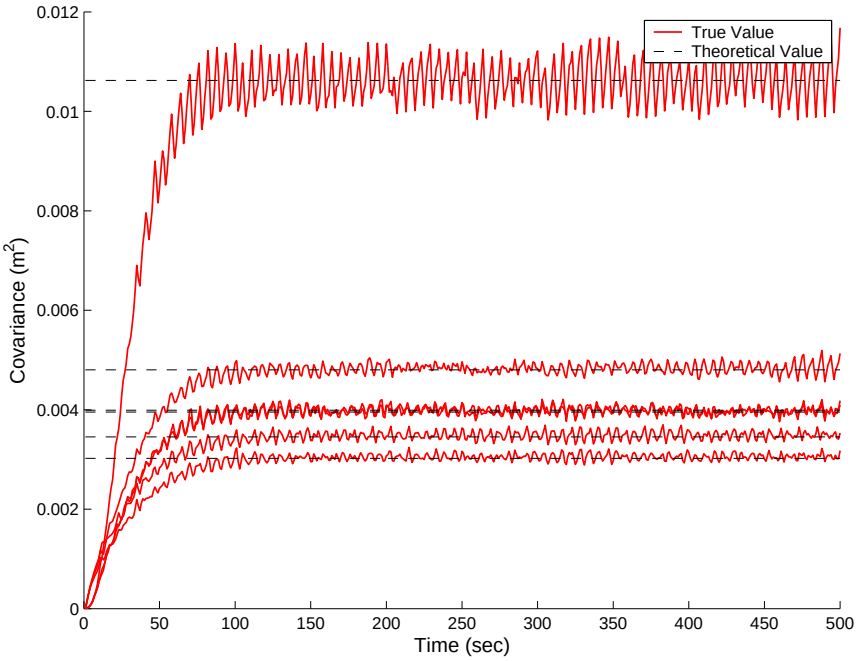


Figure 3.1: True covariance vs. theoretical values. The diagonal elements of the covariance matrix corresponding to the position of the 3 robots are plotted.

state covariance of the equivalent continuous-time system, whose parameters depend on the measurement frequencies. In Fig. 3.1, we present the time-evolution of the diagonal elements of the covariance matrix in the actual discrete-time system (solid lines) and compare them to the theoretical continuous-time values (dashed lines) computed by solving (3.33). For these simulations, a team of 3 robots, that have access to all four types of exteroceptive measurements, discussed in Section 3.4.2, was considered. The relative positions, as well as the measurement frequencies for all robots were selected randomly.

It becomes clear that, at steady state, the actual values of the covariance fluctuate around the theoretically predicted values. Thus, we can employ the continuous-time analysis in order to study the properties of the localization accuracy in the formation.

3.5 Measurement frequency optimization

In this section, we formulate the problem of determining the optimal measurement frequencies as a convex optimization problem. Our goal is to determine the optimal frequencies for all available measurements, i.e., those frequencies that will yield the best possible localiza-

tion results under given constraints. Clearly, in order to improve the localization accuracy of the formation, the steady-state covariance matrix should be minimized. However, $\mathbf{P}_{ss}$ is a $3N \times 3N$ matrix, and several criteria of optimality can be defined based on it (e.g., its determinant, its maximum eigenvalue, or its trace). A difficulty that arises is that while the elements of $\mathbf{P}_{ss}$ that correspond to the position estimates of the robots have units of m^2, the elements that correspond to orientation have units of rad^2. Clearly, we cannot treat these two types of elements equally. One approach is to introduce a weighting matrix $\mathbf{W}$, and try to minimize a function of the weighted matrix $\mathbf{W}\mathbf{P}_{ss}\mathbf{W}^T$. However, any selection of $\mathbf{W}$ that would incorporate both the orientation and the position uncertainty in the objective function would be *ad-hoc* and thus difficult to motivate. We have therefore selected to focus only on the diagonal elements of $\mathbf{P}_{ss}$ that correspond to the position estimates of the robots, while ensuring that the orientation uncertainty of each robot does not exceed a threshold ϵ_ϕ (this is necessary, in order to guarantee small linearization errors). We thus formulate the following optimization problem:

$$\begin{aligned}
\textbf{minimize} \quad & \text{trace}(\mathbf{W}_p\mathbf{P}_{ss}\mathbf{W}_p^T) \\
\textbf{subject to} \quad & \mathbf{F}_c\mathbf{P}_{ss} + \mathbf{P}_{ss}\mathbf{F}_c^T + \mathbf{Q}_c - \mathbf{P}_{ss}\mathbf{C}\mathbf{P}_{ss} = \mathbf{0} \\
& \mathbf{C} = \sum_{i=1}^{M} f_i\mathbf{C}_i \\
& 0 \leq f_i \leq f_{i_{\max}}, \text{for } i = 1 \ldots M \\
& \sum_{i=1}^{M} f_i \leq f_{\text{total}} \\
& \mathbf{e}_{3i}^T\mathbf{P}_{ss}\mathbf{e}_{3i} \leq \epsilon_\phi, \text{for } i = 1 \ldots N
\end{aligned} \tag{3.34}$$

In the preceding expressions $\mathbf{e}_i$ denotes the i-th canonical basis vector in the $3N$-dimensional space, and the weighting matrix is defined as

$$\mathbf{W}_p = \sum_{i=1}^{N} \left(\mathbf{e}_{3i-2}\mathbf{e}_{3i-2}^T + \mathbf{e}_{3i-1}\mathbf{e}_{3i-1}^T\right) \tag{3.35}$$

This definition means that the objective function is the sum of all the diagonal elements of $\mathbf{P}_{ss}$ that correspond to the positions of the robots. The linear constraints on the mea-

surement frequencies express the facts that: (i) each sensor has a maximum sampling rate, $f_{i_{\max}}$, that cannot be exceeded, and (ii) the total frequency of the measurements cannot exceed a threshold, f_{total}, which is determined by the available communication and computational resources. We note that more general constraints can be incorporated in this formulation. For example, different types of measurements may have different costs associated with them, and this can be easily taken into consideration, by introducing weights for each of their frequencies. Additionally, if the positioning accuracy of some robots in the team is of higher importance than that of others, this can be easily taken into account by introducing weights, i.e., by defining a weighting matrix of the form

$$\mathbf{W}_{wp} = \sum_{i=1}^{N} w_i \left(\mathbf{e}_{3i-2}\mathbf{e}_{3i-2}^T + \mathbf{e}_{3i-1}\mathbf{e}_{3i-1}^T \right) \tag{3.36}$$

For clarity of presentation, the case of equal weights for all robots and all frequencies will be considered here.

In [45], it is shown that the steady-state solution of the ARE in (3.33) is a convex function of the matrix $\mathbf{C}$. Because the elements of $\mathbf{C}$ are linear functions of the measurement frequencies (cf. (3.32)), we conclude that $\mathbf{P}_{ss}$ is a *convex function of the measurement frequencies*. As a result, the optimization problem (3.34) is a convex one (the objective is a convex function, and the feasible set is convex). This is a very important property, because it guarantees that the problem has a unique global minimum which can be found using standard gradient-based optimization techniques [12]. Moreover, we can reformulate the above optimization problem as a Semi-Definite Programming (SDP) problem:

Lemma 11 *The original problem in (3.34) is equivalent to the following one:*

$$\begin{aligned}
\textbf{minimize} \quad & \operatorname{trace}(\mathbf{W}_p \mathcal{P} \mathbf{W}_p^T) \\
\textbf{subject to} \quad & \begin{bmatrix} \mathcal{P} & \mathbf{I}_{3N} \\ \mathbf{I}_{3N} & \mathbf{J} \end{bmatrix} \succeq \mathbf{0} \\
& \begin{bmatrix} -\mathbf{J}\mathbf{F}_c - \mathbf{F}_c^T\mathbf{J} + \sum_{i=1}^{M} f_i \mathbf{C}_i & \mathbf{J}\mathbf{Q}_c^{1/2} \\ \mathbf{Q}_c^{1/2}\mathbf{J} & \mathbf{I}_{3N} \end{bmatrix} \succeq \mathbf{0}
\end{aligned}$$

$$
\begin{aligned}
& 0 \leq f_i \leq f_{i_{\max}}, \text{for } i = 1 \dots M \\
& \sum_{i=1}^{M} f_i \leq f_{\text{total}} \\
& \mathbf{e}_{3i}^T \mathcal{P} \mathbf{e}_{3i} \leq \epsilon_\phi, \text{for } i = 1 \dots N
\end{aligned}
\tag{3.37}
$$

where the variables in this problem are the matrices $\mathcal{P}$ and $\mathbf{J}$, belonging to the positive semidefinite cone $\mathbf{S}_+^{3N}$, and the measurement frequencies, f_i, $i = 1 \dots M$.

Proof:

We first note that by employing the properties of the Schur complement, the first inequality constraint in problem (3.37) *is written as:*

$$
\mathcal{P} \succeq \mathbf{J}^{-1} \tag{3.38}
$$

while the second matrix inequality is equivalent to:

$$
-\mathbf{J}\mathbf{F}_c - \mathbf{F}_c^T \mathbf{J} + \sum_{i=1}^{M} f_i \mathbf{C}_i - \mathbf{J}\mathbf{Q}_c\mathbf{J} \quad \succeq \quad \mathbf{0} \tag{3.39}
$$

Using these relations, problem (3.37) is written equivalently as

$$
\begin{aligned}
\textbf{minimize} \quad & \text{trace}(\mathbf{W}_p \mathcal{P} \mathbf{W}_p^T) \\
\textbf{subject to} \quad & \mathbf{J}^{-1} - \mathcal{P} \preceq \mathbf{0} \\
& \mathbf{J}\mathbf{F}_c + \mathbf{F}_c^T \mathbf{J} - \left(\sum_{i=1}^{M} f_i \mathbf{C}_i \right) + \mathbf{J}\mathbf{Q}_c\mathbf{J} \preceq \mathbf{0} \\
& 0 \leq f_i \leq f_{i_{\max}}, \text{for } i = 1 \dots M \\
& \sum_{i=1}^{M} f_i \leq f_{\text{total}} \\
& \mathbf{e}_{3i}^T \mathcal{P} \mathbf{e}_{3i} \leq \epsilon_\phi, \text{for } i = 1 \dots N
\end{aligned}
\tag{3.40}
$$

This is a convex optimization problem, since the objective as well as the inequality constraints are convex. Our goal is to show that this problem is equivalent to the problem described in (3.34), in the sense that the optimal frequencies for this problem are also opti-

mal for (3.34).

We observe that for any feasible point, Y, for (3.34), with $Y = (f_1, \ldots, f_M, \mathbf{P}_{ss}) \in \mathbb{R}^M \times \mathbf{S}_+^{3N}$, we can construct the point $\Xi = (f_1, \ldots, f_M, \mathcal{P} = \mathbf{P}_{ss}, \mathbf{J} = \mathbf{P}_{ss}^{-1}) \in \mathbb{R}^M \times \mathbf{S}_+^{3N} \times \mathbf{S}_+^{3N}$ which is also feasible for (3.40), and yields the same *objective value.*

Similarly, given any feasible point for the problem (3.40), we can also construct a feasible point for (3.34). Let $\Xi^\star = (f_1^\star, \ldots, f_M^\star, \mathcal{P}^\star, \mathbf{J}^\star)$ be the optimal solution to the problem (3.40). Then, solving the ARE

$$\mathbf{F}_c \mathbf{P}_{ss}^\star + \mathbf{P}_{ss}^\star \mathbf{F}_c^T + \mathbf{Q}_c - \mathbf{P}_{ss}^\star \left(\sum_{i=1}^{M} f_i^\star \mathbf{C}_i \right) \mathbf{P}_{ss}^\star = \mathbf{0} \tag{3.41}$$

for $\mathbf{P}_{ss}^\star$ yields a feasible point $Y^\star = (f_1^\star, \ldots, f_M^\star, \mathbf{P}_{ss}^\star)$ for the problem described in (3.34). In Appendix B.1 it is shown that the objective value corresponding to $Y^\star$ in (3.34) is equal *to the objective value corresponding to $\Xi^\star$ in (3.40). Using this key result, we can employ proof by contradiction to show that $Y^\star$ is optimal for (3.34). Specifically, if $Y^\star$ were not optimal, there would exist a point $\breve{Y}$ that would give an objective value smaller than that of $Y^\star$. But in that case, we would be able to construct a point $\breve{\Xi}$ for problem (3.40), that would give a smaller objective value than $\Xi^\star$. However, this is a contradiction since $\Xi^\star$ is optimal. Thus, the optimal solution for the measurement frequencies arising from problem (3.40) is also optimal for problem (3.34).*

■

The above proof relies on the fact that the objective value corresponding to $Y^\star$ is equal to the optimal value of problem (3.40). To provide intuition about this key result, whose proof can be found in Appendix B.1, we consider the simple case where the weighting matrix $\mathbf{W}_p$ is replaced by the identity matrix, and thus the minimization objective in (3.40) is simply trace($\mathcal{P}$). We note that since $\mathcal{P}$ is bounded below only by $\mathbf{J}^{-1}$, selecting $\mathcal{P} = \mathbf{J}^{-1}$ yields the minimum cost. Thus, at the optimal solution we have $\mathcal{P}^\star = \mathbf{J}^{\star -1}$, and substitution in (3.39) yields

$$\mathbf{F}_c \mathcal{P}^\star + \mathcal{P}^\star \mathbf{F}_c^T + \mathbf{Q}_c - \mathcal{P}^\star \left(\sum_{i=1}^{M} f_i^\star \mathbf{C}_i \right) \mathcal{P}^\star \preceq \mathbf{0} \tag{3.42}$$

or equivalently,

$$\mathbf{F}_c\mathcal{P}^\star + \mathcal{P}^\star\mathbf{F}_c^T + \mathbf{Q}_c' - \mathcal{P}^\star\left(\sum_{i=1}^{M} f_i^\star \mathbf{C}_i\right)\mathcal{P}^\star = \mathbf{0}, \quad \mathbf{Q}_c' \succeq \mathbf{Q}_c$$

Thus, $\mathcal{P}^\star$ satisfies an ARE with $\mathbf{Q}_c' \succeq \mathbf{Q}_c$. However, the solution of an ARE is a monotonically increasing function of $\mathbf{Q}_c$ [45], and therefore the smallest value of the objective function, trace($\mathcal{P}$), is obtained when $\mathbf{Q}_c'$ is minimum. Clearly, this occurs when $\mathbf{Q}_c' = \mathbf{Q}_c$, thus the optimal solution $\mathcal{P}^\star$ satisfies (3.42) with equality. Note that this ARE is identical to the one in (3.41), hence $\mathcal{P}^\star = \mathbf{P}_{ss}^\star$, which means that the objective values of the two problems are equal. We stress that this proof outline is only valid when $\mathbf{W}_p$ is invertible. This is clearly not the case for the selection of $\mathbf{W}_p$ in (3.35), and this results in a significantly more complicated proof in Appendix B.1. However, the main underlying ideas remain the same.

An interesting remark is that by employing the principle of strong duality, which holds for convex SDPs under mild qualifications that are valid in the particular problem [12], we can obtain a bound for the suboptimality of any solution. In particular, for any convex SDP problem we can define a *dual* SDP maximization problem [12]. When strong duality holds, the optimal solutions to the primal and dual problems yield the same objective value. This implies that, if any solution to the dual problem of (3.37) is available, we immediately have a lower bound on the minimum attainable objective value for (3.37).

Most SDP solvers automatically generate the dual problem, and proceed by simultaneously solving the primal and dual problems in an iterative fashion. Therefore the problem of determining the optimal measurement frequencies is solved by an *any-time algorithm*, since at any point during the solution procedure, a suboptimal solution is available. Moreover, by comparing the objective value of this intermediate solution to that of the corresponding intermediate solution of the dual problem, and employing strong duality, we obtain a concrete measure of "how good" the solution is. In a scenario where a large number of sensors is involved, and in which computation time is a significant factor (e.g., if we are solving in real-time to determine the best sensing strategy in a slowly varying formation), we may wish to trade-off optimality for efficiency, and in this case, the any-time property of the solution algorithm is very important.

At this point, we comment on the applicability of the method to cases where the as-

sumption of a constant formation shape does not hold. A significant property of the solution to the ARE in (3.33) is that it is independent of the initial conditions, since the system under consideration is observable. This implies that if the geometry of the robot formation changes temporarily, for example due to the presence of obstacles that need to be avoided, then, once the robots return to the initial configuration, the solution becomes valid again. For practical purposes, this observation means that if we know in advance that a robot team will move in a known formation most of the time, then it might be desirable, from an implementation point of view, to use the measurement frequencies obtained with the proposed method for the entire duration of the robots' run.

If alternatively, the optimal sequence of measurements for a time-varying formation were sought, a tree-search within a finite time horizon of n time-steps would be necessary [54]. However, the complexity of such a search is exponential in the number of time-steps, and can become intractable even for a search within a short-time horizon, if many measurements are available in the system. Such a search would need to be performed necessarily in real time, employing the most current pose estimates for the robots, and the results would need to be transmitted to all the members of the team. Contrary to that, the proposed method lends itself to off-line execution[2], before the robot team is deployed, and additionally, programming the sensors to record measurements at fixed time intervals is simpler. Clearly, the proposed approach is suboptimal when the robots do not maintain a fixed formation, and its performance has to be evaluated on a case-by-case basis.

3.6 Experimental results

To demonstrate the application of our method for optimal resource allocation, we have conducted experiments with a heterogeneous robot team comprised of one iRobot Packbot robot and 3 Pioneer-I robots. The robots move outdoors in a diamond-shaped formation, where the Packbot is the "leader", as shown in Fig. 3.2. Each of the Pioneers is equipped with a laser scanner, and is able to detect the robots of the team that lie within its field of view. Using a linefitting technique, relative position (i.e., range and bearing) as well

[2]If the geometry of the relative positions of the robots of a team changes slowly, then our algorithm can also be used on-line, to provide an approximate solution to the optimal measurement scheduling.

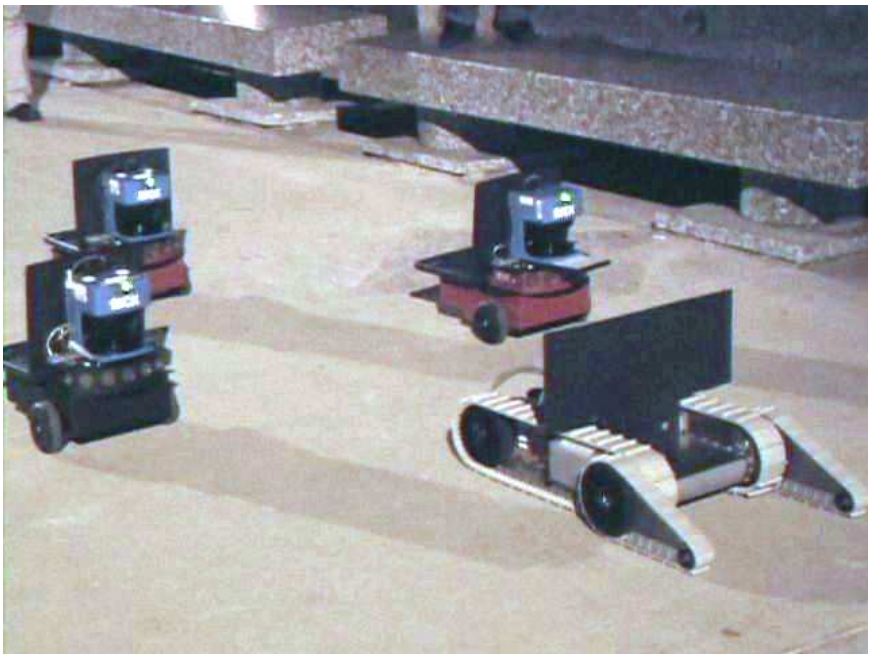


Figure 3.2: The heterogeneous robot team used in our experiments.

as relative orientation measurements are obtained. It is important to note that since the same laser points are used in order to measure the relative position and relative orientation of a particular robot, these measurements are correlated, and must be treated as a single, vector-valued measurement.

In addition to the relative pose measurements, absolute position and orientation measurements are provided to the team by a GPS receiver and a magnetic compass, which are mounted on the Packbot. In total, 5 relative pose measurements (the robot in the rear is able to measure the relative pose of all other robots, while the ones on the sides can only detect the formation leader) and 2 absolute measurements (absolute position and orientation of the Packbot) are available. The absolute measurements are available at a maximum frequency of 1 Hz, while the relative pose measurements are available at a maximum frequency of 3 Hz. In Fig. 3.3, the geometry of the formation is shown, and the available relative pose measurements are presented by the dash-dotted arrows. R_1 is the Packbot, while R_2 - R_4 are the Pioneer robots, and M_{ij} denotes the measurement of the relative pose of robot j with respect to robot i. The formation moves on a 50-meter path parallel to the global x axis at a velocity of $V_o = 0.2$ m/sec. During the experiments the robots keep records of

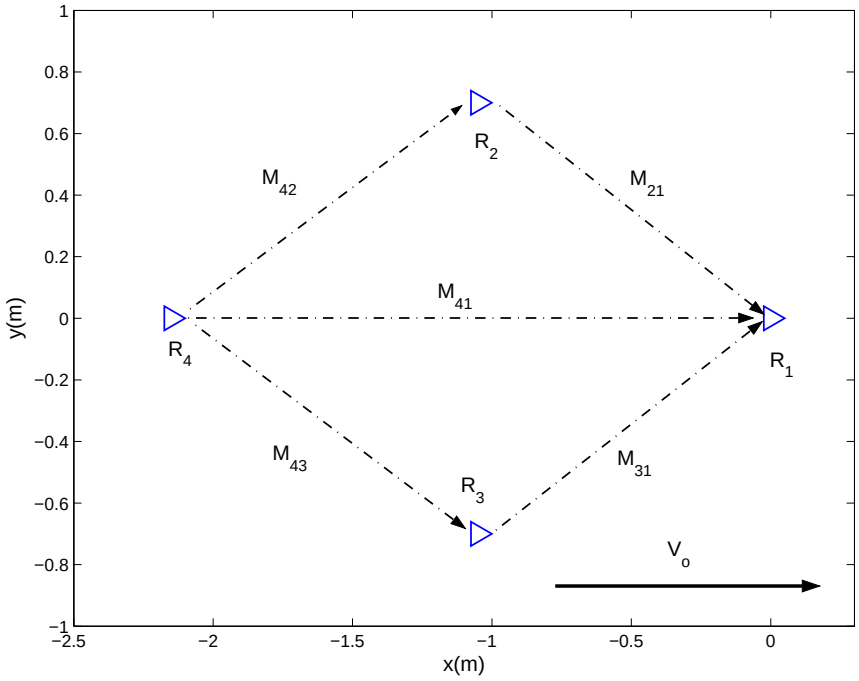


Figure 3.3: Robot formation and motion direction. The dash-dotted arrows represent the relative pose measurements available to the robots.

the raw sensor data, thus enabling us to run the EKF off-line with various measurement frequencies, and facilitating comparison between different sensing strategies.

In order to maintain the desired formation shape, a simple leader-follower control scheme is implemented. Each of the Pioneer-I robots adjusts its rotational and translational velocity using a PI-controller. The feedback input to the controller is the difference between the desired and the measured relative pose of the formation leader with respect to the measuring robot. Since control is performed locally on each robot, it does not introduce any communication overhead, and additionally, it is very inexpensive computationally. Although very simple, this controller is sufficient for the purposes of our experiments, in which the formation is commanded to move in an almost straight line. In fact, the deviations from the desired geometry, that arise due to the simple controller we have employed, facilitate the demonstration of the robustness of our measurement frequency optimization method to small changes in formation shape[3].

[3]We should note that our objective is to determine optimal measurement frequencies given a formation geometry, and not to design an optimal controller for maintaining such a desired geometry. This second problem has received considerable attention in the literature, and the interested reader is referred to [155]

Table 3.1: Optimal measurement frequencies for the experiment

GPS	Comp.	M_{21}	M_{42}	M_{41}	M_{43}	M_{31}
1.0	1.0	0.216	0.234	0.099	0.234	0.216

By constraining the maximum total frequency of measurements that can be processed by the system to be equal to 3 Hz, the optimal frequencies of all measurements are shown in Table 3.1. These results are obtained by a Matlab implementation of the algorithm, that requires 11 sec of CPU time on a 1.6 GHz Pentium M processor. From the numerical results in Table 3.1, we note that the absolute position and absolute orientation sensors are utilized at their maximum frequency, while the remaining resources are allocated to the relative pose measurements. It is interesting to note that the measurement between the rear robot and the leader is assigned a smaller frequency (although not zero), which should be attributed to the fact that this measurement is less accurate, due to the larger distance from the leader.

In Fig. 3.4 we present the time evolution of the covariance along the x and y axes, for the robots of the team (solid lines). The time evolution of the actual covariance is compared to the theoretically predicted values (dashed lines), computed by solving the SDP (3.37). Although the time duration of the experiment did not allow for the covariance matrix to converge fully to its steady-state value, these figures indicate, that the deviation between the theoretically predicted values, and those computed by the EKF, is very small. This deviation is due to the facts that (i) there is a small discretization error inherent in the transition between the continuous- and discrete-time system models [91], (ii) in the EKF the *estimates* for the pose of the robots are employed to evaluate the measurement Jacobians, and these estimates are generally not precisely equal to the desired poses of the robots, (iii) the laser scanners provide measurements at a frequency which is only approximately constant, and (iv) the formation maintains the desired geometry within some error, determined from the controller's performance.

Variations in the formation geometry during the experiment are shown in Fig. 3.5, where we plot the estimated coordinates of the relative position of R_1 with respect to R_4, as a function of time. As evident, the estimates deviate significantly from their nominal values

for an overview of existing approaches.

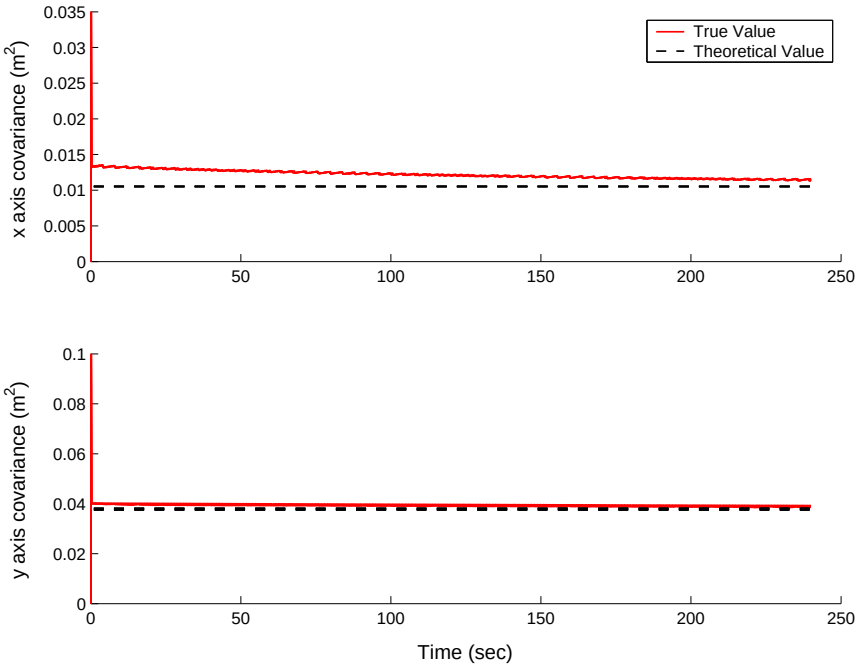


Figure 3.4: Time evolution of the covariance along the two coordinate axes for all the robots, when the optimal measurement frequencies are used. The (red) solid lines represent the actual covariance values computed by the EKF, while the dashed lines represent the theoretically computed steady-state values.

of $(\Delta x_{14}, \Delta y_{14}) = (2, 0)$ m. These deviations are primarily due to the rough terrain that the robots move on, which often resulted in the Pioneers' caster wheels getting stuck. As a consequence of the fluctuations in the relative poses of the robots, the covariance of the relative pose measurements was also time-varying, since the number of laser points used for linefitting was not constant for each robot pair. It is significant to observe that despite these differences from the nominal values, the theoretically predicted covariance is very close to the actual one, which verifies the applicability of our approach to practical scenarios.

In order to demonstrate the positioning accuracy improvement that is achieved using the proposed optimization algorithm, we compare the performance of the optimal strategy with that of an "intuitive" strategy, where the available resources are divided equally among all the available measurements (i.e., when we use all measurements at the same rate, $f_j = 3/7$ Hz). In Fig. 3.6, the time evolution of the covariance in these two scenarios is shown. As evident, there is a clear improvement of performance by using the frequency values produced by the proposed algorithm. Evaluating the *steady-state* covariance attained with the equal-frequency strategy shows that it is approximately 130% and 50% larger along the x and y axes, respectively, compared to the optimal values obtained with our approach. Due to the slow transient response of the covariance, the steady-state value for the case of equal frequencies is not reached in the duration of this experiment. This explains the smaller difference in covariance between the optimal and the "intuitive" approach that appears in Fig. 3.6.

3.7 Simulation results

This section presents simulation results that demonstrate certain additional interesting properties of the problem of determining the optimal sensing frequencies for groups of robots. We here consider a formation with the same geometry as the one shown in Fig. 3.3, but we now examine the case where all robots are equipped with a distance and a bearing sensor, that are capable of providing independent measurements, with standard deviations $\sigma_\rho = 0.05$ m, and $\sigma_\theta = 1^{\text{o}}$, respectively. Additionally, we assume that all robots have a 360^{o} field of view, and can potentially record relative measurements of all other robots. The leader robot receives absolute position measurements with standard deviations equal to

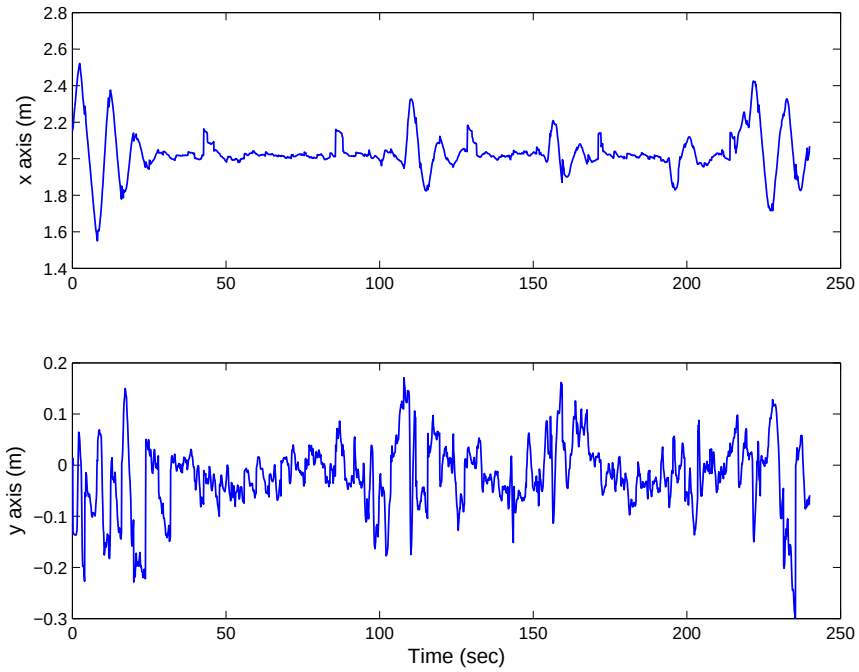


Figure 3.5: Time evolution of the estimates for the relative position of the leader with respect to the rear robot.

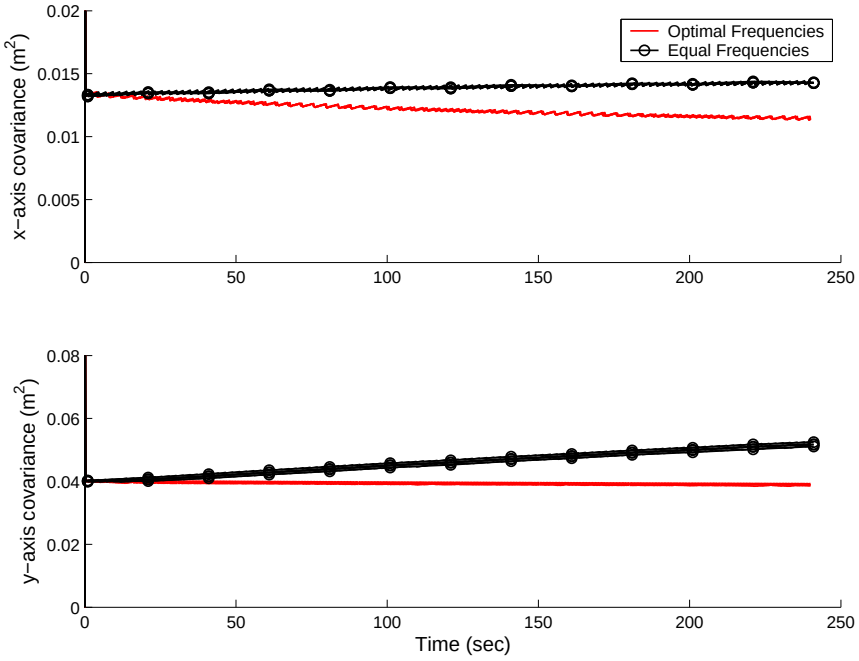


Figure 3.6: Comparison of the covariance values that arise when using the optimal measurement frequencies (solid lines) vs. equal measurement frequencies for all exteroceptive measurements (dashed lines with circles). The two plots correspond to the covariance along the x- and y-axis respectively, for all robots.

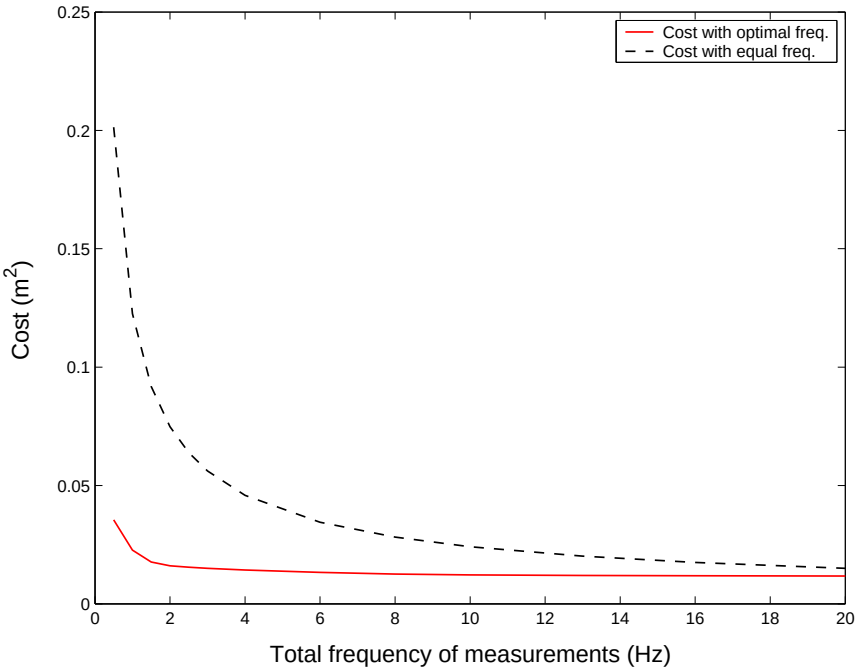


Figure 3.7: Cost function vs. Total frequency of measurements.

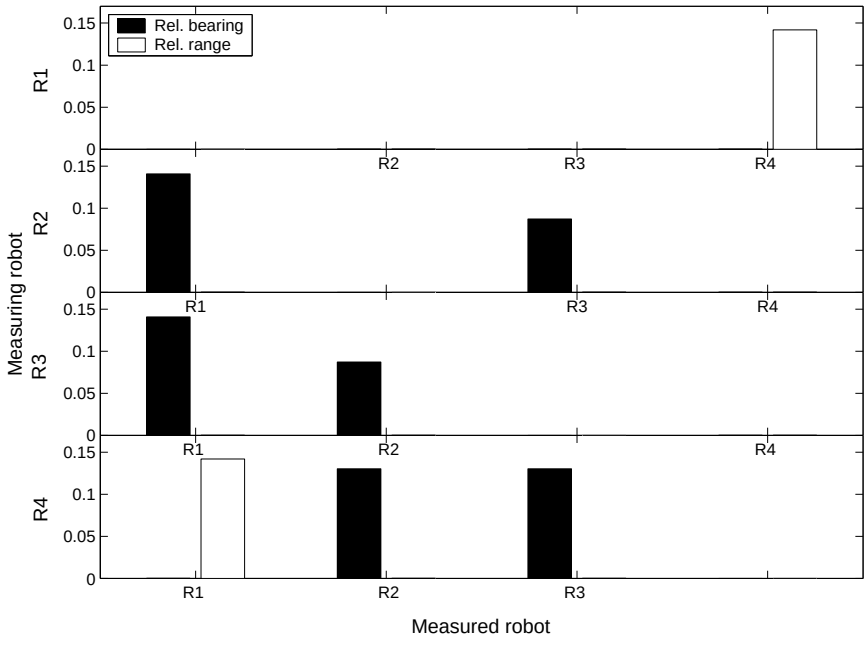


Figure 3.8: Optimal values for the relative range and bearing frequencies.

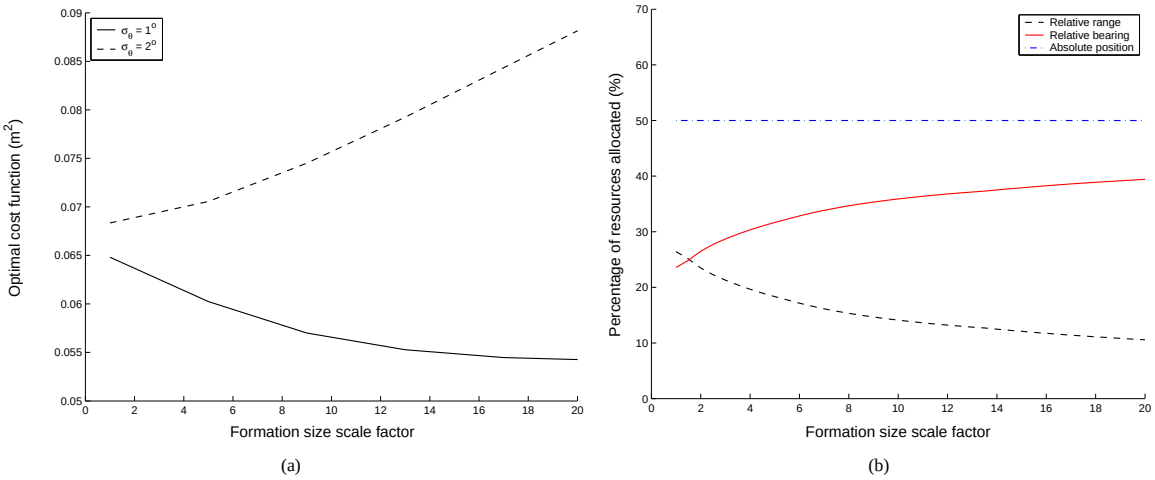


Figure 3.9: (a) The optimal cost as a function of the formation size, for two values of the relative bearing errors' standard deviation. (b) The percentage of resources allocated to each type of measurement, as a function of the formation size, for $\sigma_\theta = 1^\text{o}$.

$\sigma_p = 0.3$ m along each axis, and absolute orientation measurements with standard deviation $\sigma_\phi = 3^\text{o}$. The maximum frequency of all measurements is equal to 1 Hz, and the threshold on the orientation variance for the robots, is equal to $\epsilon_\phi = 0.0027$ rad^2, corresponding to a standard deviation of 3^o.

We first examine the effect of varying the total frequency of measurements processed by the robots. In Fig. 3.7, the optimal value of the cost function is plotted as a function of the total frequency of measurements (solid line) ($f_\text{total} = 0.5 \ldots 20$ Hz), and compared to the cost that arises if equal measurement frequencies are employed (dashed line). In this plot, the substantial improvement in localization accuracy attained using our method becomes apparent. For example, for $f_\text{total} = 1$ Hz, the cost when using equal frequencies is 560% larger than when using the optimal frequencies. Moreover, in this plot we observe a law of diminishing return: there is a sharp improvement in performance by increasing the total number of measurements per time-step, when this number is small, but the incremental gain reduces as the frequency of measurements increases further. Since the necessary communication and computational resources increase linearly with the number of measurements performed by the robots, it becomes clear that unless resources are abundant, it is not beneficial for the robots to process a very large number of measurements.

We now constrain the total frequency of measurements to be equal to $f_\text{total} = 2$ Hz, and

run the optimization algorithm. At the optimal solution, the GPS receiver is utilized at its maximum frequency ($f_{GPS} = 1$ Hz), and interestingly, no absolute orientation measurements need to be recorded. The optimal frequencies for the range and bearing measurements are shown in Fig. 3.8, in the form of a bar plot, where each row of the plot corresponds to the measurements recorded by one robot. The fact that no absolute orientation measurements are used implies that the correlations between the position and orientation estimates of the robots suffice for guaranteing orientation variance smaller than ϵ_ϕ for all robots. However, it should be made clear that this is *not* a general result. For example, if we double the standard deviation of the absolute position measurements, the results of the optimization under the same conditions show that absolute orientation measurements *are* processed by the robots. Nevertheless, the fact that for certain formations some measurement frequencies may turn out to be equal to zero implies that the corresponding sensors are not necessary, and can be omitted, thus resulting in lower cost and easier implementation.

In the last set of experiments, we assume that no absolute orientation sensors are available to the robots, and thus the absolute position measurements of R_1 constitute the only source of absolute state information. We once again select $f_{\text{total}} = 2$ Hz, and vary the formation size, by scaling all distances among robots by a factor ranging between 1 and 20. The solid line in Fig. 3.9(a) presents the optimal cost as a function of the formation size, for $\sigma_\theta = 1^{\text{o}}$. It is worth noting that in this case as the formation scale factor increases, the robots' localization accuracy becomes better. This is attributed to the fact that in the sensor model for relative measurements, the noise variance is independent of the distance between robots. Therefore, the bearing measurements provide better orientation information for the measuring robot, as the robots get further apart, since the errors in the measured robot's position have less impact. This interpretation is also corroborated by Fig. 3.9(b) where we plot the proportion of resources (i.e., proportion of the total measurement frequency) assigned to each type of measurement, as the formation size increases. We observe that as robots become more distant, more relative bearing information is utilized. However this is, once again, not a general result: if we increase the standard deviation of the bearing measurements by a mere factor of 2, to $\sigma_\theta = 2^{\text{o}}$, then as the formation becomes larger, the robots' localization accuracy degrades (this is shown by the dashed line in Fig. 3.9(a)). In

this case, the bearing measurements contribute less localization information, and cannot compensate for the loss of information in the range measurements, due to the increased distances among robots.

As a closing remark, we note that the parameters affecting the selection of optimal measurement frequencies include the number of robots, the size and geometric configuration of the formation in space, the robots' velocity, the accuracy of all available sensors, the type and number of available measurements, and the maximum frequency of each sensor. The results presented in this section illustrate the fact that the interactions between these factors are quite intricate, and determining general "rules of thumb" for the optimal sensing strategy appears difficult, if not infeasible. This further establishes the necessity for a design tool that allows, given all the relevant parameters of a particular robot team, to determine measurement strategies that are provably optimal. In this work, we have presented a method that yields these optimal results, within the described problem formulation.

3.8 Summary

In this chapter we have described a methodology for studying and optimizing the accuracy of localization in robot formations. The key element of our approach is the transition from the discrete-time system model, where the computation of the asymptotic accuracy is analytically intractable, to a *continuous-time* one. The frequency at which each sensor input is processed specifies the accuracy of the corresponding measurement in the continuous-time model. This relation enables us to formulate a *convex optimization problem* for the measurement frequencies, where the constraints on the communication, processing, and power resources of the team are naturally incorporated. Moreover, this problem can be cast as a semidefinite programming (SDP) problem, whose unique global solution can be computed using well-studied and efficient minimization algorithms.

The results of our work can be employed in practice for determining the sensing frequencies for robot formations of any size and shape comprised of robots with various types of sensors and sensing capabilities. The optimal sensing frequencies can be used not only for obtaining the best localization results, but also for determining the necessity of certain sensors (e.g., sensors with zero frequency can be omitted) which can lead to significant cost

savings. As a closing remark, we point out that the applicability of the proposed method is not limited to the problem of formation localization. The idea of employing a transition from the discrete-time to a continuous-time system model is general, and can be applied to any scheduling problem for which the continuous-time system is linear time invariant.

Chapter 4

The Multi-State Constraint Kalman Filter (MSC-KF)

4.1 Introduction

The work presented in the preceding two chapters has focused on vehicles moving in 2D. For robots operating in most man-made environments (e.g., indoors, on paved roads, etc), the assumption of 2D motion is satisfied; however, in an increasing number of robotics applications, the motion of vehicles cannot be described by 2D kinematics. For example, estimation of the full 3D pose (3 degrees of freedom for position and 3 for attitude) is necessary for spacecraft [10,20,69,159], Unmanned Aerial Vehicles (UAVs) [73,74,79,80,125], autonomous underwater vehicles [39, 40], robots operating on rugged outdoor terrain [75, 120, 158], and, in some cases, robots moving indoors [59, 112, 169].

The proprioceptive and exteroceptive sensors that can be employed for pose tracking in 3D differ from those that are suitable for 2D localization. For motion in 3D, proprioceptive information is typically obtained from Inertial Measurement Units (IMUs), which provide measurements of rotational velocity and linear acceleration. Recent advances in MEMS-based IMU technology have made possible the manufacturing of small, low-power, and lightweight IMUs, which attain good measurement accuracy at low cost. This, in turn, enables the use of inertial sensing for position tracking in small-scale systems, such as miniature UAVs and ground robots. However, pose tracking with IMU measurements alone

is susceptible to significant drift, due to the integration of noise and biases. As a result, for accurate pose estimation over longer time periods, additional sensing modalities must be used to aid inertial sensing.

In the case of 3D motion, and in the absence of GPS, cameras are the most attractive option for obtaining exteroceptive information. While 3D laser scanners have been employed in certain cases [163], their high cost, weight, and power consumption render their use impractical, at least with currently existing technology. On the other hand, cameras are inexpensive, small, and have low power requirements, properties that make them easy to incorporate in any robot's sensing payload. Moreover, an important advantage of visual sensing is that images are high-dimensional measurements, with rich information content. For these reasons, the topic of *vision-aided inertial navigation* has recently received considerable attention in the research community [1].

A major difficulty when utilizing visual feature measurements for localization is the high volume of available data. Feature extraction methods can typically detect and track hundreds of features in images [56, 84, 86], and processing all these measurements for real-time pose estimation is a truly challenging problem. If computing resources were unlimited, one would jointly estimate the positions of all features along with the camera trajectory, by employing a SLAM algorithm. This would result in the best possible estimation accuracy. However, in any real application the available computing resources are limited, and one is faced with a fundamental trade-off between the computational complexity of an algorithm and the resulting estimation accuracy.

In this chapter we propose a solution to the problem of processing a large number of features, motivated by the properties of visual features encountered in practice. Specifically, a key observation is that, typically, the vast majority of features can only be tracked for a small number of frames, while only few of them can be detected for long periods of time, or when a robot re-visits an area. We term the former *transient features*, while the latter *stable features*. It becomes clear that including both types of features in the map of a SLAM formulation constitutes suboptimal use of the available resources: once transient features are lost, maintaining them in the SLAM state vector does not offer any additional localization information.

Our proposed solution is to utilize all transient features for improving the robot's *motion* estimates, while stable features are employed for *loop closing*, in order to maintain long-term bounded errors. In this chapter, we describe an algorithm that is capable of processing the transient-feature measurements in a statistically optimal fashion, while having computational complexity only linear in the number of features [109]. This algorithm, termed multi-state constraint Kalman filter (MSC-KF), is described in Section 4.4. Moreover, in Section 4.5 a dual-layer system architecture is described, which utilizes the MSC-KF in conjunction with a module that processes loop-closure information. By exploiting both the real-time performance of the MSC-KF, and the localization information provided by the stable features, the resulting system is capable of producing pose estimates that are available not only at a high rate, but also with bounded long-term errors [110]. Before delving into the details of the methods, in the next section we present the key ideas of our approach.

4.2 Overview of the approach

The design of the MSC-KF estimator is motivated by the observation that, when a static feature is viewed from several camera poses, it is possible to define *geometric constraints* involving all these poses. This is demonstrated in Fig. 4.1, which illustrates the scenario where a feature (denoted by a star) is viewed from a number of consecutive camera poses (denoted by circles). Each measurement of the feature (denoted by an arrow) defines a constraint between the feature and the camera. In a SLAM algorithm, these constraints are utilized by including the feature in the state vector, and estimating its position simultaneously with the camera pose. In contrast, in the MSC-KF the feature is *not* included in the state vector. Instead, the feature measurements are processed so as to remove the dependence on the feature position, which leaves a number of constraints involving the camera poses *only*, as shown in the bottom part of Fig. 4.1. The key contribution of the MSC-KF is a measurement model that optimally utilizes these constraints within the EKF framework.

We note that the MSC-KF only processes local motion information, in the form of IMU measurements and features tracked in consecutive images. Thus, it is essentially a combined *visual/inertial odometry* estimator. As a result, the uncertainty of the pose estimates in the MSC-KF will gradually increase over time. To compensate for the error growth, we

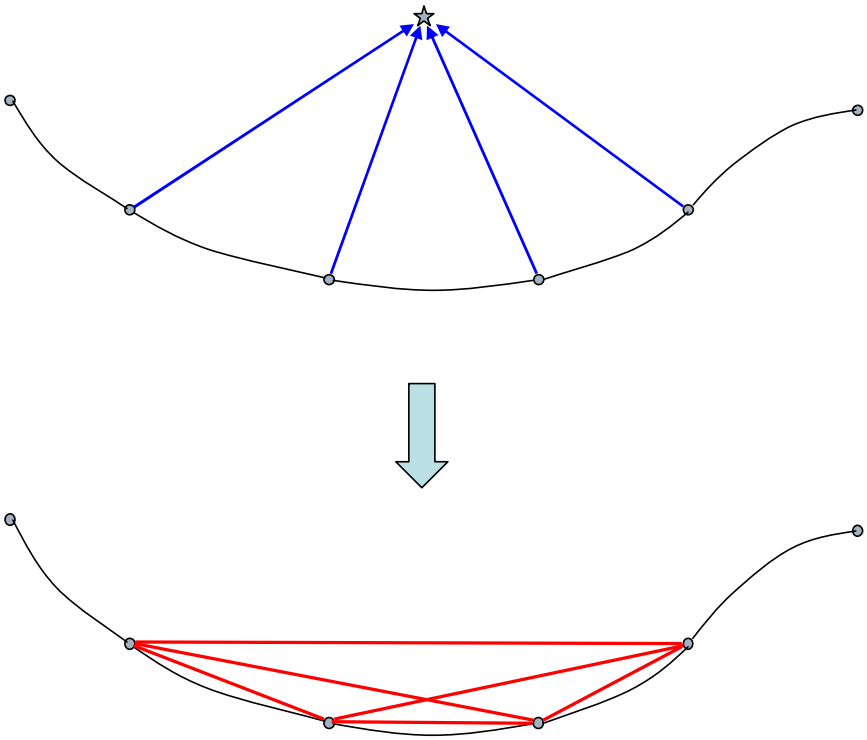


Figure 4.1: Demonstration of the key idea behind the MSC-KF. The constraints of the feature measurements (top) are transformed into constraints involving the camera poses only (bottom).

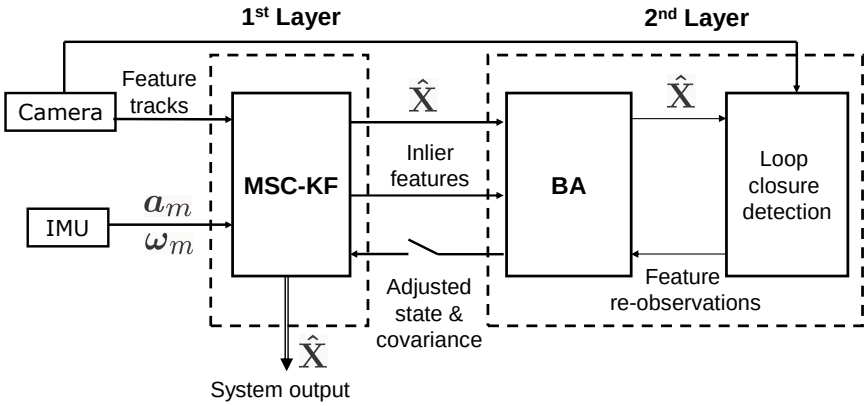


Figure 4.2: The block diagram of the dual-layer localization architecture.

design a system that, in addition to the local motion information, also utilizes loop-closure constraints (i.e., the re-observations of stable features that occur when the robot re-visits an area). The key characteristic of the system is its dual-layer estimation architecture (cf. Fig. 4.2): At the first layer, the MSC-KF fuses the visual and inertial measurements, and reports the pose estimates at the IMU data rate. At the second layer of the architecture we employ a least-squares Bundle-Adjustment (BA) estimator [162] in conjunction with a loop-closure detection module. Every time a loop-closing event is detected, the re-observations of previously seen features are processed by the BA algorithm, to obtain an improved pose estimate. This estimate is then fed back to the MSC-KF, thus resulting in diminished localization errors.

4.3 Prior work

Before presenting the details of the MSC-KF algorithm, in this section we examine alternative candidate approaches for using the transient feature information. We point out that the MSC-KF algorithm is general, and can be applied with any type of exteroceptive feature measurements (e.g., features extracted in laser-scan data). However, the problem of processing a large number of transient features is primarily important for vision-based localization, and for this reason our review of relevant methods will focus on this domain. The existing approaches can broadly be categorized in three classes:

SLAM methods

One family of algorithms for fusing proprioceptive information with visual feature observations follows the SLAM paradigm. In these methods, the current robot pose, as well as the 3D positions of all visual landmarks are jointly estimated, and estimation is typically carried out by an EKF (e.g., [21, 28, 80, 99, 125, 130, 153]). While in these approaches all the measurements of the transient features can be processed optimally (up to linearization), the cost of doing so is *quadratic* in the number of features. For this reason, EKF-based SLAM approaches for vision-based localization focus on a *small subset* of features that can be tracked over long time periods, and all other measurements are discarded. Clearly, this leads to loss of the information provided by the transient features.

Use of pair-wise constraints

Several algorithms exist that, contrary to SLAM, estimate the pose of the camera *only* (i.e., do not jointly estimate the feature positions), with the aim of achieving real-time operation. The most computationally efficient of these methods utilize the feature measurements to derive constraints between pairs of images. For example in [138, 139], an image-based motion estimation algorithm is applied to consecutive pairs of images, to obtain displacement estimates that are subsequently fused with inertial measurements. Similarly, in [10, 31, 32] constraints between current and previous image are defined using the epipolar geometry, and combined with IMU measurements in an EKF. In [150, 151] the epipolar geometry is employed in conjunction with a statistical motion model, while in [131] epipolar constraints are fused with the dynamical model of an airplane.

These algorithms fuse image-based displacement measurements with motion information. In addition, a number of *visual odometry* approaches exist, which propagate the robot pose using displacement measurements *only*. These methods typically employ a stereo-pair of cameras for displacement estimation [90, 122–124], since the absolute magnitude (scale) of motion cannot be determined when only a single camera is available. The use of feature measurements for imposing constraints between *pairs* of images is similar in philosophy to the MSC-KF. However, one fundamental difference is that our algorithm can express constraints between *multiple* camera poses, and can thus attain higher estimation accuracy, in

the common case where features are visible in more than two images.

Pairwise constraints have also been employed in algorithms that maintain a state vector comprised of multiple camera poses. In [47], an augmented-state Kalman filter is implemented, in which a sliding window of robot poses is maintained in the filter state. On the other hand, in [38–40], *all* camera poses are simultaneously estimated. In both of these algorithms, *pairwise* relative-pose measurements are derived from the images, and used for state updates. The drawback of this approach is that, when a feature is seen in multiple images, the additional constraints between the multiple poses are discarded, thus resulting in loss of information. Furthermore, when the same image measurements are processed for computing several displacement estimates, these are not statistically independent, as shown in [111].

Use of multi-state constraints

A number of approaches exist that, similarly to the MSC-KF, utilize the feature measurements to impose constraints on a sliding window of poses. For instance, the algorithms presented in [75, 120] temporarily initialize features, use them for imposing constraints on windows of consecutive camera poses, and then discard them. These methods, however, employ only a *loose* coupling between the visual and inertial measurements: the IMU rotational velocity measurements are used to independently compute attitude estimates, which are subsequently fused with the results of the visual odometry module. In this case, the resulting estimates are suboptimal, because the IMU biases are not updated. A loose coupling of visual and inertial measurements is also employed in the system presented in [126], which uses multiple cameras for visual odometry, and then fuses the result with that of pure IMU-based pose tracking. In contrast, in the MSC-KF the visual and proprioceptive measurements are fused in a *tightly coupled* formulation, which results in increased accuracy.

A window of camera poses is also maintained in the Variable State Dimension Filter (VSDF) [93]. The VSDF is a hybrid batch/recursive method, that carries out bundle adjustment over a sliding window of camera poses. The VSDF (i) uses *delayed linearization* to increase robustness against linearization inaccuracies, and (ii) exploits the sparsity of the information matrix, that naturally arises when no dynamic motion model is used. How-

Algorithm 1 Multi-State Constraint Filter (MSC-KF)

Propagation: For each IMU measurement received, propagate the filter state and covariance (cf. Section 4.4.2).

Image registration: Every time a new image is recorded,

- augment the state and covariance matrix with a copy of the current camera pose estimate (cf. Section 4.4.3).
- image processing module begins operation.

Update: When the feature measurements of a given image become available, perform an EKF update (cf. Sections 4.4.4 and 4.4.6).

ever, in cases where a dynamic motion model is available (such as in vision-aided inertial navigation) the computational complexity of the VSDF is at best *quadratic* in the number of features [29].

In contrast to the VSDF, the MSC-KF is able to exploit the benefits of delayed linearization, while having complexity only linear in the number of features. By directly expressing the constraints between multiple camera poses it avoids the computational burden and loss of information associated with pairwise displacement estimation. Moreover, in contrast to SLAM-type approaches, it does not require the inclusion of the 3D feature positions in the filter state vector, but still attains optimal pose estimation (up to linearization). As a result of these properties, the algorithm is very efficient, and as shown in Section 4.6, is capable of high-precision vision-aided inertial navigation in real time.

4.4 The MSC-KF estimator

We now describe the MSC-KF estimator for vision-aided inertial navigation. The goal of the estimator is to track the 3D pose of the IMU-affixed frame $\{I\}$ with respect to a *global frame* of reference $\{G\}$. In order to simplify the treatment of the effects of the earth's rotation on the IMU measurements, the global frame is chosen as an Earth-Centered, Earth-Fixed (ECEF) one. An overview of the algorithm is given in Algorithm 1. The IMU measurements are processed immediately as they become available, for propagating the EKF state and covariance. On the other hand, each time an image is recorded, the current camera pose

estimate is appended to the state vector, and the image processing algorithm commences feature extraction. State augmentation allows us to create a state vector comprising a sliding window of the N latest camera poses. During EKF updates, the measurements of each tracked feature are used for imposing constraints between these poses. Therefore, at any time instant the EKF state vector comprises (i) the evolving IMU state, $\mathbf{X}_{\text{IMU}}$, and (ii) a history of N past poses of the camera. In the following, we describe the various components of the algorithm in detail.

4.4.1 Structure of the EKF state vector

The evolving IMU state is described by the vector:

$$\mathbf{X}_{\text{IMU}} = \begin{bmatrix} {}^{I}_{G}\bar{q}^T & \mathbf{b}_g{}^T & {}^{G}\mathbf{v}_I{}^T & \mathbf{b}_a{}^T & {}^{G}\mathbf{p}_I^T \end{bmatrix}^T \tag{4.1}$$

where ${}^{I}_{G}\bar{q}$ is the unit quaternion [13] describing the rotation from frame $\{G\}$ to frame $\{I\}$, ${}^{G}\mathbf{p}_I$ and ${}^{G}\mathbf{v}_I$ are the IMU position and velocity with respect to $\{G\}$, and finally $\mathbf{b}_g$ and $\mathbf{b}_a$ are 3×1 vectors that describe the biases affecting the gyroscope and accelerometer measurements, respectively. The IMU biases are modeled as random walk processes, driven by the white Gaussian noise vectors $\mathbf{n}_{wg}$ and $\mathbf{n}_{wa}$, respectively [3]. Following (4.1), the IMU error-state is defined as:

$$\widetilde{\mathbf{X}}_{\text{IMU}} = \begin{bmatrix} \boldsymbol{\delta\theta}_I^T & \widetilde{\mathbf{b}}_g^T & {}^{G}\widetilde{\mathbf{v}}_I^T & \widetilde{\mathbf{b}}_a^T & {}^{G}\widetilde{\mathbf{p}}_I^T \end{bmatrix}^T \tag{4.2}$$

For the position, velocity, and biases, the standard additive error definition is used (i.e., the error in the estimate $\hat{x}$ of a quantity x is defined as $\widetilde{x} = x - \hat{x}$). However, for the quaternion a different error definition is employed. In particular, if $\hat{\bar{q}}$ is the estimated value of the quaternion $\bar{q}$, then the orientation error is described by the *error quaternion* $\delta\bar{q}$, which is defined by the relation:

$$\bar{q} = \delta\bar{q} \odot \hat{\bar{q}}, \quad \text{where} \quad \delta\bar{q} = \begin{bmatrix} \frac{1}{2}\boldsymbol{\delta\theta}^T & \sqrt{1 - \frac{1}{4}\boldsymbol{\delta\theta}^T\boldsymbol{\delta\theta}} \end{bmatrix}^T \simeq \begin{bmatrix} \frac{1}{2}\boldsymbol{\delta\theta}^T & 1 \end{bmatrix}^T \tag{4.3}$$

In this expression, the symbol $\odot$ denotes quaternion multiplication. Intuitively, the quaternion $\delta\bar{q}$ describes the (small) rotation that causes the true and estimated attitude to coincide.

Since attitude corresponds to 3 degrees of freedom, using $\boldsymbol{\delta\theta}$ to describe the attitude errors is a minimal representation.

Assuming that N camera poses are included in the EKF state vector at time-step k, this vector has the following form:

$$\hat{\mathbf{X}}_k = \begin{bmatrix} \hat{\mathbf{X}}_{\text{IMU}_k}^T & {}^{C_1}_{G}\hat{q}^T & {}^G\hat{\mathbf{p}}_{C_1}^T & \dots & {}^{C_N}_{G}\hat{q}^T & {}^G\hat{\mathbf{p}}_{C_N}^T \end{bmatrix}^T \tag{4.4}$$

where ${}^{C_i}_{G}\hat{q}$ and ${}^G\hat{\mathbf{p}}_{C_i}$, $i = 1 \dots N$ are the estimates of the camera attitude and position, respectively, for the N poses maintained in the sliding window. The EKF error-state vector is defined accordingly:

$$\widetilde{\mathbf{X}}_k = \begin{bmatrix} \widetilde{\mathbf{X}}_{\text{IMU}_k}^T & \boldsymbol{\delta\theta}_{C_1}^T & {}^G\widetilde{\mathbf{p}}_{C_1}^T & \dots & \boldsymbol{\delta\theta}_{C_N}^T & {}^G\widetilde{\mathbf{p}}_{C_N}^T \end{bmatrix}^T \tag{4.5}$$

4.4.2 Propagation

During each propagation step the estimates for the N poses of the sliding window remain unchanged, while the IMU state estimate is propagated using the IMU measurements. Specifically, at every time-step the IMU provides measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$ of the rotational velocity and the linear acceleration signals. These measurements are processed for propagating the IMU pose estimate:

$$\hat{\mathbf{X}}_{\text{IMU}_{k+1|k}} = \psi(\hat{\mathbf{X}}_{\text{IMU}_{k|k}}, \boldsymbol{\omega}_m, \mathbf{a}_m) \tag{4.6}$$

The state propagation function, ψ, is implemented using Runge-Kutta numerical integration of the continuous-time IMU system model, as described in Appendix C.1. Moreover, the EKF covariance matrix has to be propagated. For this purpose, we introduce the following partitioning for the covariance matrix:

$$\mathbf{P}_{k|k} = \begin{bmatrix} \mathbf{P}_{II_{k|k}} & \mathbf{P}_{IC_{k|k}} \\ \mathbf{P}_{IC_{k|k}}^T & \mathbf{P}_{CC_{k|k}} \end{bmatrix} \tag{4.7}$$

where $\mathbf{P}_{II_{k|k}}$ is the 15×15 covariance matrix of the evolving IMU state, $\mathbf{P}_{CC_{k|k}}$ is the $6N \times 6N$ covariance matrix of the camera pose estimates, and $\mathbf{P}_{IC_{k|k}}$ is the correlation

between the errors in the IMU state and the camera pose estimates. With this notation, the covariance matrix of the propagated state is given by:

$$\mathbf{P}_{k+1|k} = \begin{bmatrix} \mathbf{P}_{II_{k+1|k}} & \mathbf{\Phi}(k+1,k)\mathbf{P}_{IC_{k|k}} \\ \mathbf{P}_{IC_{k|k}}^T \mathbf{\Phi}(k+1,k)^T & \mathbf{P}_{CC_{k|k}} \end{bmatrix} \tag{4.8}$$

where the terms $\mathbf{P}_{II_{k+1|k}}$ and $\mathbf{\Phi}(k+1,k)$ are computed by Runge-Kutta numerical integration, as explained in Appendix C.1.

4.4.3 State augmentation

Upon recording a new image, the camera pose estimate is computed from the IMU pose estimate as:

$$\hat{{}^C_G\bar{q}} = {}^C_I\bar{q} \odot \hat{{}^I_G\bar{q}}, \quad \text{and} \quad \hat{{}^G\mathbf{p}}_C = \hat{{}^G\mathbf{p}}_I + \mathbf{C}(\hat{{}^I_G\bar{q}})^T\, {}^I\mathbf{p}_C \tag{4.9}$$

where ${}^C_I\bar{q}$ is the quaternion expressing the rotation between the IMU and camera frames, and ${}^I\mathbf{p}_C$ is the position of the origin of the camera frame with respect to $\{I\}$, both of which are known [96]. This camera pose estimate is appended to the state vector, and the covariance matrix of the EKF is augmented accordingly:

$$\mathbf{P}_{k|k} \leftarrow \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix} \mathbf{P}_{k|k} \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix}^T \tag{4.10}$$

where the Jacobian $\mathbf{J}$ is derived from (4.9) as:

$$\mathbf{J} = \begin{bmatrix} \mathbf{C}\left({}^C_I\bar{q}\right) & \mathbf{0}_{3\times 9} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 6N} \\ \lfloor \mathbf{C}(\hat{{}^I_G\bar{q}})^T {}^I\mathbf{p}_C \times \rfloor & \mathbf{0}_{3\times 9} & \mathbf{I}_3 & \mathbf{0}_{3\times 6N} \end{bmatrix} \tag{4.11}$$

4.4.4 Measurement model

We now present the measurement model employed for updating the state estimates, which is the key element of the MSC-KF algorithm. Since the EKF is used for state estimation, for constructing a measurement model it suffices to define a residual, $\mathbf{r}$, that depends linearly

on the state errors, $\widetilde{\mathbf{X}}_{k+1|k}$, according to the general form:

$$\mathbf{r} = \mathbf{H}\widetilde{\mathbf{X}}_{k+1|k} + \text{noise} \tag{4.12}$$

In this expression $\mathbf{H}$ is the measurement Jacobian matrix, and the noise term must be zero-mean, white, and *uncorrelated* to the state error, for the EKF framework to be applied.

We present the measurement model by considering the case of a *single* feature, f_j, that has been observed from a set of M_j camera poses $({}^{C_i}_{G}\bar{q}, {}^{G}\mathbf{p}_{C_i})$, $i \in \mathcal{S}_j$. Each of the M_j observations of the feature is described by the model:

$$\mathbf{z}_i^{(j)} = h_{\text{proj}}({}^{G}\mathbf{p}_{f_j}, \Pi_i) + \mathbf{n}_i^{(j)} = \frac{1}{{}^{C_i}z_j}\begin{bmatrix} {}^{C_i}x_j \\ {}^{C_i}y_j \end{bmatrix} + \mathbf{n}_i^{(j)}, \qquad i \in \mathcal{S}_j \tag{4.13}$$

where $\Pi_i = \{{}^{C_i}_{G}\bar{q}, {}^{G}\mathbf{p}_{C_i}\}$ denotes the i-th camera pose, and $\mathbf{n}_i^{(j)}$ is the 2×1 image noise vector, with covariance matrix $\mathbf{R}_i^{(j)} = \sigma_{\text{im}}^2\mathbf{I}_2$. The feature position expressed in the camera frame, ${}^{C_i}\mathbf{p}_{f_j}$, is given by:

$${}^{C_i}\mathbf{p}_{f_j} = \begin{bmatrix} {}^{C_i}x_j \\ {}^{C_i}y_j \\ {}^{C_i}z_j \end{bmatrix} = \mathbf{C}({}^{C_i}_{G}\bar{q})({}^{G}\mathbf{p}_{f_j} - {}^{G}\mathbf{p}_{C_i}) \tag{4.14}$$

where ${}^{G}\mathbf{p}_{f_j}$ is the 3D feature position in the global frame. Since this is unknown, in the first step of our algorithm we employ least-squares minimization to obtain an estimate, ${}^{G}\hat{\mathbf{p}}_{f_j}$, of the feature position. This is achieved using the measurements $\mathbf{z}_i^{(j)}$, $i \in \mathcal{S}_j$, and the filter estimates of the camera poses at the corresponding time instants (cf. Appendix C.2).

Once the estimate of the feature position is obtained, we compute the measurement residual:

$$\mathbf{r}_i^{(j)} = \mathbf{z}_i^{(j)} - \hat{\mathbf{z}}_i^{(j)} \tag{4.15}$$

where

$$\hat{\mathbf{z}}_i^{(j)} = \frac{1}{{}^{C_i}\hat{z}_j} \begin{bmatrix} {}^{C_i}\hat{x}_j \\ {}^{C_i}\hat{y}_j \end{bmatrix} \ , \ \begin{bmatrix} {}^{C_i}\hat{x}_j \\ {}^{C_i}\hat{y}_j \\ {}^{C_i}\hat{z}_j \end{bmatrix} = \mathbf{C}({}^{C_i}_{G}\hat{\bar{q}})({}^{G}\hat{\mathbf{p}}_{f_j} - {}^{G}\hat{\mathbf{p}}_{C_i})$$

Linearizing about the estimates for the camera pose and for the feature position, the residual of (4.15) can be approximated as:

$$\mathbf{r}_i^{(j)} \simeq \mathbf{H}_{\mathbf{X}_i}^{(j)} \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{H}_{f_i}^{(j)} {}^{G}\widetilde{\mathbf{p}}_{f_j} + \mathbf{n}_i^{(j)} \tag{4.16}$$

In the preceding expression $\mathbf{H}_{\mathbf{X}_i}^{(j)}$ and $\mathbf{H}_{f_i}^{(j)}$ are the Jacobians of the measurement $\mathbf{z}_i^{(j)}$ with respect to the state and the feature position, respectively, and ${}^{G}\widetilde{\mathbf{p}}_{f_j}$ is the error in the position estimate of f_j. The Jacobians are given by:

$$\mathbf{H}_{\mathbf{X}_i}^{(j)} = \begin{bmatrix} \mathbf{0}_{2\times 15} & \mathbf{0}_{2\times 6} & \cdots & \underbrace{\mathbf{J}_i^{(j)} \lfloor {}^{C_i}\hat{\mathbf{X}}_{f_j} \times \rfloor \quad -\mathbf{J}_i^{(j)} \mathbf{C}({}^{C_i}_{G}\hat{\bar{q}})}_{\text{Jacobian wrt pose i}} & \cdots \end{bmatrix} \tag{4.17}$$

and

$$\mathbf{H}_{f_i}^{(j)} = \mathbf{J}_i^{(j)} \mathbf{C}({}^{C_i}_{G}\hat{\bar{q}}) \tag{4.18}$$

where

$$\mathbf{J}_i^{(j)} = \nabla_{{}^{C_i}\hat{\mathbf{p}}_{f_j}} \mathbf{z}_i^{(j)} = \frac{1}{{}^{C_i}\hat{z}_j} \begin{bmatrix} 1 & 0 & -\frac{{}^{C_i}\hat{x}_j}{{}^{C_i}\hat{z}_j} \\ 0 & 1 & -\frac{{}^{C_i}\hat{y}_j}{{}^{C_i}\hat{z}_j} \end{bmatrix}$$

By stacking the residuals of all M_j measurements of this feature, we obtain:

$$\mathbf{r}^{(j)} \simeq \mathbf{H}_{\mathbf{X}}^{(j)} \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{H}_{f}^{(j)} {}^{G}\widetilde{\mathbf{p}}_{f_j} + \mathbf{n}^{(j)} \tag{4.19}$$

where $\mathbf{r}^{(j)}$, $\mathbf{H}_{\mathbf{X}}^{(j)}$, $\mathbf{H}_{f}^{(j)}$, and $\mathbf{n}^{(j)}$ are block vectors or matrices with elements $\mathbf{r}_i^{(j)}$, $\mathbf{H}_{\mathbf{X}_i}^{(j)}$, $\mathbf{H}_{f_i}^{(j)}$, and $\mathbf{n}_i^{(j)}$, for $i \in \mathcal{S}_j$. Since the feature observations in different images are conditionally independent, the covariance matrix of $\mathbf{n}^{(j)}$ is $\mathbf{R}^{(j)} = \sigma_{\text{im}}^2 \mathbf{I}_{2M_j}$.

Note that since the state estimate $\hat{\mathbf{X}}_{k+1|k}$ is used to compute the feature position estimate (cf. Appendix C.2), the error ${}^{G}\widetilde{\mathbf{p}}_{f_j}$ in (4.19) is correlated with the errors $\widetilde{\mathbf{X}}_{k+1|k}$.

Thus, the residual $\mathbf{r}^{(j)}$ is not in the form of (4.12), and cannot be directly applied for measurement updates in the EKF. To overcome this problem, we define a residual $\mathbf{r}_o^{(j)}$, by projecting $\mathbf{r}^{(j)}$ on the left nullspace of the matrix $\mathbf{H}_f^{(j)}$. Specifically, if we let $\mathbf{U}$ denote the unitary matrix whose columns form the basis of the left nullspace of $\mathbf{H}_f^{(j)}$, we obtain:

$$\mathbf{r}_o^{(j)} = \mathbf{U}^T(\mathbf{z}^{(j)} - \hat{\mathbf{z}}^{(j)}) \simeq \mathbf{U}^T\mathbf{H}_{\mathbf{X}}^{(j)}\widetilde{\mathbf{X}}_{k+1|k} + \mathbf{U}^T\mathbf{n}^{(j)} \tag{4.20}$$

$$= \mathbf{H}_o^{(j)}\widetilde{\mathbf{X}}_{k+1|k} + \mathbf{n}_o^{(j)} \tag{4.21}$$

Since the $2M_j \times 3$ matrix $\mathbf{H}_f^{(j)}$ has full column rank, its left nullspace is of dimension $2M_j - 3$. Therefore, $\mathbf{r}_o^{(j)}$ is a $(2M_j - 3) \times 1$ vector. This residual is *independent* of the errors in the feature coordinates, and thus EKF updates can be performed based on it. Equation (4.21) defines a *linearized* constraint between all the camera poses from which the feature f_j was observed. This expresses all the available information that the measurements $\mathbf{z}_i^{(j)}$ provide for the M_j states, and thus the resulting EKF update is optimal, except for the inaccuracies caused by linearization. A proof of this statement is presented in Appendix C.3.

It should be mentioned that in order to compute the residual $\mathbf{r}_o^{(j)}$ and the measurement matrix $\mathbf{H}_o^{(j)}$, the unitary matrix $\mathbf{U}$ does not need to be explicitly evaluated. Instead, the projection of the vector $\mathbf{r}^{(j)}$ and of the matrix $\mathbf{H}_{\mathbf{X}}^{(j)}$ on the nullspace of $\mathbf{H}_f^{(j)}$ can be computed very efficiently using Givens rotations [51], in $O(M_j^2)$ operations. Additionally, since the matrix $\mathbf{U}$ is unitary, the covariance matrix of the noise vector $\mathbf{n}_o^{(j)}$ is given by:

$$E\{\mathbf{n}_o^{(j)}\mathbf{n}_o^{(j)T}\} = \sigma_{\text{im}}^2\mathbf{U}^T\mathbf{U} = \sigma_{\text{im}}^2\mathbf{I}_{2M_j-3}$$

It is important to note that the residual defined in (4.20) is not the only possible expression of the constraints that are induced by observing a static feature in multiple images. An alternative approach would be, for example, to employ the epipolar constraints that are defined between $2M_j - 3$ pairs of the images, or to use the multi-linear constraints defined by the M_j measurements directly [57]. However, the resulting constraints are highly nonlinear, and moreover, they are not statistically independent, since each measurement is used in defining multiple constraints. Our experiments have shown that employing linearization of these constraints yields inferior results compared to the approach described above.

4.4.5 Outlier rejection

Prior to using each feature's measurements for updates, an outlier rejection test is performed. Specifically, for each feature the Mahalanobis distance:

$$\gamma = \mathbf{r}_o^{(j)T} \left(\mathbf{H}_o^{(j)} \mathbf{P}_{k+1|k} \mathbf{H}_o^{(j)T} + \sigma_{\text{im}}^2 \mathbf{I}_{2M_j-3} \right)^{-1} \mathbf{r}_o^{(j)} \tag{4.22}$$

is computed, and compared against the 95-th percentile of the χ^2 cumulative distribution function with $2M_j - 3$ degrees of freedom. If γ is smaller than this threshold, the feature is accepted as an inlier, and used in the updates.

Note that, in contrast to outlier rejection based on vision alone, in this outlier rejection scheme the MSC-KF state estimate is used as a prior, to help identify outliers. Additionally, it is important to observe that *all* the measurements of the feature are simultaneously used for the rejection test. As a result, features that correspond to slowly-moving objects, or whose tracking is unreliable, can be more easily detected and discarded. These properties, which arise from the tight coupling of the visual and inertial measurements in the MSC-KF, result in very robust outlier detection.

4.4.6 EKF updates

In Section 4.4.4 we presented the measurement model that expresses the constraints imposed by observing *one* static feature from multiple camera poses. We now present in detail the update phase of the EKF, in which the constraints from observing $L > 1$, features, are available. A feature is used for EKF updates when it can no longer be detected by the feature tracker. Following the procedure described in the preceding section, we compute a residual vector $\mathbf{r}_o^{(j)}$, $j = 1 \dots L$, as well as a corresponding measurement matrix $\mathbf{H}_o^{(j)}$, $j = 1 \dots L$ for each of these features (cf. (4.20)). By stacking all residuals in a single vector, we obtain:

$$\mathbf{r}_o = \mathbf{H}_{\mathbf{X}} \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{n}_o \tag{4.23}$$

where $\mathbf{r}_o$ and $\mathbf{n}_o$ are vectors with block elements $\mathbf{r}_o^{(j)}$ and $\mathbf{n}_o^{(j)}$, $j = 1 \dots L$, respectively, and $\mathbf{H}_{\mathbf{X}}$ is a matrix with block rows $\mathbf{H}_o^{(j)}$, $j = 1 \dots L$.

Since the feature measurements are statistically independent, the noise vectors $\mathbf{n}_o^{(j)}$ are uncorrelated. Therefore, the covariance matrix of the noise vector $\mathbf{n}_o$ is equal to $\mathbf{R}_o = \sigma_{\text{im}}^2 \mathbf{I}_d$, where $d = \sum_{j=1}^{L}(2M_j - 3)$ is the dimension of the residual $\mathbf{r}_o$. One issue that arises in practice is that d can be a quite large number. For example, if 10 features are seen in 10 camera poses each, the dimension of the residual is 170. In order to reduce the computational complexity of the EKF update, we employ the QR decomposition of the matrix $\mathbf{H_X}$ [10]. Specifically, we denote this decomposition as

$$\mathbf{H_X} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{T}_H \\ \mathbf{0}_{(d-r)\times r} \end{bmatrix}$$

where $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are unitary matrices of dimensions $d \times r$ and $d \times (d-r)$, respectively, whose columns form bases for the range and nullspace of $\mathbf{H_X}$. Moreover, $\mathbf{T}_H$ is a $r \times r$ upper triangular matrix. In these expressions, r is equal to the rank of $\mathbf{H_X}$, which, in turn, expresses the number of independent constraints provided by the feature measurements. With this definition, (4.23) yields:

$$\mathbf{r}_o = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{T}_H \\ \mathbf{0} \end{bmatrix} \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{n}_o \Rightarrow \tag{4.24}$$

$$\begin{bmatrix} \mathbf{Q}_1^T \mathbf{r}_o \\ \mathbf{Q}_2^T \mathbf{r}_o \end{bmatrix} = \begin{bmatrix} \mathbf{T}_H \\ \mathbf{0}_{(d-r)\times r} \end{bmatrix} \widetilde{\mathbf{X}}_{k+1|k} + \begin{bmatrix} \mathbf{Q}_1^T \mathbf{n}_o \\ \mathbf{Q}_2^T \mathbf{n}_o \end{bmatrix} \tag{4.25}$$

From the last equation it becomes clear that by projecting the residual $\mathbf{r}_o$ on the basis vectors of the range of $\mathbf{H_X}$, we retain all the useful information in the measurements. The residual $\mathbf{Q}_2^T \mathbf{r}_o$ is only noise, and can be completely discarded. For this reason, instead of the residual shown in (4.23), we employ the following residual for the EKF update:

$$\mathbf{r}_n = \mathbf{Q}_1^T \mathbf{r}_o = \mathbf{T}_H \widetilde{\mathbf{X}}_{k+1|k} + \mathbf{n}_n \tag{4.26}$$

In this expression $\mathbf{n}_n = \mathbf{Q}_1^T \mathbf{n}_o$ is an $r \times 1$ noise vector, whose covariance matrix is equal to

$\mathbf{R}_n = \mathbf{Q}_1^T \mathbf{R}_o \mathbf{Q}_1 = \sigma_{\text{im}}^2 \mathbf{I}_r$. The EKF update proceeds by computing the Kalman gain:

$$\mathbf{K} = \mathbf{P}_{k+1|k} \mathbf{T}_H^T \left(\mathbf{T}_H \mathbf{P}_{k+1|k} \mathbf{T}_H^T + \mathbf{R}_n \right)^{-1} \tag{4.27}$$

while the correction to the state is given by the vector

$$\Delta \mathbf{X} = \mathbf{K} \mathbf{r}_n \tag{4.28}$$

Due to the representation of attitude errors, the correction vector is not used for additive updates to all the state variables. For example, the vector $\Delta \mathbf{X}_{1:3}$, which corresponds to the IMU attitude, is used to form the quaternion correction:

$$\delta \bar{q} = \begin{bmatrix} \frac{1}{2} \Delta \mathbf{X}_{1:3} \\ \sqrt{1 - \frac{1}{4} \Delta \mathbf{X}_{1:3}^T \Delta \mathbf{X}_{1:3}} \end{bmatrix} \tag{4.29}$$

This is subsequently employed for updating the IMU quaternion estimate, according to:

$${}^I_G \hat{\bar{q}}_{k+1|k+1} = \delta \bar{q} \odot {}^I_G \hat{\bar{q}}_{k+1|k} \tag{4.30}$$

The quaternion estimates for the camera poses that are included in the state vector are updated similarly. On the other hand, the variables that correspond to position, velocity, and IMU biases, are updated using the standard additive updates. Finally, the state covariance matrix is updated according to:

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I}_\mu - \mathbf{K} \mathbf{T}_H \right) \mathbf{P}_{k+1|k} \left(\mathbf{I}_\mu - \mathbf{K} \mathbf{T}_H \right)^T + \mathbf{K} \mathbf{R}_n \mathbf{K}^T \tag{4.31}$$

where $\mu = 6N + 15$ is the dimension of the covariance matrix.

It is interesting to examine the computational complexity of the operations needed during the EKF update. The residual $\mathbf{r}_n$, as well as the matrix $\mathbf{T}_H$, can be computed using Givens rotations in $O(r^2 d)$ operations, without the need to explicitly form $\mathbf{Q}_1$. On the other hand, (4.31) involves multiplication of square matrices of dimension μ, an $O(\mu^3)$ operation. Therefore, the cost of the EKF update is $\max(O(r^2 d), O(\mu^3))$. If, on the other hand, the residual vector $\mathbf{r}_o$ was employed, without projecting it on the range of $\mathbf{H}_\mathbf{X}$, the

computational cost of computing the Kalman gain would have been $O(d^3)$. Since typically $d \gg \mu, r$, we see that the use of the residual $\mathbf{r}_n$ results in substantial savings in computation.

4.4.7 Discussion

We now examine some of the properties of the described algorithm. As shown in the previous section, the filter's computational complexity is *linear* in the number of observed features, and at most *cubic* in the number of states that are included in the state vector. Thus, the number of poses that are included in the state is the most significant factor in determining the computational cost of the algorithm. Since this number is a selectable parameter, it can be tuned according to the available computing resources, and the accuracy requirements of a given application. If required, the length of the filter state can be also adaptively controlled during filter operation, to adjust to the varying availability of resources[1].

One source of difficulty in recursive state estimation with camera observations is the nonlinear nature of the measurement model. Vision-based motion estimation is very sensitive to noise, and, especially when the observed features are at large distances, false local minima can cause convergence to inconsistent solutions [121]. The problems introduced by nonlinearity have been addressed in the literature using techniques such as Sigma-point Kalman filtering [68], particle filtering [36], and the inverse depth representation for features [99]. Two characteristics of the described algorithm increase its robustness to linearization inaccuracies: (i) the inverse feature depth parametrization used in the measurement model (cf. Appendix C.2) and (ii) the *delayed linearization* of measurements [93]. By the algorithm's construction, multiple observations of each feature are collected, prior to using them for EKF updates, resulting in more accurate evaluation of the measurement Jacobians.

4.5 Processing loop-closure information

In this section, we describe a system that employs the MSC-KF in conjunction with a module that processes loop-closure information, to provide estimates with long-term bounded errors (cf. Fig. 4.2). We point out that a simple approach for utilizing loop-closure information

[1]It should be clear that, in order to maintain optimality, N must be greater or equal to the maximum feature-track length. If the window size is smaller, not all the available constraints are utilized.

would be to include in the MSC-KF state vector a number of landmarks (stable features), similarly to SLAM, and to use the re-observations of these landmarks for improving the estimation accuracy. However, this approach has two limitations: First, even if we knew in advance which landmarks will be re-observed, their inclusion in the EKF state vector would require updating their position estimates (and the associated covariance matrix) *every* time a filter update takes place. This would incur a significant computational cost. Secondly, and most importantly, we typically *cannot* predict which landmarks will be re-observed in the future. As a result, we would need to maintain a large number of landmarks in the state vector, many of which would never be seen again.

For these reasons, we have opted for a different approach when processing loop-closure information. In particular, a separate module of our system uses the recorded images, as well as the history of camera-pose estimates, to detect when an area is re-visited (cf. Fig 4.2). Since the MSC-KF estimates are typically very accurate (e.g., errors less than 0.5% of the distance traveled) detecting *candidate* loop-closures along the trajectory can be performed very efficiently, based on simple criteria, as described in Section 4.5.1. Once a candidate location is identified, only then images are processed to detect features observed during both visits. These feature re-observations are subsequently processed in a BA algorithm, along with the IMU measurements and the features that passed the Mahalanobis gating test in the MSC-KF.

The proposed dual-layer estimator approach offers two key benefits: First, the processing is essentially trajectory-driven: the computational cost of loop closing is incurred *only* when loop closing occurs, which is typically an infrequent event. Secondly, we note that, as the camera keeps exploring new areas, all features are treated as transient features, and processed by the MSC-KF. When the camera revisits a location, only then are the stable features identified, by matching between images recorded in the first and second visits. As a result, the approach does not necessitate *predicting* which features are stable enough for re-detection.

In addition to using loop-closure information, the use of an iterative BA algorithm leads to improved linearization. Since the MSC-KF algorithm is an EKF-based estimator, it linearizes the measurements only once, and the gradual buildup of linearization errors can

eventually lead to inconsistent estimates. To reduce the effect of linearization inaccuracies, BA can be run intermittently (or continuously, as a background process), even when no loop closure occurs, and its results can be used to "reset" the MSC-KF state and covariance estimates, and remove any accumulated linearization errors. In fact, this mode of operation, in which BA is run in parallel with the MSC-KF and periodically updates its estimates, is ideally suited for implementation in multi-core architectures, which are rapidly becoming prevalent.

4.5.1 Stable feature re-detection

We now describe the approach used in our implementation for matching stable features when the camera revisits an area. We point out that this is *not* the main contribution of this work, and alternative algorithms could be employed for this purpose. The method presented here is well-suited to the scenario of a vehicle moving on a road network, which is the case in the experimental results presented in Section 4.6.

In our system, candidate loop-closure sections of the trajectory are identified based on two criteria: (i) spatial closeness of the trajectory, and (ii) motion in approximately the same direction. The second criterion is necessary, because when using perspective cameras (i.e., not omnidirectional), typically no features can be reliably matched if the vehicle re-visits an area while moving in a substantially different direction. To limit the search space, candidate loop closures are sought only in portions of the trajectory where the vehicle either stops or turns. Once the trajectory segments that are candidates for loop-closure are identified, a number of regularly-spaced images is selected in each segment, and SIFT keypoints are extracted from them [84]. Those keypoints that are reliably matched in at least three images, both within and across segments, are then passed for processing to the BA algorithm. Using this approach, typically a few hundreds of stable features are successfully detected every time a vehicle re-visits an area.

4.5.2 Bundle adjustment

We now describe the formulation of the batch Maximum a Posteriori (MAP) estimator for processing the inertial and visual measurements, which is employed in the second layer of

the system architecture. We consider the case in which K IMU measurements and K images are available, recorded at every time-step in the interval[2] $[t_1, t_K]$. The MAP estimate for all IMU states and all feature positions can be determined by minimizing the cost function:

$$\begin{aligned} J =& \left|\left|\mathbf{X}_{\text{IMU}_1} - \mathbf{Z}_1\right|\right|_{\mathbf{R}_{\text{prior}}} + \sum_{\ell,j} \left|\left|\mathbf{z}_\ell^{(j)} - h_{\text{proj}}({}^G\mathbf{p}_{f_j}, \Pi_\ell)\right|\right|_{\mathbf{R}_\ell^{(j)}} \\ &+ \sum_{\ell=1}^{K-1} \left|\left|\mathbf{X}_{\text{IMU}_{\ell+1}} - \psi(\mathbf{X}_{\text{IMU}_\ell}, \boldsymbol{\omega}_m, \mathbf{a}_m)\right|\right|_{\mathbf{Q}_\ell} \end{aligned} \tag{4.32}$$

where $\left|\left|\mathbf{x}\right|\right|_{\mathbf{A}}$ denotes the matrix-weighted norm $\mathbf{x}^T\mathbf{A}^{-1}\mathbf{x}$. The three terms in this cost function correspond to the following types of information that is available to the system:

- The first term in J expresses the prior information about the initial state of the IMU. Typically, we have an estimate for the pose and velocity of the IMU at the start of the system's operation, while for the IMU biases such prior information is obtained from the sensor specifications, or by sensor calibration. In (4.32) the prior estimate and its covariance are denoted by $\mathbf{Z}_1$ and $\mathbf{R}_{\text{prior}}$, respectively.

- The second term in (4.32) is the weighted squared error between the actual and predicted feature measurements, and expresses the constraints due to the visual observations. This term is the cost that is typically minimized by photogrammetric bundle-adjustment algorithms [162]. We note that the indices ℓ and j in this term assume appropriate values to index all the available feature measurements. This includes both the feature tracks provided by the MSC-KF, as well as the stable feature re-observations provided by the loop-closure module.

- The last term in (4.32) expresses the constraints due to the IMU process model. Each of the $K-1$ summands is the weighted difference between the estimated IMU state at time-step $t_{\ell+1}$, and the IMU state predicted using the inertial measurements. To compute this predicted state $\psi(\mathbf{X}_{\text{IMU}_\ell}, \boldsymbol{\omega}_m, \mathbf{a}_m)$, we numerically integrate the IMU system model over the time interval $[t_\ell, t_{\ell+1}]$, starting from the estimate $\mathbf{X}_{\text{IMU}_\ell}$. The covari-

[2]To simplify the presentation in this section, we assume that IMU measurements and images are concurrently recorded. In a real implementation, however, IMU measurements are most often available at a higher rate than images. This case is treated analogously, by performing multiple propagation steps in the computation of *each* of the summands of the third term in (4.32).

ance matrix $\mathbf{Q}_\ell$, which expresses the uncertainty of the IMU-state-change estimate, is similarly computed by numerically integrating the Lyapunov equation (cf. (C.6)), starting from a zero initial value. The Jacobian of the term $\psi(\cdot)$, needed by the iterative minimization algorithm, is computed by numerically integrating (C.7), starting with the identity matrix as an initial value.

In order to minimize the cost function J with respect to all IMU states and all feature positions, we employ Gauss-Newton iterative minimization. Since the vast majority of the features observed are tracked in a small number of frames, in each iteration we utilize the technique of first marginalizing out all features, solving for the IMU states, and then back-substituting for the feature positions, similarly to [37]. This leads to a sparse structure for the Hessian matrix of the system, which we solve using sparse skyline Cholesky factorization [162]. Because the iterative minimization uses as an initial guess the MSC-KF output, which is typically very accurate, only a few iterations (usually 3-4) are required for convergence.

4.5.3 Feedback to the MSC-KF

Once the minimization has converged, the IMU and camera state estimates corresponding to the current MSC-KF sliding window are fed back to the first layer. Moreover, the corresponding covariance matrix is computed, and is used to replace the current MSC-KF covariance matrix. The computation of the covariance matrix can be sped up significantly, by taking into consideration the properties of Cholesky factorization. Specifically, from the Gauss-Newton iteration, the Cholesky factor of the Hessian matrix corresponding to the history of all IMU states is available:

$$\mathbf{A} = \mathbf{G}^T\mathbf{G} \Rightarrow \begin{bmatrix} \mathbf{A}_{oo} & \mathbf{A}_{oa} \\ \mathbf{A}_{oa}^T & \mathbf{A}_{aa} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{oo}^T & \mathbf{0} \\ \mathbf{G}_{oa}^T & \mathbf{G}_{aa}^T \end{bmatrix} \begin{bmatrix} \mathbf{G}_{oo} & \mathbf{G}_{oa} \\ \mathbf{0} & \mathbf{G}_{aa} \end{bmatrix}$$

where blocks denoted by the subscript "oo" correspond to older poses, blocks denoted by "aa" correspond to the poses that are currently active in the MSC-KF sliding window, and "ao" corresponds to the cross-terms between these. Employing the properties of the block-matrix inversion and substituting from the above expression, the covariance matrix of the

active states is computed as:

$$\mathbf{P}_{aa} = \left(\mathbf{A}_{aa} - \mathbf{A}_{oa}^T \mathbf{A}_{oo}^{-1} \mathbf{A}_{oa}\right)^{-1} = \mathbf{G}_{aa}^{-1} \mathbf{G}_{aa}^{-T} \tag{4.33}$$

At this point, we note that the Cholesky factorization of $\mathbf{A}$ (and thus, the block $\mathbf{G}_{aa}$) is already available, since it has been computed during the minimization process. Therefore, the computational cost for obtaining the covariance matrix of the MSC-KF state is minimal (note that $\mathbf{G}_{aa}$ is a triangular matrix).

4.5.4 Marginalization of old states

An important issue is that the computational cost of BA increases with the number of states in the estimated trajectory (due to sparsity, the increase is approximately linear in time). Thus, for very long trajectories the computational burden can become intractable. To address this problem, we can choose to permanently marginalize out certain older poses and the features seen from these poses. By limiting the number of estimated states, this process allows the processing time for BA to remain bounded. Clearly, after marginalization the linearization of the measurements that involve the removed states is not recomputed, and hence marginalization leads to an approximation of the cost function. However, if only older, "mature" states (i.e., states for which the estimates are deemed accurate) are removed, the approximation will be very good. Finally, we note that once a pose is marginalized, we no longer have the ability to close loops using this pose. Therefore, care should be taken in order to always maintain a set of poses in areas that are likely to be revisited by the vehicle.

4.6 Experimental results

We hereafter present experimental results demonstrating the performance of the MSC-KF and the dual-layer estimator architecture for vision-aided inertial navigation. The system has been employed for estimating the trajectory of a vehicle moving in an urban environment. Some example images from the recorded sequences are shown in Fig. 4.3. Images are recorded by a firewire camera with resolution 640×480 pixels, and processed at a rate of 7.5 Hz. Inertial measurements are provided by an Inertial Science ISIS IMU, at a rate of

Figure 4.3: Some images from the datasets collected in the experiments.

100 Hz. For the results presented here, transient features were extracted using the Harris operator [56], and matched using normalized cross-correlation, while stable feature extraction and matching was performed using the SIFT algorithm [84]. Processing was carried out off-line, on a 2 GHz CPU, and the algorithm run-times reported in what follows correspond only to the time required by the estimation algorithms.

Experiment 1

In this experiment, we demonstrate the performance of the MSC-KF when used without feedback from the BA module. For this experimental run the vehicle drove for about 16 minutes, covering a distance of approximately 7.6 km. A total of 8151 images were processed, with an average of 800 features being tracked in each image. These feature measurements were processed by the MSC-KF, in which a sliding window of 32 camera poses was maintained. The MSC-KF required approximately 191 seconds for processing the entire image sequence, which corresponds to a processing throughput approximately five times faster than real-time.

In Fig. 4.4, the estimated trajectory is shown in the blue line, and compared to the GPS

ground truth, denoted by red dots. Fig. 4.5 presents the same results superimposed on a satellite map of the area. In this figure, the MSC-KF estimates are plotted in black, while the GPS ground truth is represented by white dots. Even though Figs. 4.4 and 4.5 plot the position estimates on the ground plane, we stress that the MSC-KF carries out estimation of the full 3D pose of the system. In fact, in this particular experiment, the elevation profile of the trajectory involves significant changes, of about 20 m. This can be seen in Fig. 4.6, in which the elevation estimate is plotted, and compared to the GPS measurements[3]. It should be pointed out that, unfortunately, the GPS ground truth is not of high quality. Due to the dense foliage, the GPS satellite signals were often obstructed, which resulted in frequent GPS drop-outs, large measurement errors (especially for the altitude), and additionally caused the GPS measurements to exhibit an irregular time-lag. As a result, while the GPS ground truth can be employed for qualitatively evaluating the accuracy of the estimates, it is not possible to reliably compute the estimation errors.

Despite the limitations of the GPS ground truth, from Figs. 4.4-4.6 it becomes clear that the MSC-KF can very accurately track the trajectory of the vehicle in 3D. By comparing the estimates of the vehicle trajectory to the GPS measurements, we conclude that the position errors remain smaller than approximately 20 m, throughout the trajectory. For a trajectory of length 7.6 km, this corresponds to an error smaller than 0.27% of the distance traveled. The attained estimation accuracy is also seen in Figs. 4.7-4.9, that plot the 3σ values for the IMU attitude, position, and velocity, respectively. The plotted values are equal to 3 times the square root of the corresponding elements of the MSC-KF covariance matrix.

The results presented above demonstrate that the MSC-KF is capable of very accurate pose estimation in real-time. It should be pointed out that this level of accuracy is achieved *without* utilizing any additional localization information (e.g., knowledge of the map, vehicle wheel odometry, or kinematic model of the car) and by processing images at a relatively low rate of 7.5 Hz. This demonstrates the benefits of fusing visual with inertial measurements using the tightly-coupled formulation of the MSC-KF.

[3]We note that the position estimates were transformed to a local frame having its x and y axes aligned with the east-west, and north-south directions, respectively, for visualization purposes only. The filter computes the estimates in the ECEF frame, as described in Section 4.4.

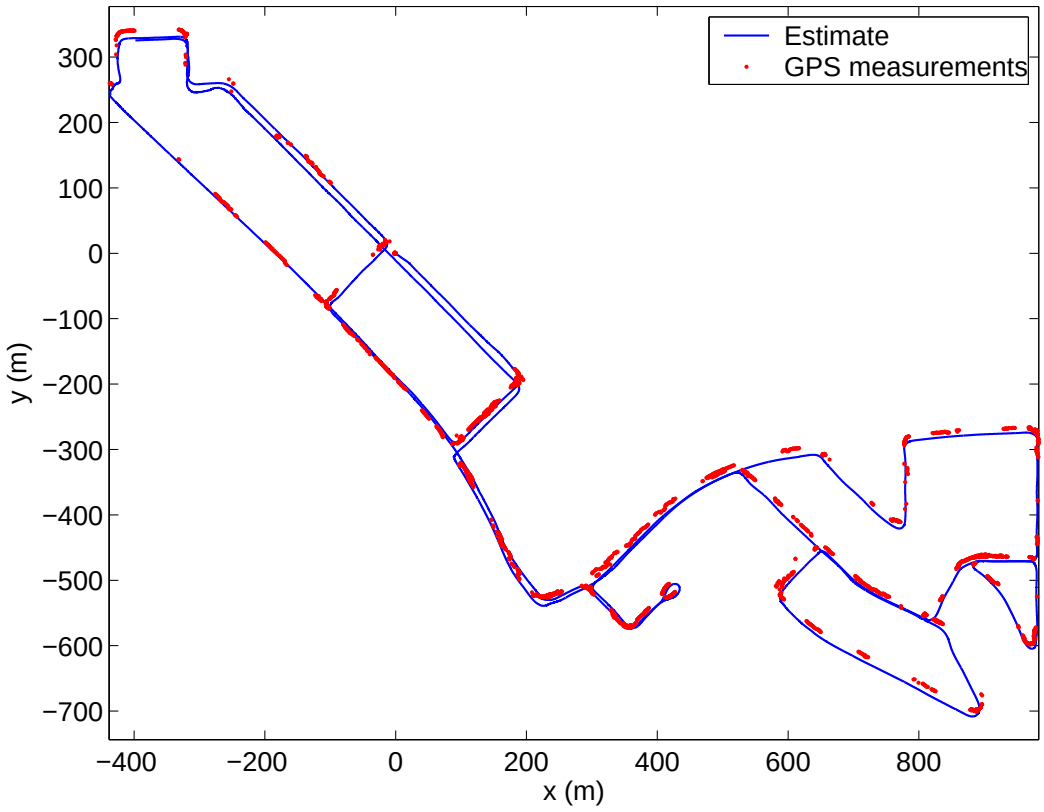


Figure 4.4: The estimated trajectory (blue line), and GPS ground truth (red dots), for the first experiment. The starting position is at $(0, 0)$.

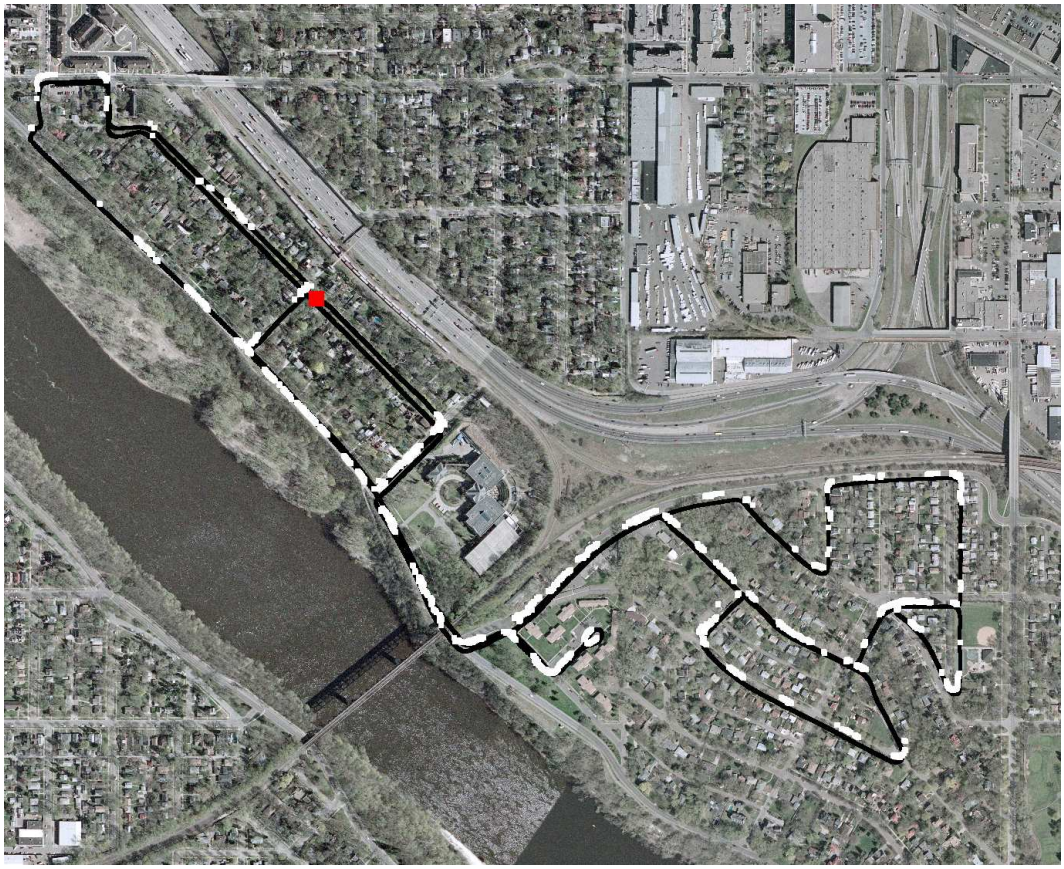


Figure 4.5: The estimated trajectory for the first experiment, overlaid on a satellite image of the area. The initial position of the vehicle is denoted by a red square. The white dots represent the GPS measurements.

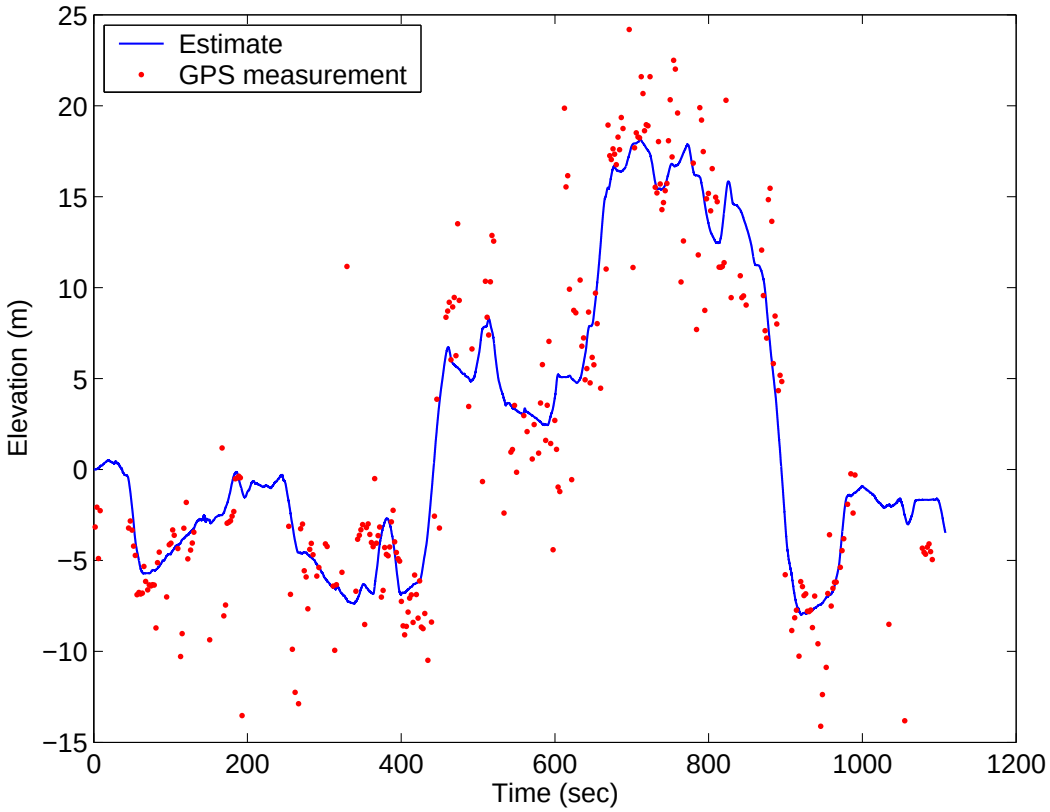


Figure 4.6: The estimated elevation (blue line) and corresponding GPS measurements (red dots), for the first experiment.

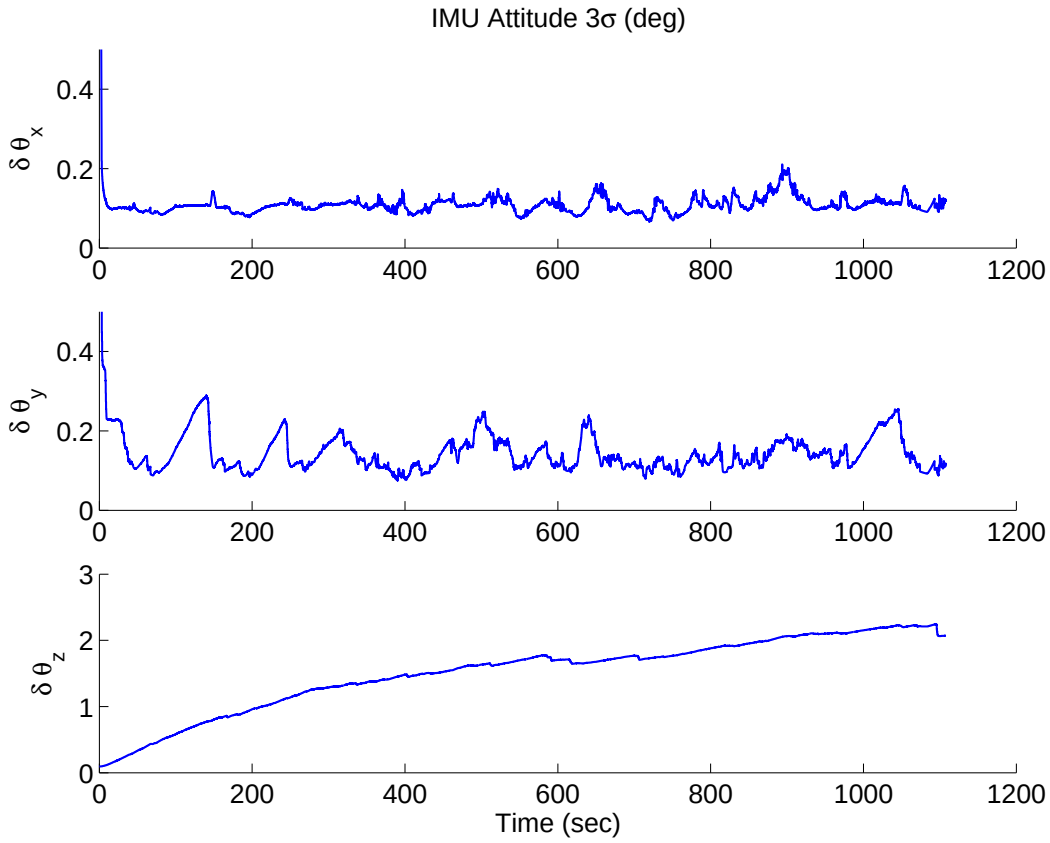


Figure 4.7: The 3σ values for the IMU attitude, computed by the MSC-KF during the first experiment. The three subplots correspond to the x, y, and z axes, respectively.

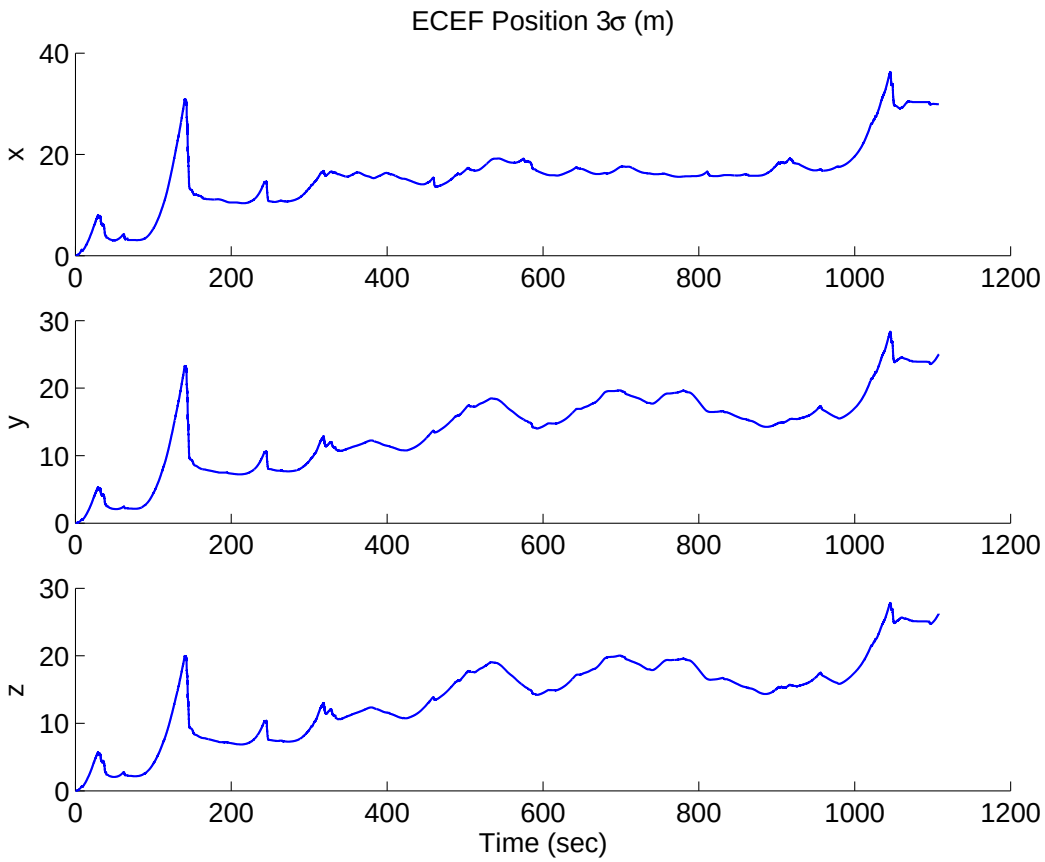


Figure 4.8: The 3σ values for the IMU position, computed by the MSC-KF during the first experiment. The three subplots correspond to the x, y, and z axes, respectively.

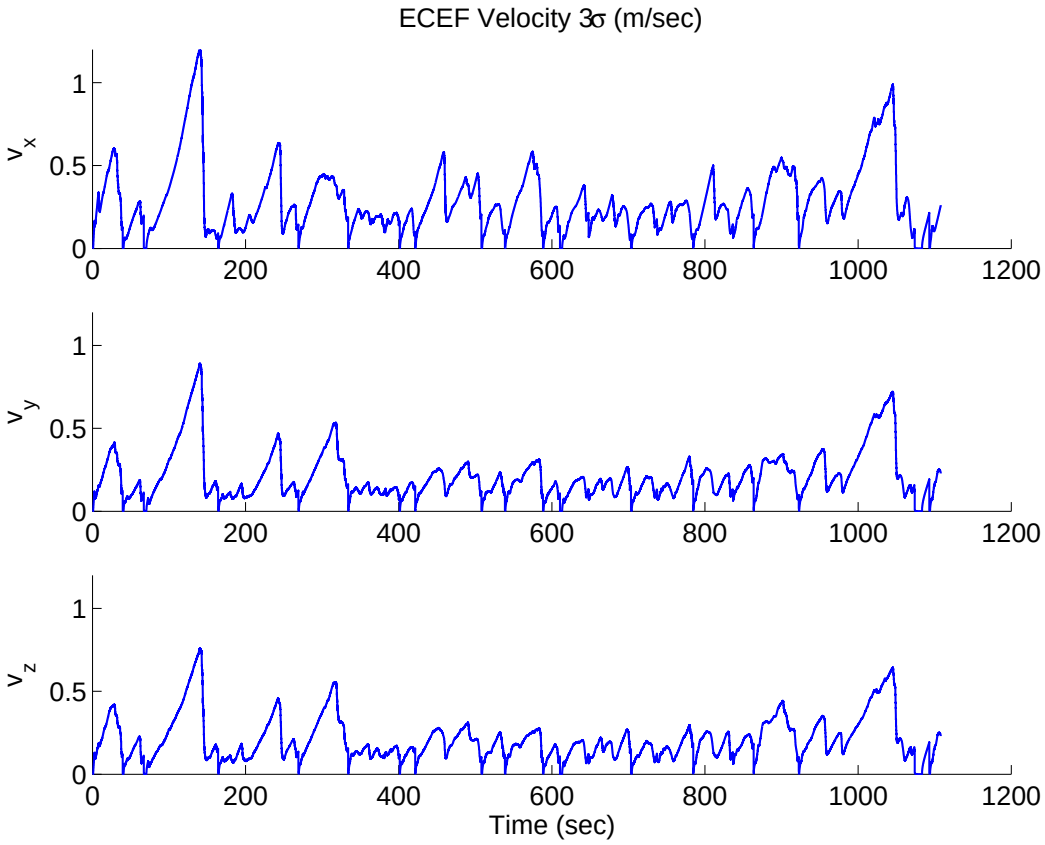


Figure 4.9: The 3σ values for the IMU velocity, computed by the MSC-KF during the first experiment. The three subplots correspond to the x, y, and z axes, respectively. The "sawtooth-pattern" appearance of the plots is due to the fact that the velocity is more accurately estimated in parts of the trajectory where the car is stopped or moving very slowly.

Experiment 2

We now present results from a second experiment, where the MSC-KF is used in conjunction with the BA estimator, thus enabling the processing of loop-closure information. In this test, the car covered a distance of 6.6 km within 17.5 minutes, and two loops were detected. In the MSC-KF a sliding window of 32 poses is maintained. On the other hand, the non-linear minimization is run every time a loop is successfully detected, or every 1000 images, in order to periodically reduce the buildup of linearization errors. Moreover, in order to prevent the computational cost of the minimization routine from becoming too large, we employ marginalization of older poses. In particular, all poses that are both older than 2 minutes, and do not lie in a segment of the trajectory that is a loop-closure candidate, are marginalized out.

With these settings, pose estimation requires a total of 328 sec of CPU time, which corresponds to approximately 3 times faster than real-time processing. Of these 328 sec, the MSC-KF and BA algorithms account for 221 sec and 107 sec, respectively. We therefore see that the computational overhead of the second layer of the dual-layer estimator is less than 50%, in this case. This overhead is relatively small considering that, by using the entire two-layer architecture instead of the first layer alone, we obtain the benefits of (i) using loop closure information, and (ii) obtaining better linearization.

The estimated trajectory is plotted versus GPS ground truth in cartesian axes in Fig. 4.10, and overlaid on a satellite image of the area, in Fig. 4.11, respectively. For comparison, Fig. 4.12 shows the trajectory estimate arising when the MSC-KF is used alone. Close inspection of the data reveals that when the dual-layer estimator is utilized, the position errors remain smaller than approximately 10 m, throughout the trajectory, while when only the MSC-KF is used, the maximum error is approximately 20 m. Thus, the benefit from the use of loop closure information becomes evident. However, it should be pointed out that even the 20 m of position error, recorded when the MSC-KF is used without feedback from the nonlinear minimization, corresponds to only 0.3% of the trajectory length.

In Figs. 4.13-4.15, the 3σ values for the IMU attitude, position, and velocity, are shown, respectively. The solid blue lines in these figures correspond to the case when loop closure

information is utilized, while the dashed black lines correspond to the result of the MSC-KF. The loop closure events (occurring at 895 and 957 sec) can be clearly identified by the sharp drop in uncertainty. Interestingly, we observe that the state estimates that primarily benefit from the additional localization information are those of (i) the IMU rotation about its z axis (which is approximately parallel to the gravity vector during the experiment), and (ii) the IMU position. This makes sense, intuitively, since these states are the ones that experience the most pronounced uncertainty accumulation, when using visual/inertial odometry alone. In contrast, the IMU tilt angles (i.e., the rotation about its x and y axes) are estimated very accurately throughout the trajectory, due to the presence of the gravity field.

Figs. 4.13-4.15 also demonstrate the fact that the MSC-KF slightly *underestimates* the covariance matrix. Specifically, for the first 895 sec of the trajectory (i.e., prior to the loop detection), the MSC-KF covariance estimate is slightly smaller than the estimate arising when the covariance is periodically reset from the BA (cf. Section 4.5.3). This indicates that the MSC-KF tends to become *inconsistent*, which is a known issue with EKF-based pose estimation [7, 18, 66, 70]. A potential remedy for this problem would be to extend the method of [64] to the case of 3D SLAM. In [64], only the *first* estimate of each state is used in computing all Jacobians. For the case of 2D SLAM, this technique is shown to result in better consistency than the standard-EKF approach, where the *latest* estimates are employed for evaluating the Jacobians. In out future work, we plan to investigate the performance of such a *first-estimates Jacobian* (FEJ)-EKF in the case of 3D vision-aided inertial navigation.

As a final remark, it is worth noting that in the experiments presented here, the camera motion is almost parallel to the optical axis, a condition which is particularly adverse for image-based motion estimation algorithms [121, 164]. Moreover, in the images a significant number of moving objects appear, such as cars, pedestrians, and trees moving due to the wind. However, the fact that all the measurements of each feature are simultaneously processed allows us to achieve better linearization, as well as more robust outlier rejection, compared to EKF-SLAM algorithms where each feature measurement is processed immediately. This contributes to the high precision of the estimates, as seen in the results of this

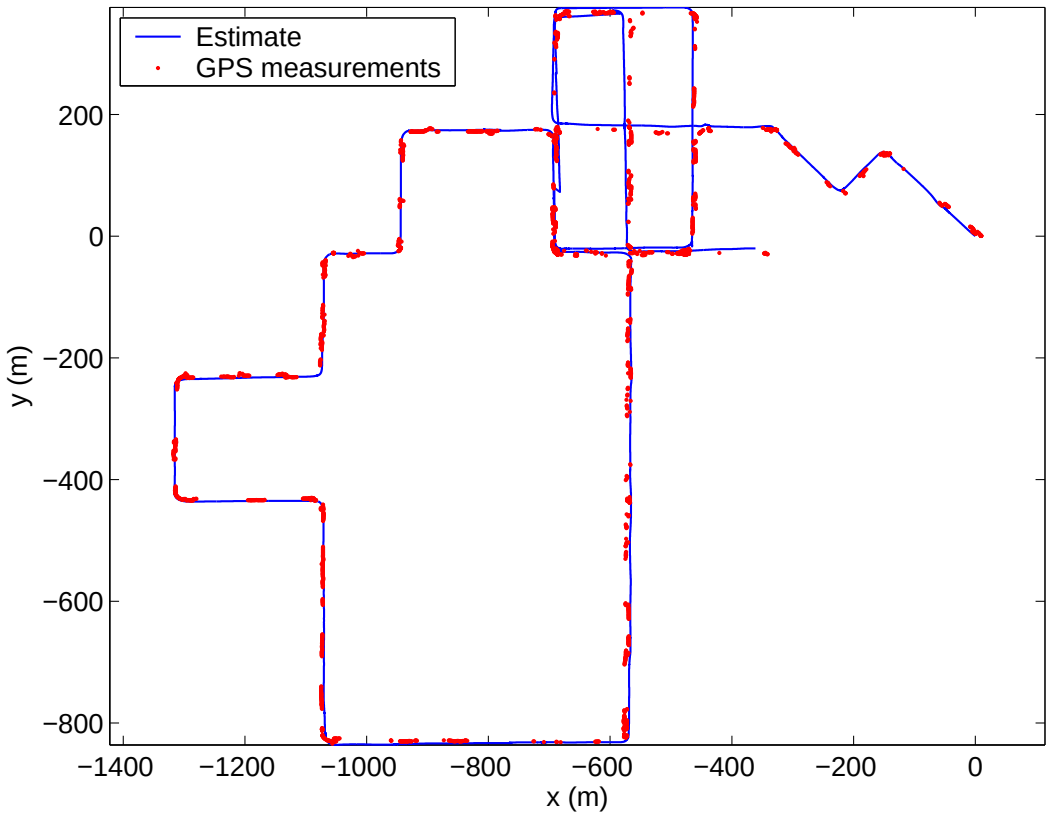


Figure 4.10: The estimated trajectory (blue line), and GPS ground truth (red dots), for the second experiment. In this plot the result of the complete two-layer system, which utilizes loop closure information, is shown. The starting position is at $(0, 0)$.

section.

4.7 Summary

In this chapter we have presented the Multi-State Constraint Kalman filter (MSC-KF), an EKF-based estimator that optimally processes transient-feature measurements. The key contribution of the MSC-KF is a measurement model that is able to express the constraints that arise when a static feature is observed from multiple consecutive camera poses. This model does not require including the 3D feature positions in the state vector of the EKF, and is optimal, up to the errors introduced by linearization. The resulting EKF-based pose estimation algorithm has computational complexity *linear* in the number of features, and is thus suitable for processing a large number of feature measurements in real time.

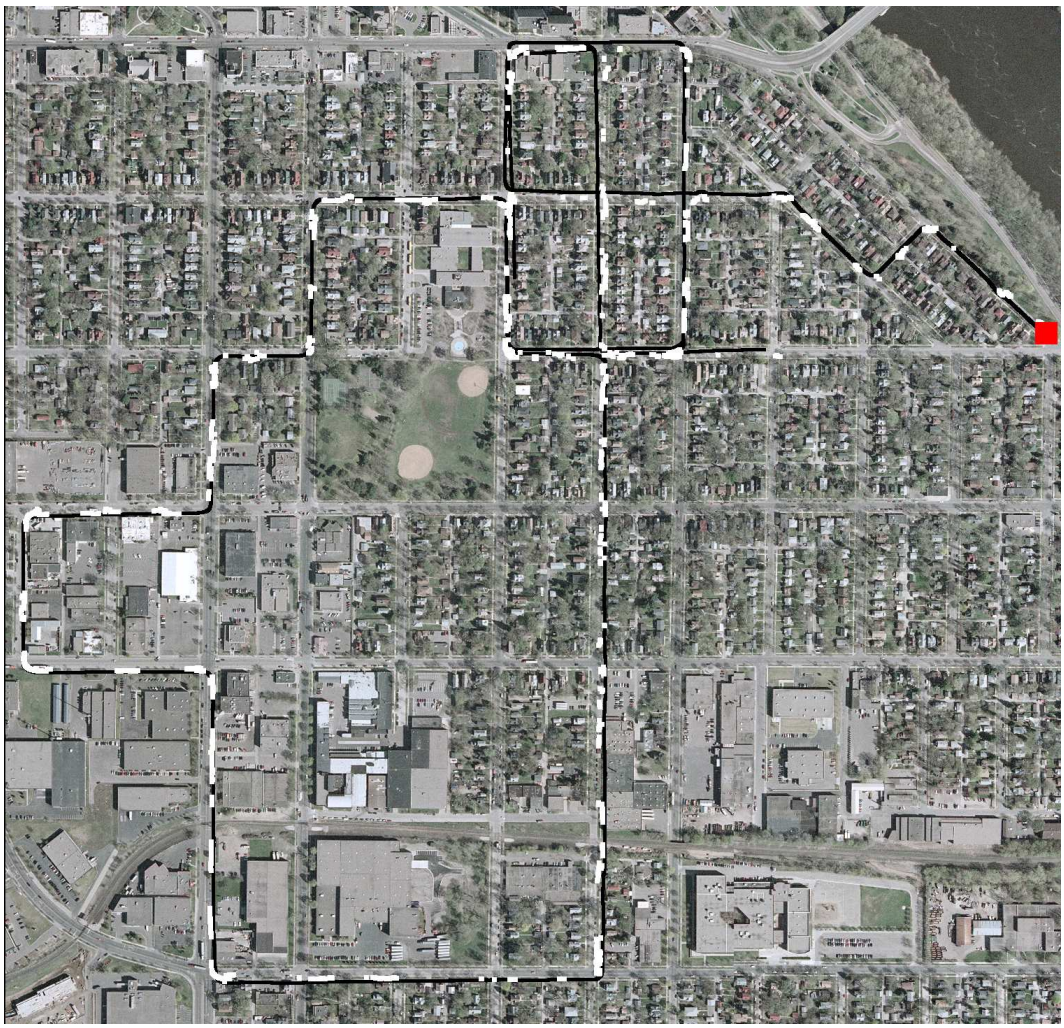


Figure 4.11: The estimated trajectory for the second experiment, overlaid on a satellite image of the area. The initial position of the vehicle is denoted by a red square. The white dots represent the GPS measurements.

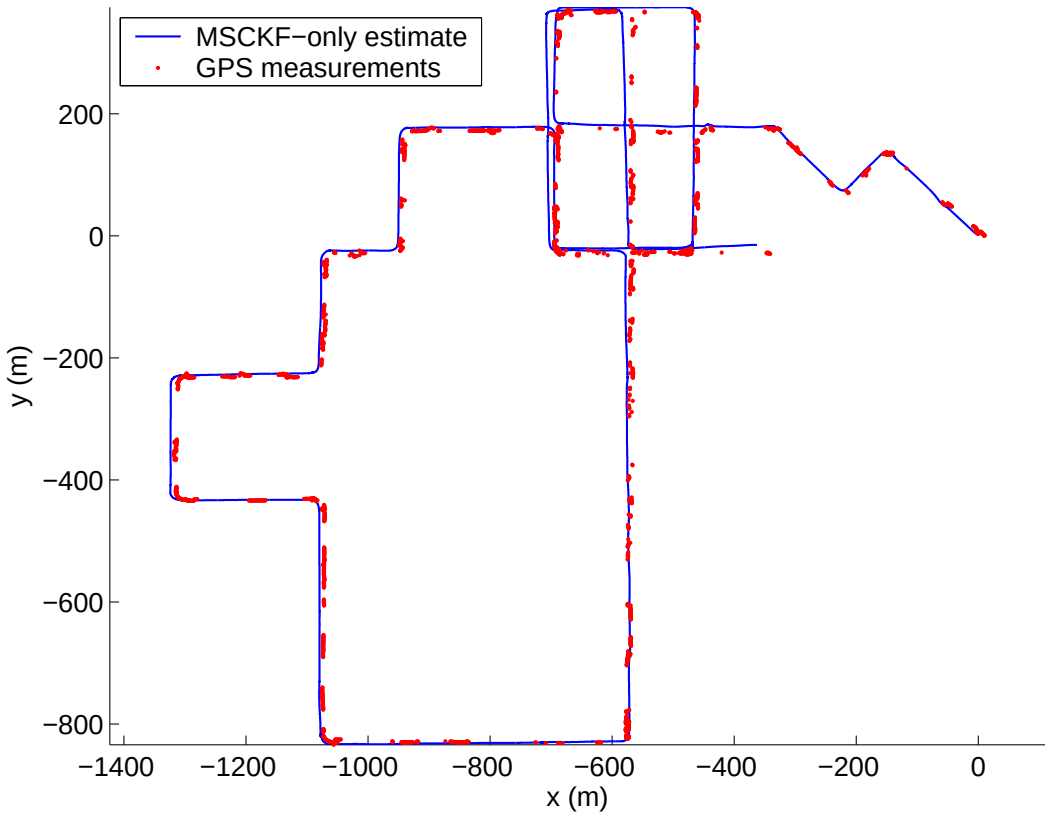


Figure 4.12: The MSC-KF estimate for the trajectory (blue line), and GPS ground truth (red dots), for the second experiment. The starting position is at $(0, 0)$.

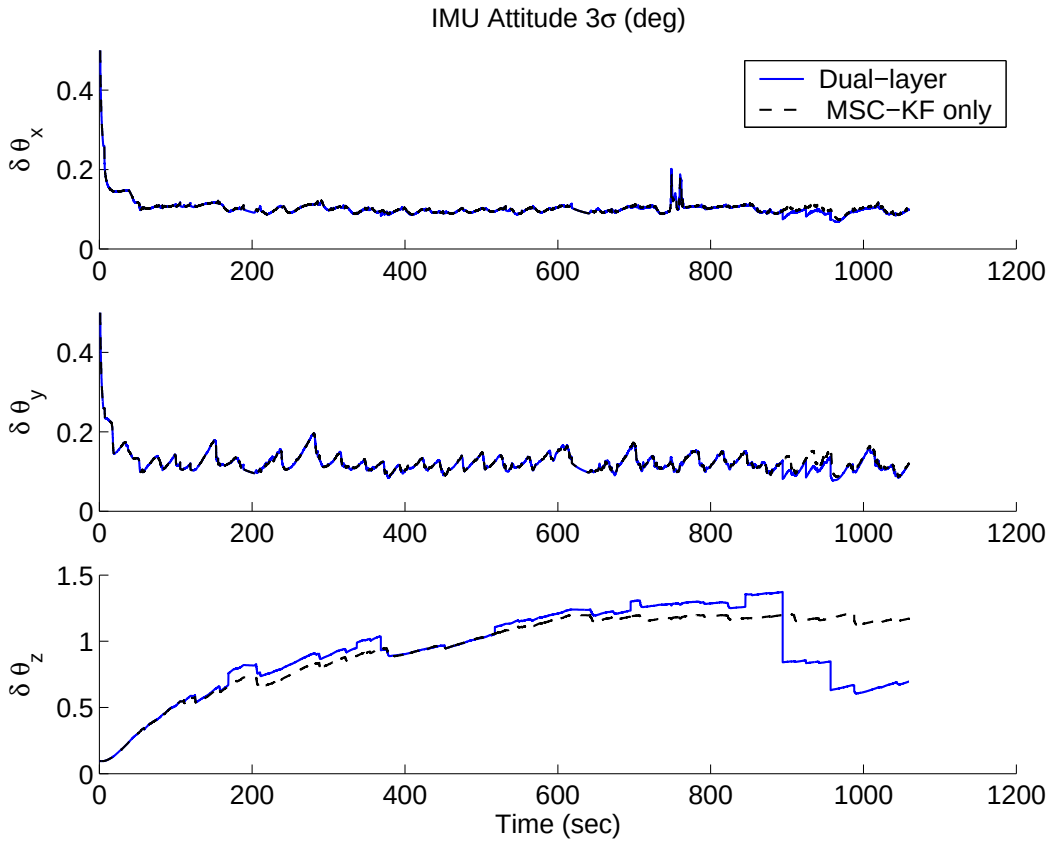


Figure 4.13: The 3σ values for the IMU attitude during the second experiment. The solid blue lines correspond to the case where loop-closure information is processed, while the dashed ones correspond to the MSC-KF.

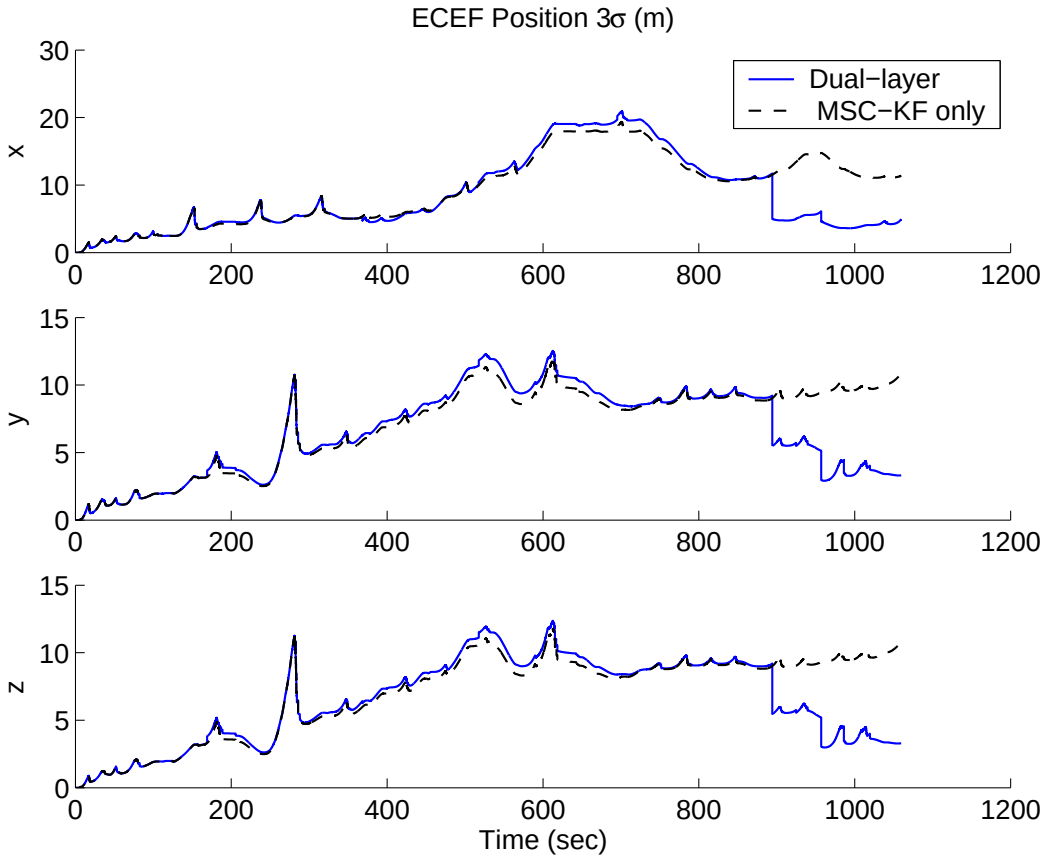


Figure 4.14: The 3σ values for the IMU position during the second experiment. The solid blue lines correspond to the case where loop-closure information is processed, while the dashed ones correspond to the MSC-KF.

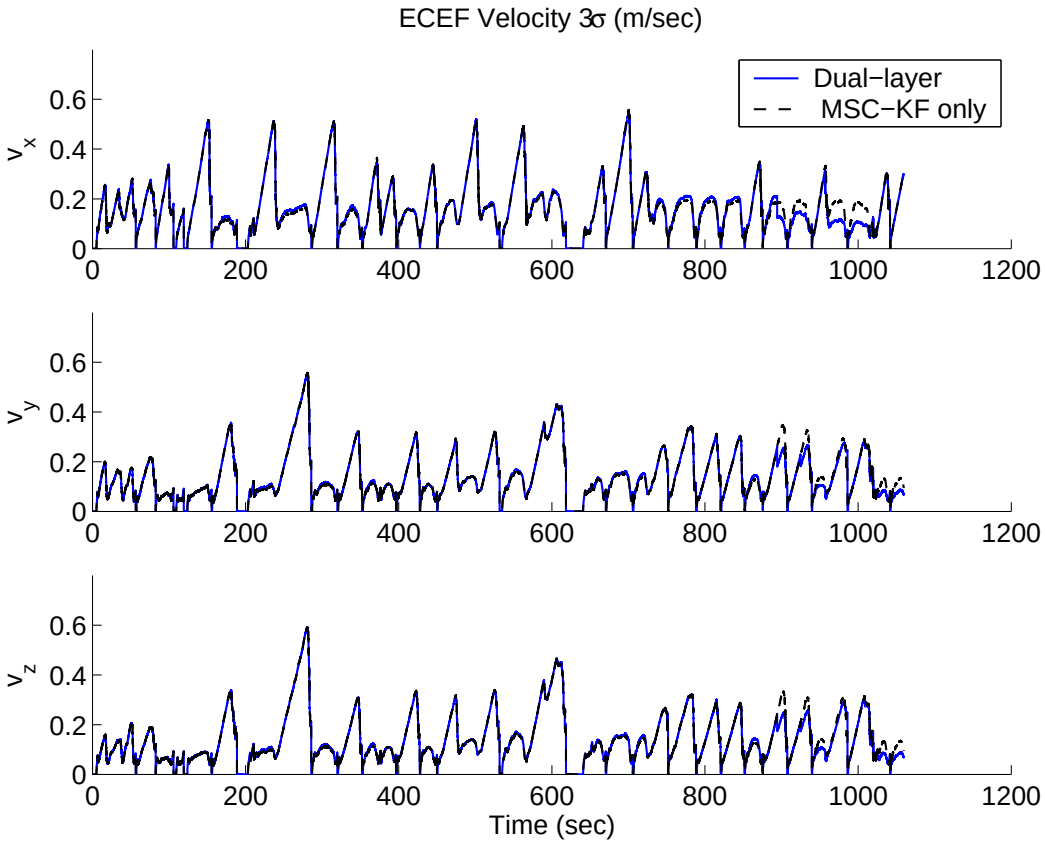


Figure 4.15: The 3σ values for the IMU velocity during the second experiment. The solid blue lines correspond to the case where loop-closure information is processed, while the dashed ones correspond to the MSC-KF.

Moreover, we have demonstrated how the MSC-KF can be integrated in a localization system. Specifically, we have described a two-layer estimator architecture, where in the first layer the MSC-KF performs combined visual/inertial odometry in real time, while in the second layer a batch nonlinear minimization algorithm (bundle adjustment) is intermittently run, for processing loop-closure constraints. As key advantages of the proposed two-layer architecture we can identify: (i) The estimates are available in real time, and at the IMU data rate, (ii) The estimation errors remain bounded when loops are detected, thus enabling long-term localization, and (iii) The computational overhead of utilizing loop-closure constraints is minimal, even though loop closure significantly improves localization accuracy. These properties render the proposed architecture suitable for large-scale localization applications.

Chapter 5

Concluding Remarks

5.1 Summary of contributions

The work presented in the preceding chapters has focused on providing methodologies for *analytical prediction* and *optimization* of the accuracy of mobile robot localization. The key contributions of the work are summarized in the following:

- **Analytical performance characterization**

  Chapter 2 presented theoretical tools for evaluating the performance, in terms of positioning accuracy, of Cooperative Localization (CL), Simultaneous Localization and Mapping (SLAM), and Cooperative SLAM (C-SLAM). Analytical expressions were derived that determine the *guaranteed accuracy* of localization, as *closed-form functions* of (i) the accuracy of all available sensors, (ii) the number of robots and/or landmarks, (iii) the spatial distribution of the robots and/or landmarks, and (iv) the structure of the graph describing the robot-to-robot and robot-to-landmark measurements. These expressions enable predicting the accuracy that a given robot system will attain in an application, and can therefore serve as valuable tools in the design of a localization system.

  The derived analytical results allowed us to identify key properties of the positioning uncertainty, especially for CL. For instance, it has been shown analytically that during CL the rate at which the position uncertainty increases in time is independent of the

accuracy of the exteroceptive measurements, and only depends on the number of robots in the team and the quality of their dead-reckoning capabilities. This indicates that during system design, it may be more important to invest in accurate odometry sensors, rather than exteroceptive ones. The derived analytical expressions make it possible to infer such "rules of thumb" for robot design, and facilitate developing intuition about the several factors affecting localization performance.

- **Optimal localization in robot formations**

  In Chapter 3, we examined a special case of interest in CL, that of localization in robot formations. First, it was shown that in this case a precise description of the steady-state estimation accuracy can be obtained, by considering an equivalent *continuous-time* system model. Then, it was shown how this continuous-time formulation can be employed for determining optimal sensing strategies, that can achieve the maximum possible localization accuracy under constraints imposed by the robots' limited computational and communication resources.

  Specifically, it was shown that the frequency at which each sensor input is processed determines the accuracy of the corresponding measurement in the continuous-time model. This enables us to formulate a *convex optimization problem* for the measurement frequencies, where the constraints on the communication, processing, and power resources of the team are naturally incorporated. Moreover, this problem was exactly reformulated to a semidefinite programming (SDP) problem, whose unique global solution can be computed using well-studied and efficient minimization algorithms. Experimental and simulation studies have indicated that the proposed approach yields results superior to those of competing methods, and additionally, it can be employed in order to identify the effects of different design choices on the accuracy of pose estimation.

- **Optimal use of the transient-feature measurements**

  In Chapter 4, we focused on the issue of vision-aided inertial navigation for mobile robots. One of the key challenges in this domain is the very large number of feature measurements, which can overwhelm the limited computational capabilities of

a real-time system. Based on the observation that the vast majority of features are only "short-lived," we have proposed an algorithm that *optimally* processes the measurements of transient features, i.e., of features only observed in a small number of consecutive frames. This algorithm, termed Multi-State Constraint Kalman filter (MSC-KF), utilizes the constraints that arise when a static feature is observed from consecutive camera poses in a statistically optimal fashion (up to linearization). The computational complexity of the MSC-KF is only *linear* in the number of features, which renders the method suitable for processing a large number of feature measurements in real time.

Moreover, we have described a two-layer estimator architecture, incorporating the MSC-KF and a batch nonlinear minimization algorithm. The proposed architecture utilizes both the motion constraints due to the transient features, as well as the loop-closure information, which becomes available when a robot re-visits an area. The merging of these two types of positioning information produces pose estimates that are both available in real time *and* have bounded long-term errors. The operation of the system has been demonstrated in large-scale real-world experiments, involving a vehicle localizing in an urban environment.

Our aspiration is that, by providing performance guarantees and methods for the optimal allocation of the available system resources, the above-described contributions can help increase the *reliability* and *cost efficiency* of robot designs, both of which are prerequisites for their widespread use and commercial success.

5.2 Future research directions

The performance analysis of Chapter 2 has unveiled several interesting properties of position estimation in groups of communicating mobile robots. However, several open questions remain. Of particular interest is the study of state estimation in large robot networks, with stringent communication and/or computation constraints. One unexplored question in this domain is the *asymptotic performance* of state estimation as the network size increases. The effects of the network graph structure and of the communication and computation

constraints in this case are not fully understood, and we believe that the theoretical results presented in Chapter 2 can facilitate the study of such properties. For example, a recent study has utilized the results of our analysis to examine the effect of special graph topologies on localization [76].

Moreover, as the network size increases, the use of centralized estimators becomes infeasible, and approximate, *distributed estimation* schemes are necessary. So far in our work, an optimal (up to linearization) estimator was considered (i.e., the estimator computes the Minimum Mean Squared Error (MMSE) estimate given the measurements provided to it), and in Chapter 3 a methodology was proposed for obtaining the best sensing strategy within this framework. However, this is not the only possible choice for estimator design. An alternative approach would be to employ an approximate estimator that, instead of processing a small number of measurements in an optimal fashion, it utilizes an approximate estimation scheme that allows it to process a larger subset of the available measurements. Such an approach was recently proposed in [88], based on the key ideas presented in Chapter 3. However, the selection of the most suitable approximation scheme, to attain optimal resource utilization, remains an open issue.

Finally, vision-based real-time localization is currently one of the most active research areas in robotics. The MSC-KF, presented in Chapter 4, offers an attractive option for processing the large number of feature measurements, and is a versatile method, that can be easily incorporated as part of a localization system. For instance, in [113] the MSC-KF was used in conjunction with mapped-landmark measurements, in a system for lander-pose estimation during planetary Entry, Descent, and Landing (EDL). However, even though the complexity of the MSC-KF is linear in the number of features, the total number of available features can still be too large to allow real-time processing. To address this problem, one can seek to identify *optimal feature selection* techniques, that will allow the algorithm to concentrate only on those features that can guarantee the highest possible localization accuracy.

Appendix A

Appendices for Chapter 2

A.1 Proof of Lemma 1

The proof is carried out by induction, and requires the following two intermediate results:

- **Monotonicity with respect to the measurement covariance matrix**

  If $\mathbf{R}_1 \succeq \mathbf{R}_2$, then for any $\mathbf{P} \succeq \mathbf{0}$

$$\mathbf{\Phi}\left(\mathbf{P}-\mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_1\right)^{-1}\mathbf{H}\mathbf{P}\right)\mathbf{\Phi}^T+\mathbf{Q}\succeq \mathbf{\Phi}\left(\mathbf{P}-\mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_2\right)^{-1}\mathbf{H}\mathbf{P}\right)\mathbf{\Phi}^T+\mathbf{Q} \quad \text{(A.1)}$$

This statement is proven by taking into account the properties of linear matrix inequalities:

$$\begin{aligned}
\mathbf{R}_1 &\succeq \mathbf{R}_2 \Rightarrow \\
\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_1\right)^{-1} &\preceq \left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_2\right)^{-1} \Rightarrow \\
\mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_1\right)^{-1}\mathbf{H}\mathbf{P} &\preceq \mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_2\right)^{-1}\mathbf{H}\mathbf{P} \Rightarrow \\
\mathbf{\Phi}\left(\mathbf{P}-\mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_1\right)^{-1}\mathbf{H}\mathbf{P}\right)\mathbf{\Phi}^T &\succeq \mathbf{\Phi}\left(\mathbf{P}-\mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T+\mathbf{R}_2\right)^{-1}\mathbf{H}\mathbf{P}\right)\mathbf{\Phi}^T
\end{aligned}$$

from which (A.1) follows directly.

- **Monotonicity with respect to the state covariance matrix**

The solution to the Riccati recursion at time $k+1$ is monotonic with respect to the solution at time k, i.e., if $\mathbf{P}_k^{(1)}$ and $\mathbf{P}_k^{(2)}$ are two different solutions to the same Riccati recursion at time k, with $\mathbf{P}_k^{(1)} \succeq \mathbf{P}_k^{(2)}$, then $\mathbf{P}_{k+1}^{(1)} \succeq \mathbf{P}_{k+1}^{(2)}$. In order to prove the result in the general case, in which $\mathbf{P}_k^{(1)}$ and $\mathbf{P}_k^{(2)}$ are positive semidefinite (and not necessarily positive definite), we use the following expression that relates the one-step ahead solutions to two Riccati recursions with identical $\mathbf{H}$, $\mathbf{Q}$ and $\mathbf{R}$ matrices, but different initial values $\mathbf{P}_k^{(1)}$ and $\mathbf{P}_k^{(2)}$. It is [58]

$$\begin{aligned}&\mathbf{P}_{k+1}^{(2)} - \mathbf{P}_{k+1}^{(1)} = \\ &\mathbf{F}_{p,k}\left(\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right) - \left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^{(2)}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\right)\mathbf{F}_{p,k}^T\end{aligned} \tag{A.2}$$

where $\mathbf{F}_{p,k}$ is a matrix whose exact structure is not important for the purposes of this proof. Since we have assumed $\mathbf{P}_k^{(1)} \succeq \mathbf{P}_k^{(2)}$ we can write $\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)} \preceq 0$. Additionally, the matrix

$$\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^{(2)}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)$$

is positive semidefinite, and therefore we have

$$\begin{aligned}-\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^{(2)}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right) \preceq \mathbf{0} \Rightarrow \\ \left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right) - \left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^{(2)}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right) \preceq \mathbf{0} \Rightarrow \\ \mathbf{F}_{p,k}\left(\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right) - \left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^{(2)}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\left(\mathbf{P}_k^{(2)} - \mathbf{P}_k^{(1)}\right)\right)\mathbf{F}_{p,k}^T \preceq \mathbf{0} \Rightarrow \\ \mathbf{P}_{k+1}^{(2)} - \mathbf{P}_{k+1}^{(1)} \preceq \mathbf{0}\end{aligned}$$

The last relation implies that $\mathbf{P}_{k+1}^{(1)} \succeq \mathbf{P}_{k+1}^{(2)}$, which is the desired result.

We can now employ induction to prove the main result of Lemma 1. Assuming that, at

some time instant k, the condition $\mathbf{P}_k^u \succeq \mathbf{P}_k$ holds, we can write

$$\begin{aligned}
\mathbf{P}_{k+1}^u &= \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k^u - \mathbf{P}_k^u\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k^u\mathbf{H}^T + \mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_k^u\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}\mathbf{Q}_u\mathbf{G}^T \\
&\succeq \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_k\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}\mathbf{Q}_u\mathbf{G}^T \\
&\succeq \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_k\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T \\
&\succeq \mathbf{\Phi}_{k+1}\left(\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k\right)\mathbf{\Phi}_{k+1}^T + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T = \mathbf{P}_{k+1}
\end{aligned}$$

where the monotonicity of the Riccati recursion with respect to the covariance matrix, the property $\mathbf{Q}_u \succeq \mathbf{Q}_{k+1}$ and the monotonicity of the Riccati recursion with respect to the measurement covariance matrix have been used in the last three lines. Thus $\mathbf{P}_k^u \succeq \mathbf{P}_k \Rightarrow \mathbf{P}_{k+1}^u \succeq \mathbf{P}_{k+1}$. For $k = 0$ the condition $\mathbf{P}_k^u \succeq \mathbf{P}_k$ holds, and therefore the proof by induction is complete.

A.2 Proof of Lemma 2

We first prove a useful intermediate result:

- **Concavity of the Riccati recursion**

  We note that the right-hand side of the Riccati recursion

$$\mathbf{P}_{k+1} = \mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T \tag{A.3}$$

  can be written as

$$\begin{aligned}
&\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I} \\ \mathbf{0}\end{bmatrix} \\
&-\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1}\end{bmatrix}\begin{bmatrix}\mathbf{H}^T \\ \mathbf{0}\end{bmatrix}\left(\begin{bmatrix}\mathbf{H} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1}\end{bmatrix}\begin{bmatrix}\mathbf{H}^T \\ \mathbf{I}\end{bmatrix}\right)^{-1} \times \\
&\begin{bmatrix}\mathbf{H} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I} \\ \mathbf{0}\end{bmatrix} + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T
\end{aligned}$$

We now show that the above expression is concave with respect to the matrix

$$\begin{bmatrix} \mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1} \end{bmatrix}$$

A sufficient condition for this is that the function

$$f(\mathbf{X}) = \mathbf{A}\mathbf{X}\mathbf{B}\left(\mathbf{C}\mathbf{X}\mathbf{C}^T\right)^{-1}\mathbf{B}^T\mathbf{X}\mathbf{A}^T \tag{A.4}$$

is convex with respect to the positive semidefinite matrix $\mathbf{X}$, when $\mathbf{A}$,$\mathbf{B}$,$\mathbf{C}$ are arbitrary matrices of compatible dimensions. This is equivalent to proving the convexity of the following function of the scalar variable t [12]:

$$f_t(t) = \mathbf{A}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{B}\left(\mathbf{C}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{C}^T\right)^{-1}\mathbf{B}^T(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{A}^T \tag{A.5}$$

with domain those values of t for which $\mathbf{X}_o + t\mathbf{Z}_o \succeq \mathbf{0}, \mathbf{X}_o \succeq \mathbf{0}$. $f_t(t)$ is convex if and only if the scalar function

$$f_t'(t) = \mathbf{z}^T\mathbf{A}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{B}\left(\mathbf{C}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{C}^T\right)^{-1}\mathbf{B}^T(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{A}^T\mathbf{z} \tag{A.6}$$

is convex for any vector $\mathbf{z}$ of appropriate dimensions [12]. Moreover, a function is convex if and only if its epigraph is a convex set, and therefore $f(\mathbf{X})$ is convex, if and only if the set

$$\{s, t | \mathbf{z}^T\mathbf{A}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{B}\left(\mathbf{C}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{C}^T\right)^{-1}\mathbf{B}^T(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{A}^T\mathbf{z} \leq s\}$$

is convex. Employing the properties of Schur complements, we can write

$$\begin{aligned} &\mathbf{z}^T\mathbf{A}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{B}\left(\mathbf{C}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{C}^T\right)^{-1}\mathbf{B}^T(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{A}^T\mathbf{z} \leq s \Leftrightarrow \\ &\begin{bmatrix} \mathbf{C}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{C}^T & \mathbf{B}^T(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{A}^T\mathbf{z} \\ \mathbf{z}^T\mathbf{A}(\mathbf{X}_o + t\mathbf{Z}_o)\mathbf{B} & s \end{bmatrix} \succeq \mathbf{0} \Leftrightarrow \\ \begin{bmatrix} \mathbf{C}\mathbf{X}_o\mathbf{C}^T & \mathbf{B}^T\mathbf{X}_o\mathbf{A}^T\mathbf{z} \\ \mathbf{z}^T\mathbf{A}\mathbf{X}_o\mathbf{B} & 0 \end{bmatrix} + t\begin{bmatrix} \mathbf{C}\mathbf{Z}_o\mathbf{C}^T & \mathbf{B}^T\mathbf{Z}_o\mathbf{A}^T\mathbf{z} \\ \mathbf{z}^T\mathbf{A}\mathbf{Z}_o\mathbf{B} & 0 \end{bmatrix} + s\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} &\succeq \mathbf{0} \end{aligned} \tag{A.7}$$

The last condition defines a convex set in (s,t) [12], and thus the epigraph of $f'_t(t)$ is convex. Consequently, $f_t(t)$ and $f(\mathbf{X})$ are both convex functions, which implies that the right-hand side of (A.3) is a concave function of the matrix

$$\begin{bmatrix} \mathbf{P}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{k+1} \end{bmatrix}$$

■

We now employ this result to prove, by induction, Lemma 2. Assuming that at time-step k the inequality $\bar{\mathbf{P}}_k \succeq E\{\mathbf{P}_k\}$ holds, we will show that it also holds for the time-step $k+1$. We have

$$\begin{aligned}
\mathbf{P}_{k+1} &= \mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T \Rightarrow \\
E\{\mathbf{P}_{k+1}\} &= E\{\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k + \mathbf{G}\mathbf{Q}_{k+1}\mathbf{G}^T\} \\
&= E\{\mathbf{P}_k - \mathbf{P}_k\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R}_{k+1}\right)^{-1}\mathbf{H}\mathbf{P}_k\} + \mathbf{G}E\{\mathbf{Q}_{k+1}\}\mathbf{G}^T \\
&\preceq E\{\mathbf{P}_k\} - E\{\mathbf{P}_k\}\mathbf{H}^T\left(\mathbf{H}E\{\mathbf{P}_k\}\mathbf{H}^T + E\{\mathbf{R}_{k+1}\}\right)^{-1}\mathbf{H}E\{\mathbf{P}_k\} + \mathbf{G}E\{\mathbf{Q}_{k+1}\}\mathbf{G}^T
\end{aligned}$$

where in the last line Jensen's inequality was applied [12], in order to exploit the concavity of the Riccati. By assumption, $\bar{\mathbf{P}}_k \succeq E\{\mathbf{P}_k\}$, and thus employing the monotonicity of the Riccati with respect to the covariance matrix (cf. Appendix A.1), we can write

$$\begin{aligned}
E\{\mathbf{P}_{k+1}\} &\preceq \bar{\mathbf{P}}_k - \bar{\mathbf{P}}_k\mathbf{H}^T\left(\mathbf{H}\bar{\mathbf{P}}_k\mathbf{H}^T + E\{\mathbf{R}_{k+1}\}\right)^{-1}\mathbf{H}\bar{\mathbf{P}}_k + \mathbf{G}E\{\mathbf{Q}_{k+1}\}\mathbf{G}^T \\
&= \bar{\mathbf{P}}_k - \bar{\mathbf{P}}_k\mathbf{H}^T\left(\mathbf{H}\bar{\mathbf{P}}_k\mathbf{H}^T + \bar{\mathbf{R}}\right)^{-1}\mathbf{H}\bar{\mathbf{P}}_k + \mathbf{G}\bar{\mathbf{Q}}\mathbf{G}^T \\
&= \bar{\mathbf{P}}_{k+1}
\end{aligned}$$

Thus, we have proven that $\bar{\mathbf{P}}_k \succeq E\{\mathbf{P}_k\}$ implies $\bar{\mathbf{P}}_{k+1} \succeq E\{\mathbf{P}_{k+1}\}$. For $k=0$ the condition $\bar{\mathbf{P}}_k \succeq E\{\mathbf{P}_k\}$ holds with equality, and the proof is complete.

A.3 The measurement covariance matrix for the position-only EKF

The covariance of the error of the j-th measurement obtained by robot i is given by

$$\begin{aligned} {}^{i}\mathbf{R}_{jj}(k+1) &= \mathbf{\Gamma}_{ij}(k+1)E\{\mathbf{n}_{ij}(k+1)\mathbf{n}_{ij}^{T}(k+1)\}\mathbf{\Gamma}_{ij}^{T}(k+1) \\ &= \mathbf{R}_{z_{ij}}(k+1) + \mathbf{R}_{\tilde{\phi}_{ij}}(k+1) \end{aligned} \tag{A.8}$$

This expression encapsulates all sources of noise and uncertainty that contribute to the measurement error. More specifically, $\mathbf{R}_{z_{ij}}(k+1)$ is the covariance of the noise $\mathbf{n}_{ij}(k+1)$ in the recorded relative position measurement $\mathbf{z}_{ij}(k+1)$ and $\mathbf{R}_{\tilde{\phi}_{ij}}(k+1)$ is the additional covariance term due to the error $\tilde{\phi}_i(k+1)$ in the orientation estimate of the measuring robot. Since these two error sources are uncorrelated, we obtain the decomposition shown in (A.8). The second term in this expression is given by:

$$\begin{aligned} \mathbf{R}_{\tilde{\phi}_{ij}}(k+1) &= \mathbf{C}^{T}(\hat{\phi}_i(k+1))\mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}(k+1)E\{\tilde{\phi}_i^{\,2}\}\widehat{\mathbf{\Delta p}}_{ij}^{T}(k+1)\mathbf{J}^{T}\mathbf{C}(\hat{\phi}_i(k+1)) \\ &= \sigma_{\phi_i}^{2}\mathbf{C}^{T}(\hat{\phi}_i(k+1))\mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}(k+1)\widehat{\mathbf{\Delta p}}_{ij}^{T}(k+1)\mathbf{J}^{T}\mathbf{C}(\hat{\phi}_i(k+1)) \end{aligned} \tag{A.9}$$

The last expression shows that the uncertainty $\sigma_{\phi_i}^2$ in the orientation estimate $\hat{\phi}_i(k+1)$ of the robot is amplified by the distance between the robot and corresponding landmark. We now turn our attention to the term $\mathbf{R}_{z_{ij}}(k+1)$, which expresses the covariance of the errors due to the sensor noise. Each relative position measurement is comprised of the distance ρ_{ij} and bearing θ_{ij} to the target, expressed in the measuring robot's local coordinate frame, i.e.,

$$\mathbf{z}_{ij}(k+1) = \begin{bmatrix} \rho_{ij}(k+1)\cos\theta_{ij}(k+1) \\ \rho_{ij}(k+1)\sin\theta_{ij}(k+1) \end{bmatrix} + \mathbf{n}_{z_{ij}}(k+1)$$

By linearizing, the noise in this measurement can be expressed as:

$$\mathbf{n}_{z_{ij}}(k+1) \simeq \begin{bmatrix} \cos\hat{\theta}_{ij} & -\hat{\rho}_{ij}\sin\hat{\theta}_{ij} \\ \sin\hat{\theta}_{ij} & \hat{\rho}_{ij}\cos\hat{\theta}_{ij} \end{bmatrix} \begin{bmatrix} n_{\rho_{ij}}(k+1) \\ n_{\theta_{ij}}(k+1) \end{bmatrix}$$

where $n_{\rho_{ij}}$ is the error in the range measurement, $n_{\theta_{ij}}$ is the error in the bearing measurement, assumed to be independent white zero-mean Gaussian sequences, and

$$\begin{aligned}\hat{\rho}_{ij} &= \sqrt{\widehat{\mathbf{\Delta p}}_{ij}^T(k+1)\widehat{\mathbf{\Delta p}}_{ij}(k+1)} \\ \hat{\theta}_{ij} &= \text{Atan2}(\widehat{\Delta y}_{ij}(k+1), \widehat{\Delta x}_{ij}(k+1)) - \hat{\phi}_i(k+1)\end{aligned}$$

are the estimates of the range and bearing to the landmark, expressed with respect to the robot's coordinate frame. At this point we note that

$$\begin{aligned}&\mathbf{C}(\hat{\phi}_i(k+1))\mathbf{n}_{z_{ij}}(k+1) \\ &\quad= \begin{bmatrix} \cos\hat{\phi}_i(k+1) & -\sin\hat{\phi}_i(k+1) \\ \sin\hat{\phi}_i(k+1) & \cos\hat{\phi}_i(k+1) \end{bmatrix} \begin{bmatrix} \cos\hat{\theta}_{ij} & -\hat{\rho}_{ij}\sin\hat{\theta}_{ij} \\ \sin\hat{\theta}_{ij} & \hat{\rho}_{ij}\cos\hat{\theta}_{ij} \end{bmatrix} \begin{bmatrix} n_{\rho_{ij}}(k+1) \\ n_{\theta_{ij}}(k+1) \end{bmatrix} \\ &\quad= \begin{bmatrix} \cos(\hat{\phi}_i(k+1)+\hat{\theta}_{ij}) & -\hat{\rho}_{ij}\sin(\hat{\phi}_i(k+1)+\hat{\theta}_{ij}) \\ \sin(\hat{\phi}_i(k+1)+\hat{\theta}_{ij}) & \hat{\rho}_{ij}\cos(\hat{\phi}_i(k+1)+\hat{\theta}_{ij}) \end{bmatrix} \begin{bmatrix} n_{\rho_{ij}}(k+1) \\ n_{\theta_{ij}}(k+1) \end{bmatrix} \\ &\quad= \begin{bmatrix} \frac{1}{\hat{\rho}_{ij}}\widehat{\mathbf{\Delta p}}_{ij} & \mathbf{J}\widehat{\mathbf{\Delta p}}_{ij} \end{bmatrix} \begin{bmatrix} n_{\rho_{ij}}(k+1) \\ n_{\theta_{ij}}(k+1) \end{bmatrix}\end{aligned}$$

and therefore the quantity $\mathbf{R}_{z_{ij}}(k+1)$ can be written as:

$$\mathbf{R}_{z_{ij}}(k+1) = E\{\mathbf{n}_{z_{ij}}(k+1)\mathbf{n}_{z_{ij}}^T(k+1)\} \tag{A.10}$$

$$\begin{aligned}&= \mathbf{C}^T(\hat{\phi}_i(k+1)) \begin{bmatrix} \frac{1}{\hat{\rho}_{ij}}\widehat{\mathbf{\Delta p}}_{ij} & \mathbf{J}\widehat{\mathbf{\Delta p}}_{ij} \end{bmatrix} \begin{bmatrix} \sigma_{\rho_i}^2 & 0 \\ 0 & \sigma_{\theta_i}^2 \end{bmatrix} \begin{bmatrix} \frac{1}{\hat{\rho}_{ij}}\widehat{\mathbf{\Delta p}}_{ij} & \mathbf{J}\widehat{\mathbf{\Delta p}}_{ij} \end{bmatrix}^T \mathbf{C}(\hat{\phi}_i(k+1)) \\ &= \mathbf{C}^T(\hat{\phi}_i(k+1)) \left(\frac{\sigma_{\rho_i}^2}{\hat{\rho}_{ij}^2}\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T + \sigma_{\theta_i}^2\mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T\mathbf{J}^T \right) \mathbf{C}(\hat{\phi}_i(k+1)) \\ &= \mathbf{C}^T(\hat{\phi}_i(k+1)) \left(\frac{\sigma_{\rho_i}^2}{\hat{\rho}_{ij}^2} \left(\hat{\rho}_{ij}^2\mathbf{I}_2 - \mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T\mathbf{J}^T \right) + \sigma_{\theta_i}^2\mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T\mathbf{J}^T \right) \mathbf{C}(\hat{\phi}_i(k+1)) \\ &= \mathbf{C}^T(\hat{\phi}_i(k+1)) \left(\sigma_{\rho_i}^2\mathbf{I}_2 + \left(\sigma_{\theta_i}^2 - \frac{\sigma_{\rho_i}^2}{\hat{\rho}_{ij}^2} \right) \mathbf{J}\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T\mathbf{J}^T \right) \mathbf{C}(\hat{\phi}_i(k+1))\end{aligned} \tag{A.11}$$

where the variance of the noise in the distance and bearing measurements is given by

$$\sigma_{\rho_i}^2 = E\{n_{\rho_i}^2\} \quad , \quad \sigma_{\theta_i}^2 = E\{n_{\theta_i}^2\}$$

respectively. Due to the existence of the error component attributed to $\tilde{\phi}_i(k+1)$, the exteroceptive measurements that each robot performs at a given time instant are correlated. The matrix of correlation between the errors in the measurements $\mathbf{z}_{ij}(k+1)$ and $\mathbf{z}_{i\ell}(k+1)$ is

$$\begin{aligned} {}^{i}\mathbf{R}_{j\ell}(k+1) &= \mathbf{\Gamma}_{ij}(k+1)E\{\mathbf{n}_{ij}(k+1)\mathbf{n}_{i\ell}^{T}(k+1)\}\mathbf{\Gamma}_{i\ell}^{T}(k+1) \\ &= \sigma_{\phi_i}^{2}\mathbf{C}^{T}(\hat{\phi}_i(k+1))\widehat{\mathbf{J\Delta p}}_{ij}(k+1)\widehat{\mathbf{\Delta p}}_{i\ell}^{T}(k+1)\mathbf{J}^{T}\mathbf{C}(\hat{\phi}_i(k+1)) \end{aligned} \tag{A.12}$$

The covariance matrix of all the measurements performed by robot i at the time instant $k+1$ can now be computed. To simplify the presentation, we here assume that $\mathcal{N}_{M_i} = \{1, 2, \ldots, M_i\}$, i.e., the i-th robot measures the robots with indices $j = 1, \ldots, M_i$. In that case, the covariance matrix of all the measurements performed by robot i is a block matrix whose mn-th 2×2 submatrix element is ${}^{i}\mathbf{R}_{mn}$, for $m, n = 1, \ldots, M_i$. Using (A.9), (A.11), and (A.12), this matrix can be written as

$$\boldsymbol{R}_i(k+1) = \mathbf{\Xi}_{\hat{\phi}_i}^{T}(k+1)\boldsymbol{R}_{o_i}(k+1)\mathbf{\Xi}_{\hat{\phi}_i}(k+1) \tag{A.13}$$

where

$$\begin{aligned} \boldsymbol{R}_{o_i}(k+1) &= \sigma_{\rho_i}^{2}\mathbf{I}_{2M_i} + \mathbf{D}_i(k+1)\left(\sigma_{\theta_i}^{2}\mathbf{I}_{M_i} + \sigma_{\phi_i}^{2}\mathbf{1}_{M_i\times M_i} - \operatorname{diag}\left(\frac{\sigma_{\rho_i}^{2}}{\hat{\rho}_{ij}^{2}}\right)\right)\mathbf{D}_i^{T}(k+1) \\ &= \underbrace{\sigma_{\rho_i}^{2}\mathbf{I}_{2M_i} - \mathbf{D}_i(k+1)\operatorname{diag}\left(\frac{\sigma_{\rho_i}^{2}}{\hat{\rho}_{ij}^{2}}\right)\mathbf{D}_i^{T}(k+1)}_{\mathbf{R}_1(k+1)} + \underbrace{\sigma_{\theta_i}^{2}\mathbf{D}_i(k+1)\mathbf{D}_i^{T}(k+1)}_{\mathbf{R}_2(k+1)} \\ &\quad + \underbrace{\sigma_{\phi_i}^{2}\mathbf{D}_i(k+1)\mathbf{1}_{M_i\times M_i}\mathbf{D}_i^{T}(k+1)}_{\mathbf{R}_3(k+1)} \end{aligned} \tag{A.14}$$

where

$$\mathbf{D}_i(k+1) = \begin{bmatrix} \widehat{\mathbf{J\Delta p}}_{i1}(k+1) & \ldots & \mathbf{0}_{2\times 1} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{2\times 1} & \ldots & \widehat{\mathbf{J\Delta p}}_{iM_i}(k+1) \end{bmatrix} = \mathbf{Diag}\left(\widehat{\mathbf{J\Delta p}}_{ij}(k+1)\right)$$

is a $2M_i \times M_i$ block diagonal matrix, depending on the estimated positions of the robots and landmarks. In (A.14) the covariance term $\mathbf{R}_1(k+1)$ is the covariance of the error due

to the noise in the range measurements, $\mathbf{R}_2(k+1)$ is the covariance term due to the error in the bearing measurements, and $\mathbf{R}_3(k+1)$ is the covariance term due to the error in the orientation estimates of the measuring robot.

A.4 Upper bound for the measurement covariance matrix

An upper bound on $\mathbf{R}_o(k+1)$ is obtained by considering each of its block diagonal elements, $\boldsymbol{R}_{o_i}(k+1)$. Referring to (A.14), we examine the terms $\mathbf{R}_1(k+1)$, $\mathbf{R}_2(k+1)$ and $\mathbf{R}_3(k+1)$ separately: the term expressing the effect of the noise in the range measurements is

$$\mathbf{R}_1(k+1) = \sigma_{\rho_i}^2 \mathbf{I}_{2M_i} - \mathbf{D}_i(k+1)\,\mathrm{diag}\left(\frac{\sigma_{\rho_i}^2}{\hat{\rho}_{ij}^2}\right)\mathbf{D}_i^T(k+1) \preceq \sigma_{\rho_i}^2 \mathbf{I}_{2M_i} \tag{A.15}$$

where we have used the fact that the term being subtracted from $\sigma_{\rho_i}^2 \mathbf{I}_{2M_i}$ is a positive semidefinite matrix. The covariance term due to the noise in the bearing measurement is

$$\begin{aligned}
\mathbf{R}_2(k+1) &= \sigma_{\theta_i}^2 \mathbf{D}_i(k+1)\mathbf{D}_i^T(k+1) \\
&= \sigma_{\theta_i}^2 \mathbf{Diag}\left(\hat{\rho}_{ij}^2 \begin{bmatrix} \sin^2(\hat{\theta}_{ij}) & \sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) \\ \sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) & \cos^2(\hat{\theta}_{ij}) \end{bmatrix}\right) \\
&\preceq \sigma_{\theta_i}^2 \mathbf{Diag}\left(\hat{\rho}_{ij}^2 \mathbf{I}_2\right) \\
&\preceq \sigma_{\theta_i}^2 \rho_o^2 \mathbf{I}_{2M_i}
\end{aligned} \tag{A.16}$$

where ρ_o is the maximum range at which a measurement can occur, determined either by the characteristics of the robots' sensors or by the properties of the area in which the robots move. Finally, the covariance term due to the error in the orientation of the measuring robot is $\mathbf{R}_3(k+1) = \sigma_{\phi_i}^2 \mathbf{D}_i(k+1)\mathbf{1}_{M_i \times M_i}\mathbf{D}_i^T(k+1)$. Calculation of the eigenvalues of the matrices $\mathbf{1}_{M_i \times M_i}$ and $\mathbf{I}_{M_i}$ verifies that $\mathbf{1}_{M_i \times M_i} \preceq M_i \mathbf{I}_{M_i}$, and thus we can write $\mathbf{R}_3(k+1) \preceq M_i \sigma_{\phi_i}^2 \mathbf{D}_i(k+1)\mathbf{D}_i^T(k+1)$. By derivations analogous to those employed to yield an upper bound for $\mathbf{R}_2(k+1)$, we can show that

$$\mathbf{R}_3(k+1) \preceq M_i \sigma_{\phi_i}^2 \rho_o^2 \mathbf{I}_{2M_i}$$

By combining this result with those of (A.15), (A.16), we can write $\boldsymbol{R}_{o_i}(k+1) = \mathbf{R}_1(k+1) + \mathbf{R}_2(k+1) + \mathbf{R}_3(k+1) \preceq \mathbf{R}_i^u$, where

$$\mathbf{R}_i^u = \left(\sigma_{\rho_i}^2 + M_i\sigma_{\phi_i}^2\rho_o^2 + \sigma_{\theta_i}^2\rho_o^2\right)\mathbf{I}_{2M_i} = r_i\mathbf{I}_{2M_i} \tag{A.17}$$

A.5 Connection to the Laplacian of the RPMG

In this appendix we present some results concerning the matrix:

$$\mathbf{C}_u = \mathbf{Q}_u^{1/2}\mathbf{H}_o^T\mathbf{R}_u^{-1}\mathbf{H}_o\mathbf{Q}_u^{1/2}$$

which are useful in computing the steady-state solution to the Riccati recursion.

- **Connection to the RPMG Laplacian**

We first examine the relationship of the matrix $\mathbf{C}_u$ to the Laplacian matrix of the RPMG [97]. As explained in Section 2.4.2, the matrix $\mathbf{H}_o$ is closely related to the incidence matrix of the RPMG, $\mathbf{A}_{\text{RPMG}}$:

$$\mathbf{H}_o = \mathbf{A}_{\text{RPMG}} \otimes \mathbf{I}_2 \tag{A.18}$$

In obtaining this expression, the RPMG is considered an unweighted graph. If, instead, we consider the *weighted* RPMG, where the weight of the edge corresponding to the measurement of robot j from robot i is $\sqrt{r_i}$, we obtain the incidence matrix:

$$\mathbf{A}'_{\text{RPMG}} = \mathbf{A}_{\text{RPMG}}\mathbf{R}_u^{-1/2} \tag{A.19}$$

and the Laplacian matrix of this weighted RPMG is:

$$\mathcal{L} = \mathbf{A}_{\text{RPMG}}^T\mathbf{R}_u^{-1}\mathbf{A}_{\text{RPMG}} \tag{A.20}$$

Using this observation, we can write the matrix $\mathbf{C}_u$ as

$$\mathbf{C}_u = \mathbf{Q}_u^{1/2}\left(\mathcal{L} \otimes \mathbf{I}_2\right)\mathbf{Q}_u^{1/2} \tag{A.21}$$

This result shows the close connection between $\mathbf{C}_u$ and the Laplacian matrix of the RPMG, and will be helpful in determining important properties of the matrix $\mathbf{C}_u$.

• **Rank and eigenvalues of $\mathbf{C}_u$: unobservable system**

It is well-known that the Laplacian matrix of a graph has a nullspace of dimension equal to the number of connected components of the graph [97]. In our analysis we are considering the case of a connected graph, and thus $\text{rank}(\mathcal{L}) = N-1$. Consequently, $\text{rank}(\mathbf{C}_u) = 2N-2$, since $\mathbf{Q}_u$ is a full-rank matrix, and from the properties of the Kronecker product, we obtain $\text{rank}\,(\mathcal{L} \otimes \mathbf{I}_2) = 2\,\text{rank}(\mathcal{L})$. In order to obtain a basis for the nullspace of $\mathbf{C}_u$, we note that the nullspace of $\mathcal{L}$ is spanned by the vector $\mathbf{1}_{N\times 1}$, and therefore $\mathcal{L}\mathbf{1}_{N\times 1} = \mathbf{0}_{N\times 1}$. We now note that

$$\begin{aligned}\mathbf{C}_u \left(\mathbf{Q}_u^{-1/2}\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2\right) &= \mathbf{Q}_u^{1/2}(\mathcal{L} \otimes \mathbf{I}_2)\mathbf{Q}_u^{1/2} \left(\mathbf{Q}_u^{-1/2}\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2\right) \\ &= \mathbf{Q}_u^{1/2}(\mathcal{L} \otimes \mathbf{I}_2)\,(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\end{aligned}$$

Employing the properties of the Kronecker product yields

$$(\mathcal{L} \otimes \mathbf{I}_2)\,(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2) = (\mathcal{L}\mathbf{1}_{N\times 1}) \otimes \mathbf{I}_2 = \mathbf{0}_{2N\times 2}$$

and therefore

$$\mathbf{C}_u \left(\mathbf{Q}_u^{-1/2}\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2\right) = \mathbf{0}_{2N\times 2}$$

which shows that the column vectors of the matrix $\mathbf{Q}_u^{-1/2}\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2$ span the nullspace of $\mathbf{C}_u$. These column vectors are:

$$\mathbf{c}_1 = \mathbf{Q}_u^{-1/2} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix}$$

and

$$\mathbf{c}_1 = \mathbf{Q}_u^{-1/2} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ \vdots \end{bmatrix}$$

which are orthogonal (this is easily verified by computing the dot product $\mathbf{c}_1^T \mathbf{c}_2$). Normalizing the above vectors to ensure unit length, yields an orthonormal basis for the nullspace of $\mathbf{C}_u$. It can be easily shown that the resulting basis vectors are the columns of the matrix:

$$\mathbf{V} = \sqrt{q_T} \mathbf{Q}_u^{-1/2} \left(\mathbf{1}_{N \times 1} \otimes \mathbf{I}_2 \right) \tag{A.22}$$

where q_T is defined by the relation

$$\frac{1}{q_T} = \sum_{i=1}^{N} \frac{1}{q_i}$$

- **Rank and eigenvalues of $\mathbf{C}_u$: observable system**

When at least one of the robots has access to absolute position measurements, then the RPMG becomes a *grounded* graph. In this case, the graph Laplacian (and therefore $\mathbf{C}_u$) are full-rank matrices [95].

A.6 Asymptotic covariance bound for CL

In this appendix, we show how to derive the steady-state solution of the Riccati recursion in (2.51). We denote the SVD of matrix $\mathbf{C}_u$ as $\mathbf{C}_u = \mathbf{U} \operatorname{diag}(\lambda_i) \mathbf{U}^T = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T$, and defining $\mathbf{P}_{nn_k} = \mathbf{U}^T \mathbf{P}_{n_k} \mathbf{U}$ yields the recursion

$$\mathbf{P}_{nn_{k+1}} = \mathbf{P}_{nn_k} \left(\mathbf{I}_{2N} + \mathbf{\Lambda} \mathbf{P}_{nn_k} \right)^{-1} + \mathbf{I}_{2N} \tag{A.23}$$

When the system is observable, at steady state we have $\mathbf{P}_{nn_{k+1}} = \mathbf{P}_{nn_k} = \mathbf{P}_{nn_\infty}$, and we thus need to solve the matrix equation

$$\mathbf{P}_{nn_\infty} = \mathbf{P}_{nn_\infty} \left(\mathbf{I}_{2N} + \mathbf{\Lambda} \mathbf{P}_{nn_\infty} \right)^{-1} + \mathbf{I}_{2N}$$

In this expression, all the diagonal elements of $\mathbf{\Lambda}$ are positive, since $\mathbf{C}_u$ is a positive definite matrix. Assuming a diagonal form for $\mathbf{P}_{nn_\infty}$, it is straightforward to derive the solution

$$\mathbf{P}_{nn_\infty} = \text{diag}\left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{1}{\lambda_i}}\right)$$

However, the fact that we are dealing with an observable system, means that the asymptotic solution to the Riccati recursion is unique [58]. Thus, the above derived solution is unique, and from it, the expression in (2.52) follows directly from the relation

$$\mathbf{P}_k^u = \mathbf{Q}_u^{1/2}\mathbf{P}_{n_k}\mathbf{Q}_u^{1/2} = \mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{P}_{nn_k}\mathbf{U}^T\mathbf{Q}_u^{1/2} \tag{A.24}$$

When none of the robots has access to absolute position measurements, the system is unobservable, and the asymptotic solution to the Riccati recursion in (2.51) (or, equivalently, to (A.23)) is not as straightforward, since the solution now depends on the initial value of the covariance matrix. As shown in Appendix A.5, when the robots only record relative position measurements, $\mathbf{C}_u$ is of rank $2N-2$, and therefore two of its singular values equal zero.

We first address the case in which the initial covariance matrix is zero. We observe that the right-hand side of (A.23) is a diagonal matrix in this case, and by a simple induction argument, we can show that the solution to this recursion retains a diagonal form for all $k \geq 0$. Addressing each of the diagonal elements individually, we observe that for the first $2N-2$ elements, which correspond to the nonzero singular values, we obtain the equations

$$\mathbf{P}_{nn_{k+1}}(i,i) = \mathbf{P}_{nn_k}(i,i)\left(1 + \lambda_i \mathbf{P}_{nn_k}(i,i)\right)^{-1} + 1$$

while for the last two elements we obtain

$$\mathbf{P}_{nn_{k+1}}(i,i) = \mathbf{P}_{nn_k}(i,i) + 1$$

Therefore, the asymptotic solution for $\mathbf{P}_{nn_k}$ is given by

$$\mathbf{P}_{nn_k} = \begin{bmatrix} \mathrm{diag}_\xi\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & k\mathbf{I}_2 \end{bmatrix} \tag{A.25}$$

$$= k\underbrace{\begin{bmatrix} \mathbf{0}_{\xi\times\xi} & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{I}_2 \end{bmatrix}}_{\mathbf{D}_1} + \underbrace{\begin{bmatrix} \mathrm{diag}_\xi\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix}}_{\mathbf{D}_2} \tag{A.26}$$

where $\xi = 2N-2$. Using this result, we can compute the upper bound on the position uncertainty when the initial covariance matrix is equal to zero. To compute the solution when the initial uncertainty is not zero, we introduce the matrix $\breve{\mathbf{P}}_k$, which is defined as follows:

$$\breve{\mathbf{P}}_k = \mathbf{P}_{nn_k} - k\mathbf{D}_1 \Rightarrow \mathbf{P}_{nn_k} = \breve{\mathbf{P}}_k + k\mathbf{D}_1 \tag{A.27}$$

The matrix $\breve{\mathbf{P}}_k$ has the property that it asymptotically approaches a constant value, which depends on the initial covariance matrix. In order to obtain a Riccati recursion for the time evolution of $\breve{\mathbf{P}}_k$, we substitute the above definition in (A.23):

$$\begin{aligned}
\breve{\mathbf{P}}_{k+1} + (k+1)\mathbf{D}_1 &= \left(\breve{\mathbf{P}}_k + k\mathbf{D}_1\right)\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k + k\mathbf{\Lambda}\mathbf{D}_1\right)^{-1} + \mathbf{I}_{2N} \Rightarrow \\
\breve{\mathbf{P}}_{k+1} + (k+1)\mathbf{D}_1 &= \left(\breve{\mathbf{P}}_k + k\mathbf{D}_1\right)\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + \mathbf{I}_{2N} \Rightarrow \\
\breve{\mathbf{P}}_k + (k+1)\mathbf{D}_1 &= \breve{\mathbf{P}}_k\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + k\mathbf{D}_1\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + \mathbf{I}_{2N}
\end{aligned}$$

Where we have used the fact that, since the 2 smallest eigenvalues of $\mathbf{C}_u$ equal zero, the relation $\mathbf{\Lambda}\mathbf{D}_1 = \mathbf{0}_{2N\times 2N}$ holds. By application of the matrix inversion lemma (cf. Appendix D) in the second term of the right-hand side of the above expression, we obtain

$$\begin{aligned}
\breve{\mathbf{P}}_k + (k+1)\mathbf{D}_1 &= \breve{\mathbf{P}}_k\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + k\mathbf{D}_1\left(\mathbf{I}_{2N} - \mathbf{\Lambda}\left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_k\mathbf{\Lambda}\right)^{-1}\breve{\mathbf{P}}_k\right) + \mathbf{I}_{2N} \Rightarrow \\
\breve{\mathbf{P}}_{k+1} + (k+1)\mathbf{D}_1 &= \breve{\mathbf{P}}_k\left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + k\mathbf{D}_1 + \mathbf{I}_{2N}
\end{aligned}$$

where the result $\mathbf{\Lambda}\mathbf{D}_1 = \mathbf{0}_{2N\times 2N}$ has been employed once more. Finally, from the last

expression we obtain

$$\breve{\mathbf{P}}_{k+1} = \breve{\mathbf{P}}_k \left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + \mathbf{D}_3 \tag{A.28}$$

where

$$\mathbf{D}_3 = \mathbf{I}_{2N} - \mathbf{D}_1 = \begin{bmatrix} \mathbf{I}_{\xi\times\xi} & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix}$$

The solution of this recursion is derived employing the following lemma (adapted from [58]):

Lemma 12 *Suppose* $\breve{\mathbf{P}}_k^{(0)}$ *is the solution to the Riccati recursion*

$$\begin{aligned} \breve{\mathbf{P}}_{k+1} &= \breve{\mathbf{P}}_k \left(\mathbf{I}_{2N} + \mathbf{\Lambda}\breve{\mathbf{P}}_k\right)^{-1} + \mathbf{D}_3 & \text{(A.29)} \\ &= \breve{\mathbf{P}}_k + \breve{\mathbf{P}}_k\sqrt{\mathbf{\Lambda}} \left(\mathbf{I}_{2N} + \sqrt{\mathbf{\Lambda}}\breve{\mathbf{P}}_k\sqrt{\mathbf{\Lambda}}\right)^{-1} \sqrt{\mathbf{\Lambda}}\breve{\mathbf{P}}_k + \mathbf{D}_3 & \text{(A.30)} \end{aligned}$$

with zero initial condition. Then the solution to this recursion when the initial covariance matrix is an arbitrary positive semidefinite matrix $\breve{\mathbf{P}}_0$*, is defined by the relation*

$$\breve{\mathbf{P}}_{k+1} - \breve{\mathbf{P}}_{k+1}^{(0)} = \mathbf{\Psi}_p^{(0)}(k+1,0) \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_0\mathbf{J}_{k+1}\right)^{-1} \breve{\mathbf{P}}_0 \mathbf{\Psi}_p^{(0)}(k+1,0)^T \tag{A.31}$$

where

$$\mathbf{\Psi}_p^{(0)}(k+1,0) = \left(\mathbf{I}_{2N} - \mathbf{P}\sqrt{\mathbf{\Lambda}} \left(\mathbf{I}_{2N} + \sqrt{\mathbf{\Lambda}}\mathbf{P}\sqrt{\mathbf{\Lambda}}\right)^{-1} \sqrt{\mathbf{\Lambda}}\right)^{k+1} (\mathbf{I}_{2N} + \mathbf{P}\mathbf{J}_{k+1}) \tag{A.32}$$

In these expressions $\mathbf{P}$ *is any solution to the discrete algebraic Riccati equation (DARE)*

$$\mathbf{P} = \mathbf{P} - \mathbf{P}\sqrt{\mathbf{\Lambda}} \left(\mathbf{I}_{2N} + \sqrt{\mathbf{\Lambda}}\mathbf{P}\sqrt{\mathbf{\Lambda}}\right)^{-1} \sqrt{\mathbf{\Lambda}}\mathbf{P} + \mathbf{D}_3 \tag{A.33}$$

and $\mathbf{J}_k$ *denotes the solution to the dual Riccati recursion with zero initial condition:*

$$\mathbf{J}_{k+1} = \mathbf{J}_k + \mathbf{\Lambda} - \mathbf{J}_k \left(\mathbf{I}_{2N} + \mathbf{J}_k\right)^{-1} \mathbf{J}_k, \qquad \mathbf{J}_0 = \mathbf{0}_{2N\times 2N} \tag{A.34}$$

We now apply this lemma to derive the steady-state value of $\breve{\mathbf{P}}_k$, when the initial

covariance or the robots' position estimates is an arbitrary positive semidefinite matrix $\mathbf{P}_0$, in which case we have

$$\breve{\mathbf{P}}_0 \;=\; \mathbf{P}_{nn_0} - 0 \cdot \mathbf{D}_1 \;=\; \mathbf{U}^T \mathbf{P}_{n_0} \mathbf{U} \;=\; \mathbf{U}^T \mathbf{Q}_u^{-1/2} \mathbf{P}_0 \mathbf{Q}_u^{-1/2} \mathbf{U} \tag{A.35}$$

We first note that the steady-state solution to the recursion in (A.30) with zero initial condition can be directly derived by the definition of $\breve{\mathbf{P}}_k$ in (A.27) and the steady-state solution for $\mathbf{P}_{nn}$ with zero in initial condition, given in (A.26). We obtain:

$$\breve{\mathbf{P}}_{ss}^{(0)}(k) \;=\; \mathbf{D}_2 \tag{A.36}$$

Substitution of $\breve{\mathbf{P}}_{ss}^{(0)}$ for $\mathbf{P}$ in (A.33) verifies that $\breve{\mathbf{P}}_{ss}^{(0)}$ is a solution of the DARE, and therefore:

$$\begin{aligned}
\mathbf{\Psi}_p^{(0)}(k+1,0) &= \left(\mathbf{I}_{2N} - \breve{\mathbf{P}}_{ss}^{(0)} \sqrt{\mathbf{\Lambda}} \left(\mathbf{I}_{2N} + \sqrt{\mathbf{\Lambda}} \breve{\mathbf{P}}_{ss}^{(0)} \sqrt{\mathbf{\Lambda}}\right)^{-1} \sqrt{\mathbf{\Lambda}}\right)^{k+1} \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_{ss}^{(0)} \mathbf{J}_{k+1}\right) \\
&= \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_{ss}^{(0)} \mathbf{\Lambda}\right)^{-(k+1)} \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_{ss}^{(0)} \mathbf{J}_{k+1}\right) \\
&= \begin{bmatrix} \mathrm{diag}_\xi \left(1 + \frac{\lambda_i}{2} + \lambda_i \sqrt{\frac{1}{4} + \frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi \times 2} \\ \mathbf{0}_{2 \times \xi} & \mathbf{I}_2 \end{bmatrix}^{-(k+1)} \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_{ss}^{(0)} \mathbf{J}_{k+1}\right)
\end{aligned} \tag{A.37}$$

where we have applied the matrix inversion lemma to simplify the expression. To obtain the asymptotic value of this expression as $k \to \infty$, we note that

$$\left(1 + \frac{\lambda_i}{2} + \lambda_i \sqrt{\frac{1}{4} + \frac{1}{\lambda_i}}\right) > 1 \Rightarrow \lim_{k \to \infty} \left(1 + \frac{\lambda_i}{2} + \lambda_i \sqrt{\frac{1}{4} + \frac{1}{\lambda_i}}\right)^{-(k+1)} = 0$$

Therefore we obtain:

$$\lim_{k \to \infty} \mathbf{\Psi}_p^{(0)}(k+1,0) = \mathbf{D}_1 \left(\mathbf{I}_{2N} + \breve{\mathbf{P}}_{ss}^{(0)} \mathbf{J}_{ss}\right) \tag{A.38}$$

where $\mathbf{J}_{ss}$ is the steady-state solution of the dual Riccati recursion (A.34). To compute this solution, we observe that, since the initial condition of this recursion is zero, at $k = 0$ the right-hand side of (A.34) is a diagonal matrix. By induction, it is simple to show that $\mathbf{J}_k$ will retain its diagonal structure for all $k \geq 0$, and therefore the solution to the recursion

is obtained by solving a set of independent scalar recursions, for the diagonal elements $\mathbf{J}_k(i,i), i = 1 \ldots 2N$. These recursions are given by

$$\mathbf{J}_{k+1}(i,i) \;=\; \mathbf{J}_k(i,i) + \lambda_i - \frac{\mathbf{J}_k(i,i)^2}{1+\mathbf{J}_k(i,i)}, \qquad i = 1 \ldots 2N-2 \tag{A.39}$$

while the elements $\mathbf{J}_k(2N-1,2N-1)$ and $\mathbf{J}_k(2N,2N)$ remain equal to zero for all time. By evaluating the steady state solution of these recursions (i.e., by requiring that $\mathbf{J}_{k+1}(i,i) = \mathbf{J}_k(i,i)$, and solving the resulting equations) we obtain the following solution for $\mathbf{J}_k$ at steady state:

$$\mathbf{J}_{ss} \;=\; \begin{bmatrix} \mathrm{diag}_\xi\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4}+\lambda_i}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix} \tag{A.40}$$

Substituting this expression as well as the result of (A.36) in (A.38), we obtain:

$$\lim_{k\to\infty} \mathbf{\Psi}_p^{(0)}(k+1,0) = \mathbf{D}_1$$

Using this result in (A.31) yields

$$\begin{aligned} \lim_{k\to\infty}\left(\breve{\mathbf{P}}_{k+1} - \breve{\mathbf{P}}_{k+1}^{(0)}\right) &= \lim_{k\to\infty}\left(\mathbf{\Psi}_p^{(0)}(k+1,0)\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{k+1}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{\Psi}_p^{(0)}(k+1,0)^T\right) \\ &= \mathbf{D}_1\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{ss}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{D}_1 \end{aligned}$$

and therefore

$$\breve{\mathbf{P}}_{ss} = \lim_{k\to\infty}\breve{\mathbf{P}}_{k+1} = \breve{\mathbf{P}}_{ss}^{(0)} + \mathbf{D}_1\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{ss}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{D}_1$$

This result allows us to evaluate the asymptotic covariance bound for the case when none of the robots has access to absolute position information. Using (A.27), we obtain

$$\mathbf{P}_{nn_{ss}} = k\mathbf{D}_1 + \mathbf{D}_2 + \mathbf{D}_1\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{ss}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{D}_1$$

Using the relation $\mathbf{P}_k^u = \mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{P}_{nn_k}\mathbf{U}^T\mathbf{Q}_u^{1/2}$, we can now compute each of the three terms of (2.53). Specifically, the first term equals $k\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\mathbf{U}^T\mathbf{Q}_u^{1/2}$, the second term equals

$\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_2\mathbf{U}^T\mathbf{Q}_u^{1/2}$, while the third one equals $\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{ss}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{D}_1\mathbf{U}^T\mathbf{Q}_u^{1/2}$. To prove these results, we note that the matrix $\mathbf{U}$ can be partitioned as:

$$\mathbf{U}=[\mathbf{S}\ \ \mathbf{V}] \tag{A.41}$$

where $\mathbf{S}$ is a $2N\times(2N-2)$ matrix whose columns span the range of $\mathbf{C}_u$, while $\mathbf{V}$ is the $2N\times 2$ matrix whose columns span the nullspace of $\mathbf{C}_u$ (cf. (A.22)). Using this result, we see that

$$\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\mathbf{U}^T\mathbf{Q}_u^{1/2}=\mathbf{Q}_u^{1/2}\mathbf{V}\mathbf{V}^T\mathbf{Q}_u^{1/2}$$

Substituting from (A.22) in the last expression yields:

$$\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\mathbf{U}^T\mathbf{Q}_u^{1/2}=q_T\mathbf{1}_{N\times N}\otimes\mathbf{I}_2 \tag{A.42}$$

which provides us with the first term of (2.53). The derivation of the second term is straightforward, by substituting for $\mathbf{D}_2$, while for the third term, which expresses the effect of the initial uncertainty, we obtain:

$$\begin{aligned}\mathbf{P}_{\text{init}}&=\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\left(\mathbf{I}_{2N}+\breve{\mathbf{P}}_0\mathbf{J}_{ss}\right)^{-1}\breve{\mathbf{P}}_0\mathbf{D}_1^T\mathbf{U}^T\mathbf{Q}_u^{1/2}\\&=\mathbf{Q}_u^{1/2}\mathbf{U}\mathbf{D}_1\mathbf{U}^T\mathbf{U}\left(\mathbf{I}_{2N}+\mathbf{U}^T\mathbf{Q}_u^{-1/2}\mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\mathbf{J}_{ss}\right)^{-1}\mathbf{U}^T\mathbf{Q}_u^{-1/2}\mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\mathbf{D}_1^T\mathbf{U}^T\mathbf{Q}_u^{1/2}\end{aligned}$$

where we have used (A.35). Employing the result of (A.42) in the above expression yields

$$\begin{aligned}\mathbf{P}_{\text{init}}&=q_T^2\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\mathbf{Q}_u^{-1/2}\left(\mathbf{I}_{2N}+\mathbf{Q}_u^{-1/2}\mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\mathbf{J}_{ss}\mathbf{U}^T\right)^{-1}\\&\qquad\times\mathbf{Q}_u^{-1/2}\mathbf{P}_0\mathbf{Q}_u^{-1}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right) &\text{(A.43)}\\&=q_T^2\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\mathbf{Q}_u^{-1/2}\left(\mathbf{Q}_u^{1/2}+\mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\mathbf{J}_{ss}\mathbf{U}^T\right)^{-1}\mathbf{P}_0\mathbf{Q}_u^{-1}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\\&=q_T^2\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\mathbf{Q}_u^{-1}\left(\mathbf{I}_{2N}+\mathbf{P}_0\mathbf{Q}_u^{-1/2}\mathbf{U}\mathbf{J}_{ss}\mathbf{U}^T\mathbf{Q}_u^{-1/2}\right)^{-1}\mathbf{P}_0\mathbf{Q}_u^{-1}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right) &\text{(A.44)}\end{aligned}$$

At this point, we note that the term $\mathbf{U}\mathbf{J}_{ss}\mathbf{U}^T$ appearing in the last expression equals

(cf. (A.40)):

$$\mathbf{U}\mathbf{J}_{ss}\mathbf{U}^T = \text{diag}\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4} + \lambda_i}\right) \tag{A.45}$$

Moreover, from the properties of the Kronecker product we obtain

$$\mathbf{1}_{N\times N} \otimes \mathbf{I}_2 = (\mathbf{1}_{N\times 1}\mathbf{1}_{1\times N}) \otimes \mathbf{I}_2 = (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\,(\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)$$

Additionally, we note that if $\mathbf{W}$ is a $2N \times 2N$ matrix, then

$$\begin{aligned}
(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\,\mathbf{W}\,(\mathbf{1}_{1\times N} \otimes \mathbf{I}_2) &= (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\,(\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\,\mathbf{W}\,(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\,(\mathbf{1}_{1\times N} \otimes \mathbf{I}_2) \\
&= (\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\Big((\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\,\mathbf{W}\,(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\Big)\,(\mathbf{1}_{1\times N} \otimes \mathbf{I}_2) \\
&= (\mathbf{1}_{N\times N} \otimes \mathbf{I}_2) \otimes \Big((\mathbf{1}_{1\times N} \otimes \mathbf{I}_2)\,\mathbf{W}\,(\mathbf{1}_{N\times 1} \otimes \mathbf{I}_2)\Big)
\end{aligned}$$

Using the last result, in conjunction with (A.44) and (A.45), the third term in (2.53) can be directly obtained.

A.7 Average value of the measurement covariance matrix

From (A.14) we note that evaluation of the average value of $\boldsymbol{R}_{o_i}(k+1)$ requires the computation of the expected values of the following terms (omitting time indices for succinctness):

$$\mathbf{T}_1 = \frac{\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}^T}{\hat{\rho}_{ij}^2}, \quad \mathbf{T}_2 = \widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}^T, \quad \text{and} \quad \mathbf{T}_3 = \widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{i\ell}^T \tag{A.46}$$

for $j, \ell = 1 \ldots M_i$. The average value of $\mathbf{T}_1$ is easily derived by employing the polar coordinate description of the vector $\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}$ in terms of $\hat{\rho}_{ij}$ and $\hat{\theta}_{ij}$, which yields

$$\begin{aligned}
\mathbf{T}_1 &= \frac{\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}\widehat{\boldsymbol{\Delta}\mathbf{p}}_{ij}^T}{\hat{\rho}_{ij}^2} \\
&= \frac{1}{\hat{\rho}_{ij}^2}\begin{bmatrix} \hat{\rho}_{ij}^2\cos^2(\hat{\theta}_{ij}) & \hat{\rho}_{ij}^2\sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) \\ \hat{\rho}_{ij}^2\sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) & \hat{\rho}_{ij}^2\sin^2(\hat{\theta}_{ij}) \end{bmatrix}
\end{aligned}$$

$$= \begin{bmatrix} \cos^2(\hat{\theta}_{ij}) & \sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) \\ \sin(\hat{\theta}_{ij})\cos(\hat{\theta}_{ij}) & \sin^2(\hat{\theta}_{ij}) \end{bmatrix}$$

From the last expression we conclude that for any probability density function that guarantees a uniform distribution for the bearing angle of the measurements (i.e., any symmetric probability density function), the average value of the term $\mathbf{T}_1$ is

$$E\{\mathbf{T}_1\} = \frac{1}{2}\mathbf{I}_2$$

In order to compute the expected value of the terms $\mathbf{T}_2$ and $\mathbf{T}_3$, we assume that the robots and landmarks are located in a square arena of side a, and that their positions are described by uniformly distributed random variables in the interval $[-a/2, a/2]$. We can thus write

$$\begin{aligned}
E\{\mathbf{T}_2\} &= E\{\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{ij}^T\} \\
&= E\left\{ \begin{bmatrix} \widehat{\Delta x}_{ij}^2 & \widehat{\Delta x}_{ij}\widehat{\Delta y}_{ij} \\ \widehat{\Delta y}_{ij}\widehat{\Delta x}_{ij} & \widehat{\Delta y}_{ij}^2 \end{bmatrix} \right\} \\
&= \begin{bmatrix} E\{x_j^2 - 2x_ix_j + x_i^2\} & E\{x_jy_j - x_jy_i - x_iy_j + x_iy_i\} \\ E\{y_jx_j - y_jx_i - y_ix_j + y_ix_i\} & E\{y_j^2 - 2y_jy_i + y_i^2\} \end{bmatrix} \\
&= \begin{bmatrix} 2E\{x_i^2\} & 0 \\ 0 & 2E\{y_i^2\} \end{bmatrix} \\
&= \begin{bmatrix} \frac{a^2}{6} & 0 \\ 0 & \frac{a^2}{6} \end{bmatrix} \\
&= \frac{a^2}{6}\mathbf{I}_2
\end{aligned}$$

and similarly,

$$\begin{aligned}
E\{\mathbf{T}_3\} &= E\{\widehat{\mathbf{\Delta p}}_{ij}\widehat{\mathbf{\Delta p}}_{i\ell}^T\} \\
&= E\left\{ \begin{bmatrix} \widehat{\Delta x}_{ij}\widehat{\Delta x}_{i\ell} & \widehat{\Delta x}_{ij}\widehat{\Delta y}_{i\ell} \\ \widehat{\Delta y}_{ij}\widehat{\Delta x}_{i\ell} & \widehat{\Delta y}_{ij}\widehat{\Delta y}_{i\ell} \end{bmatrix} \right\}
\end{aligned}$$

$$
\begin{aligned}
&= \begin{bmatrix} E\{x_j x_\ell - x_i x_\ell - x_j x_i + x_i^2\} & E\{x_j y_\ell - x_j y_i - x_i y_\ell + x_i y_i\} \\ E\{y_j x_\ell - y_j x_i - y_i x_\ell + y_i x_i\} & E\{y_j y_\ell - y_i y_\ell - y_j y_i + y_i^2\} \end{bmatrix} \\
&= \begin{bmatrix} E\{x_i^2\} & 0 \\ 0 & E\{y_i^2\} \end{bmatrix} \\
&= \begin{bmatrix} \frac{a^2}{12} & 0 \\ 0 & \frac{a^2}{12} \end{bmatrix} \\
&= \frac{a^2}{12}\mathbf{I}_2
\end{aligned}
$$

These results enable us to obtain the average value of the matrices $\boldsymbol{R}_{o_i}(k)$, $i = 1 \ldots N$. Employing the linearity of the expectation operator yields

$$
\begin{aligned}
\bar{\mathbf{R}}_i &= E\{\boldsymbol{R}_{o_i}(k)\} \\
&= \begin{bmatrix} \left(\frac{1}{2}\sigma_{\rho_i}^2 + \frac{a^2}{6}\sigma_{\phi_i}^2 + \frac{a^2}{6}\sigma_{\theta_i}^2\right)\mathbf{I}_2 & \cdots & \frac{a^2}{12}\sigma_{\phi_i}^2\mathbf{I}_2 \\ \vdots & \ddots & \vdots \\ \frac{a^2}{12}\sigma_{\phi_i}^2\mathbf{I}_2 & \cdots & \left(\frac{1}{2}\sigma_{\rho_i}^2 + \frac{a^2}{6}\sigma_{\phi_i}^2 + \frac{a^2}{6}\sigma_{\theta_i}^2\right)\mathbf{I}_2 \end{bmatrix} \\
&= \left(\frac{1}{2}\sigma_{\rho_i}^2 + \frac{a^2}{12}\sigma_{\phi_i}^2 + \frac{a^2}{6}\sigma_{\theta_i}^2\right)\mathbf{I}_{2M_i} + \frac{a^2}{12}\sigma_{\phi_i}^2\left(\mathbf{1}_{M_i \times M_i} \otimes \mathbf{I}_2\right)
\end{aligned}
$$

A.8 RPMG reconfigurations

Our goal is to show that, once the RPMG resumes its initial topology after a series of reconfigurations, the upper bound of (2.53) will be the same, as if no reconfigurations had occurred. We here prove this result for the case where the RPMG is initially in topology $\mathcal{T}_A$, then at time-step t_1 assumes topology $\mathcal{T}_B$, and finally at time-step t_2 it switches back to its initial topology. The case of more general series of topology changes can be treated similarly.

We first express the result of (2.53) with respect to the normalized covariance matrix $\mathbf{P}_{n_k} = \mathbf{Q}_u^{-1/2}\mathbf{P}_k^u\mathbf{Q}_u^{-1/2}$. In particular, we have

$$
\mathbf{P}_{n_{ss}}(k) = \mathbf{Q}_u^{-1/2}\mathbf{P}_{ss}^u(k)\mathbf{Q}_u^{-1/2}
$$

$$= kq_T\mathbf{Q}_u^{-1/2}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\mathbf{Q}_u^{-1/2} + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T \quad \text{(A.47)}$$

$$+ q_T^2\mathbf{Q}_u^{-1/2}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\otimes\begin{bmatrix}\alpha & \beta\\ \beta & \delta\end{bmatrix}\mathbf{Q}_u^{-1/2}$$

$$= k\mathbf{V}\mathbf{V}^T + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T$$

$$+ q_T^2\mathbf{Q}_u^{-1/2}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\otimes\begin{bmatrix}\alpha & \beta\\ \beta & \delta\end{bmatrix}\mathbf{Q}_u^{-1/2} \quad \text{(A.48)}$$

where $\xi = 2N-2$. Moreover, the third term in (2.53), which expresses the effect of the initial uncertainty, can be written as (cf. (A.43)):

$$\begin{aligned}\mathbf{P}_{\text{init}} &= q_T^2\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\otimes\begin{bmatrix}\alpha & \beta\\ \beta & \delta\end{bmatrix}\\ &= q_T^2\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\mathbf{Q}_u^{-1/2}\left(\mathbf{I}_{2N}+\mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{Q}_u^{-1/2}\left(\mathbf{1}_{N\times N}\otimes\mathbf{I}_2\right)\\ &= \mathbf{Q}_u^{1/2}\mathbf{V}\mathbf{V}^T\left(\mathbf{I}_{2N}+\mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\mathbf{V}^T\mathbf{Q}_u^{1/2}\end{aligned}$$

where

$$h(\mathbf{C}_u) = \mathbf{U}\,\mathrm{diag}\left(\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i}\right)\mathbf{U}^T = \mathbf{U}\,\mathrm{diag}(h(\lambda_i))\mathbf{U}^T \quad \text{(A.49)}$$

Therefore (A.48) is equivalently written as

$$\begin{aligned}\mathbf{P}_{n_{ss}}(k) &= k\mathbf{V}\mathbf{V}^T + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T\\ &+ \mathbf{V}\mathbf{V}^T\left(\mathbf{I}_{2N}+\mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\mathbf{V}^T \quad \text{(A.50)}\end{aligned}$$

Introducing the notation

$$f(\lambda_i) = \frac{1}{2}+\sqrt{\frac{1}{4}+\frac{1}{\lambda_i}}$$

further simplifies the preceding expression, to yield:

$$\begin{aligned}
\mathbf{P}_{n_{ss}}(k) &= k\mathbf{V}\mathbf{V}^T + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(f(\lambda_i)\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T + \mathbf{V}\mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\mathbf{V}^T \\
&= \mathbf{V}\left(k\mathbf{I}_2 + \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\right)\mathbf{V}^T \qquad \text{(A.51)} \\
&\quad + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(f(\lambda_i)\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T \\
&= \mathbf{U}\begin{bmatrix} \mathbf{0}_{\xi\times\xi} & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & k\mathbf{I}_2 + \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\end{bmatrix}\mathbf{U}^T \qquad \text{(A.52)} \\
&\quad + \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(f(\lambda_i)\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2}\end{bmatrix}\mathbf{U}^T \\
&= \mathbf{U}\begin{bmatrix} \mathrm{diag}_\xi\left(f(\lambda_i)\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & k\mathbf{I}_2 + \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_0}h(\mathbf{C}_u)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V}\end{bmatrix}\mathbf{U}^T \qquad \text{(A.53)}
\end{aligned}$$

where we have employed the partitioning of $\mathbf{U}$ shown in (A.41), in the transition from (A.51) to (A.52).

Assuming that the RPMG remains in the topology $\mathcal{T}_A$ for the time-step interval $[0, t_1]$, and that this interval is of sufficient duration for the covariance to reach steady state, then at time-step t_1 the normalized covariance matrix is given by

$$\mathbf{P}_{n_{ss}}(t_1) \;=\; \mathbf{U}_A\begin{bmatrix} \mathrm{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0})\end{bmatrix}\mathbf{U}_A^T \qquad \text{(A.54)}$$

where

$$\mathbf{M}_A(\mathbf{P}_{n_0}) = \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_0}h(\mathbf{C}_A)\right)^{-1}\mathbf{P}_{n_0}\mathbf{V} \qquad \text{(A.55)}$$

In these expressions the quantities that depend on the RPMG topology $\mathcal{T}_A$ have been denoted by the subscript A. It is important to note that the basis vectors of the nullspace of the matrix $\mathbf{C}_u$ are *independent* of the topology of the RPMG and thus no subscript is needed to distinguish them. This essentially is a consequence of the fact that the unobservable

subspace remains the same, regardless of the topology of the RPMG.

The proof proceeds in three steps: first, we prove the desired result for the case where the topology $\mathcal{T}_B$ is a connected one. Then, we carry out the proof for the case where during the second phase, all the robots localize using DR only (i.e., $\mathcal{T}_B$ is an empty graph). Finally, we prove the result for the general case, where $\mathcal{T}_B$ is any arbitrary graph topology. These three cases are described next

Connected RPMG topology during $[t_1, t_2]$

When the RPMG $\mathcal{T}_B$ during the time interval $[t_1, t_2]$ is a connected one, we can apply the result of (A.53) to obtain the normalized covariance at time-step t_2. Assuming that steady state is reached, we obtain:

$$\mathbf{P}_{n_{ss}}(t_2) \;=\; \mathbf{U}_B \begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{B_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & (t_2-t_1)\mathbf{I}_2 + \mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right) \end{bmatrix} \mathbf{U}_B^T \tag{A.56}$$

where

$$\mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right) = \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_{ss}}(t_1)h(\mathbf{C}_B)\right)^{-1}\mathbf{P}_{n_{ss}}(t_1)\mathbf{V}$$

We now derive a simpler expression for $\mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right)$. We start by applying the matrix inversion lemma (cf. Appendix D), to obtain

$$\begin{aligned} \mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right) &= \mathbf{V}^T\left(\mathbf{I}_{2N} + \mathbf{P}_{n_{ss}}(t_1)h(\mathbf{C}_B)\right)^{-1}\mathbf{P}_{n_{ss}}(t_1)\mathbf{V} && \text{(A.57)} \\ &= \mathbf{V}^T\mathbf{P}_{n_{ss}}(t_1)\mathbf{V} - \mathbf{V}^T\mathbf{P}_{n_{ss}}(t_1)\left(\mathbf{I}_{2N} + h(\mathbf{C}_B)\mathbf{P}_{n_{ss}}(t_1)\right)^{-1}h(\mathbf{C}_B)\mathbf{P}_{n_{ss}}(t_1)\mathbf{V} \\ &= \mathbf{V}^T\mathbf{P}_{n_{ss}}(t_1)\mathbf{V} - \mathbf{V}^T\mathbf{P}_{n_{ss}}(t_1)\left(\mathbf{I}_{2N} + h(\mathbf{C}_B)\mathbf{P}_{n_{ss}}(t_1)\right)^{-1}\mathbf{Z} && \text{(A.58)} \end{aligned}$$

Next, we study the matrix product $\mathbf{Z} = h(\mathbf{C}_B)\mathbf{P}_{n_{ss}}(t_1)\mathbf{V}$, appearing in the last equation. We have

$$\begin{aligned} \mathbf{Z} &= \mathbf{U}_B\,\mathrm{diag}\left(h(\lambda_{B_i})\right)\mathbf{U}_B^T\mathbf{U}_A \begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_A^T\mathbf{V} \\ &= \mathbf{U}_B\,\mathrm{diag}\left(h(\lambda_{B_i})\right)\begin{bmatrix}\mathbf{S}_B^T \\ \mathbf{V}^T\end{bmatrix}\begin{bmatrix}\mathbf{S}_A & \mathbf{V}\end{bmatrix}\begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix}\begin{bmatrix}\mathbf{S}_A^T \\ \mathbf{V}^T\end{bmatrix}\mathbf{V} \end{aligned}$$

$$= \mathbf{U}_B \operatorname{diag}\left(h(\lambda_{B_i})\right) \begin{bmatrix} \mathbf{S}_B^T\mathbf{S}_A & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \begin{bmatrix} \mathbf{0}_{\xi\times 2} \\ \mathbf{I}_2 \end{bmatrix}$$

In the last line we have used the fact that $\mathbf{S}_A^T\mathbf{V} = \mathbf{S}_B^T\mathbf{V} = \mathbf{0}_{\xi\times 2}$, which results from the orthogonality of the columns of the matrices $\mathbf{U}_A$ and $\mathbf{U}_B$. We also note that the last two diagonal elements of $\operatorname{diag}\left(h(\lambda_{B_i})\right)$ (i.e., the ones that correspond to the zero eigenvalues) are equal to zero (cf. (A.49)). Thus, $\mathbf{Z}$ can be written as

$$\begin{aligned}
\mathbf{Z} &= \mathbf{U}_B \begin{bmatrix} \operatorname{diag}_\xi\left(h(\lambda_{B_i})\right)\mathbf{S}_B^T\mathbf{S}_A & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix} \begin{bmatrix} \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \begin{bmatrix} \mathbf{S}_A^T \\ \mathbf{V}^T \end{bmatrix} \mathbf{V} \\
&= \mathbf{U}_B \begin{bmatrix} \operatorname{diag}_\xi\left(h(\lambda_{B_i})\right)\mathbf{S}_B^T\mathbf{S}_A \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{\xi\times 2} \\ \mathbf{I}_2 \end{bmatrix} \\
&= \mathbf{U}_B \begin{bmatrix} \operatorname{diag}_\xi\left(h(\lambda_{B_i})\right)\mathbf{S}_B^T\mathbf{S}_A \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & \mathbf{0}_{2\times 2} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{\xi\times 2} \\ \mathbf{I}_2 \end{bmatrix} \\
&= \mathbf{0}_{\xi\times 2}
\end{aligned}$$

This result means that the second term in (A.58)vanishes, and thus $\mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right)$ can be written as

$$\begin{aligned}
\mathbf{M}_B\left(\mathbf{P}_{n_{ss}}(t_1)\right) &= \mathbf{V}^T\mathbf{P}_{n_{ss}}(t_1)\mathbf{V} \\
&= \mathbf{V}^T\mathbf{U}_A \begin{bmatrix} \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_A^T\mathbf{V} \\
&= \mathbf{V}^T \begin{bmatrix} \mathbf{S}_A & \mathbf{V} \end{bmatrix} \begin{bmatrix} \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \begin{bmatrix} \mathbf{S}_A^T \\ \mathbf{V}^T \end{bmatrix} \mathbf{V} \\
&= \begin{bmatrix} \mathbf{0}_{2\times\xi} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \operatorname{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \begin{bmatrix} \mathbf{0}_{\xi\times 2} \\ \mathbf{I}_2 \end{bmatrix} \\
&= t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \qquad \text{(A.59)}
\end{aligned}$$

Substitution of this result in (A.56) yields

$$\mathbf{P}_{n_{ss}}(t_2) \;=\; \mathbf{U}_B \begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{B_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_B^T \tag{A.60}$$

From the last expression, we conclude that the term describing the effect of the initial uncertainty is *the same* for both topologies $\mathcal{T}_A$ and $\mathcal{T}_B$, and equals $\mathbf{M}_A(\mathbf{P}_{n_0})$.

If at time-step t_2 the RPMG assumes its initial topology $\mathcal{T}_A$ once again, then at some time $t_3 > t_2$, once steady-state is reached, the normalized covariance is given by

$$\mathbf{P}_{n_{ss}}(t_3) \;=\; \mathbf{U}_A \begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & (t_3 - t_2)\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_{ss}}(t_2)) \end{bmatrix} \mathbf{U}_A^T \tag{A.61}$$

By derivations completely analogous to those of (A.57)-(A.59), we can prove that

$$\mathbf{M}_A(\mathbf{P}_{n_{ss}}(t_2)) = t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \tag{A.62}$$

and therefore:

$$\mathbf{P}_{n_{ss}}(t_3) \;=\; \mathbf{U}_A \begin{bmatrix} \mathrm{diag}_{\xi}\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_3\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_A^T \tag{A.63}$$

This result is the same one that would result from use of (A.53) if no reconfigurations had occurred. We have thus proved the lemma for the case where the intermediate topology $\mathcal{T}_B$ is a connected one.

Empty RPMG topology during $[t_1, t_2]$

If during the intermediate phase the robots localize based only on odometry, then during this time interval their covariance bounds are propagated by

$$\mathbf{P}^u_{k+1} = \mathbf{P}^u_k + \mathbf{Q}_u$$

or, expressed using the normalized covariance,

$$\mathbf{P}_{n_{k+1}} = \mathbf{P}_{n_k} + \mathbf{I}_{2N}$$

Thus, at time-step t_2 we would have

$$\begin{aligned}
\mathbf{P}_{n_{\text{od}}}(t_2) &= \mathbf{P}_{n_{ss}}(t_1) + (t_2 - t_1)\mathbf{I}_{2N} \\
&= \mathbf{U}_A \begin{bmatrix} \text{diag}_\xi\left(f(\lambda_{A_i})\right) & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_1\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_A^T + (t_2 - t_1)\mathbf{U}_A\mathbf{U}_A^T \\
&= \mathbf{U}_A \begin{bmatrix} \text{diag}_\xi\left(f(\lambda_{A_i})\right) + (t_2 - t_1)\mathbf{I}_\xi & \mathbf{0}_{\xi\times 2} \\ \mathbf{0}_{2\times\xi} & t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \end{bmatrix} \mathbf{U}_A^T
\end{aligned} \tag{A.64}$$

By comparison of this result with that of (A.60), we observe that the basic structure of the covariance matrix is the same. Thus, by following steps analogous to those of (A.57)-(A.59) we can show that

$$\mathbf{M}_A(\mathbf{P}_{n_{\text{od}}}(t_2)) = t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \tag{A.65}$$

and therefore (A.63) holds without change.

Arbitrary RPMG topology during $[t_1, t_2]$

Up to this point, we have proven the lemma for the situations where the intermediate topology, $\mathcal{T}_B$, is either a connected or an empty graph. To show that the lemma also holds for any other possible topology $\mathcal{T}_B$ (for example, the case in which only some of the robots localize using odometry, while a subgroup of robots can still record relative position measurements) we study the Riccati recursion that describes the normalized covariance during the intermediate phase. This recursion can be written as:

$$\mathbf{P}_{n_{k+1}} = \left(\mathbf{P}_{n_k}^{-1} + \mathbf{C}_B\right)^{-1} + \mathbf{I}_{2N} \tag{A.66}$$

where $\mathbf{C}_B$ is given by

$$\mathbf{C}_B = \mathbf{Q}_u^{1/2}\mathbf{H}_{B_o}^T\mathbf{R}_{B_u}^{-1}\mathbf{H}_{B_o}\mathbf{Q}_u^{1/2}$$

We now consider a graph topology $\mathcal{T}'_B$, which is *connected*, and contains the RPMG of $\mathcal{T}_B$ as a subgraph. For the topology $\mathcal{T}'_B$, we can show that

$$\mathbf{H}_{B_o}^T \mathbf{R}_{B_u}^{-1} \mathbf{H}_{B_o} \prec \mathbf{H}_{B_o}'^T \mathbf{R}_{B_u}'^{-1} \mathbf{H}'_{B_o}$$

and thus

$$\begin{aligned} \mathbf{Q}_u^{1/2} \mathbf{H}_{B_o}^T \mathbf{R}_{B_u}^{-1} \mathbf{H}_{B_o} \mathbf{Q}_u^{1/2} \prec \mathbf{Q}_u^{1/2} \mathbf{H}_{B_o}'^T \mathbf{R}_{B_u}'^{-1} \mathbf{H}'_{B_o} \mathbf{Q}_u^{1/2} \Rightarrow \\ \mathbf{C}_B \prec \mathbf{C}'_B \end{aligned} \tag{A.67}$$

Using this result, and employing the monotonicity properties of the Riccati in (A.66), it can be shown that the normalized covariance corresponding to $\mathcal{T}_B$ and $\mathcal{T}'_B$ satisfy:

$$\mathbf{P}_{n_{ss}}(t_2) \succeq \mathbf{P}'_{n_{ss}}(t_2) \tag{A.68}$$

Similarly, using the monotonicity properties of the Riccati we obtain

$$\mathbf{P}_{n_{ss}}(t_2) \preceq \mathbf{P}_{n_{\text{od}}}(t_2)$$

Combining the last two results yields:

$$\mathbf{P}'_{n_{ss}}(t_2) \preceq \mathbf{P}_{n_{ss}}(t_2) \preceq \mathbf{P}_{n_{\text{od}}}(t_2) \tag{A.69}$$

In Appendix A.9 it is shown that the matrix

$$\mathbf{M}_A(\mathbf{X}) = \mathbf{V}^T \left(\mathbf{I}_{2N} + \mathbf{X} h(\mathbf{C}_A)\right)^{-1} \mathbf{X} \mathbf{V}$$

is a matrix-increasing function of $\mathbf{X}$. Employing this result, in conjunction with (A.69), we obtain

$$\mathbf{M}_A(\mathbf{P}'_{n_{ss}}(t_2)) \preceq \mathbf{M}_A(\mathbf{P}_{n_{ss}}(t_2)) \preceq \mathbf{M}_A(\mathbf{P}_{n_{\text{od}}}(t_2))$$

However, since $\mathcal{T}'_B$ is a connected topology, the results of the first part of this appendix hold, and thus

$$\mathbf{M}_A(\mathbf{P}'_{n_{ss}}(t_2)) = t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0})$$

Combining this result with that of (A.65) we conclude that

$$\mathbf{M}_A(\mathbf{P}_{n_{ss}}(t_2)) = \mathbf{M}_A(\mathbf{P}_{n_{\text{od}}}(t_2)) = \mathbf{M}_A(\mathbf{P}'_{n_{ss}}(t_2)) = t_2\mathbf{I}_2 + \mathbf{M}_A(\mathbf{P}_{n_0}) \tag{A.70}$$

This implies that for any possible topology, the value of $\mathbf{M}_A$ will be identical to the one derived for the case of a connected graph and for the case of Dead Reckoning, and the proof is complete.

A.9 Monotonicity of $\mathbf{M}_A$

In this appendix we show that the matrix

$$\mathbf{M}_A \;=\; \mathbf{V}^T\mathbf{X}\left(\mathbf{I}_{2N} + h(\mathbf{C}_u)\mathbf{X}\right)^{-1}\mathbf{V} \tag{A.71}$$

is matrix-increasing in the argument $\mathbf{X}$, i.e.,

$$\mathbf{X}' \succeq \mathbf{X} \Rightarrow \mathbf{M}_A(\mathbf{X}') \succeq \mathbf{M}_A(\mathbf{X}) \tag{A.72}$$

We note that if $\mathbf{X}$ is invertible (which is the case of interest), then

$$\mathbf{M}_A \;=\; \mathbf{V}^T\left(\mathbf{X}^{-1} + h(\mathbf{C}_u)\right)^{-1}\mathbf{V} \tag{A.73}$$

And from the last relation it follows that

$$\begin{aligned}
\mathbf{X}' &\succeq \mathbf{X} \Rightarrow \\
\mathbf{X}'^{-1} &\preceq \mathbf{X}^{-1} \Rightarrow \\
\mathbf{X}'^{-1} + h(\mathbf{C}_u) &\preceq \mathbf{X}^{-1} + h(\mathbf{C}_u) \Rightarrow \\
\left(\mathbf{X}'^{-1} + h(\mathbf{C}_u)\right)^{-1} &\succeq \left(\mathbf{X}^{-1} + h(\mathbf{C}_u)\right)^{-1} \Rightarrow
\end{aligned}$$

$$\begin{aligned}\mathbf{V}^T\left(\mathbf{X}'^{-1}+h(\mathbf{C}_u)\right)^{-1}\mathbf{V} &\succeq \mathbf{V}^T\left(\mathbf{X}^{-1}+h(\mathbf{C}_u)\right)^{-1}\mathbf{V}\Rightarrow\\ \mathbf{M}_A(\mathbf{X}') &\succeq \mathbf{M}_A(\mathbf{X})\end{aligned}$$

A.10 Proof of Lemma 10

First, we note that the properties in (2.113) are equivalent to the expression:

$$\mathbf{P}_k^u = \mathbf{C}_k\mathbf{P}'_k\mathbf{C}_k^T \tag{A.74}$$

where

$$\mathbf{C}_k = \mathbf{\Phi}_k\cdot\mathbf{\Phi}_{k-1}\cdots\mathbf{\Phi}_0$$

We will prove the above property by induction. Let us assume that this property holds at time-step ℓ, i.e., that:

$$\mathbf{P}_\ell^u = \mathbf{C}_\ell\mathbf{P}'_\ell\mathbf{C}_\ell^T \tag{A.75}$$

From the Riccati recursion we obtain:

$$\begin{aligned}\mathbf{P}_{\ell+1}^u &= \mathbf{\Phi}_{\ell+1}\left(\mathbf{P}_\ell^u-\mathbf{P}_\ell^u\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_\ell^u\mathbf{H}^T+\mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_\ell^u\right)\mathbf{\Phi}_{\ell+1}^T+\mathbf{G}\mathbf{Q}_u\mathbf{G}^T\\ &= \mathbf{\Phi}_{\ell+1}\left(\mathbf{P}_\ell^u-\mathbf{P}_\ell^u\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_\ell^u\mathbf{H}^T+\mathbf{R}_u\right)^{-1}\mathbf{H}\mathbf{P}_\ell^u+\mathbf{G}\mathbf{Q}_u\mathbf{G}^T\right)\mathbf{\Phi}_{\ell+1}^T \qquad (A.76)\end{aligned}$$

In the last expression, we have employed the property (cf. (2.85), and (2.107)):

$$\mathbf{G}\mathbf{Q}_u\mathbf{G}^T = \begin{bmatrix}q_1(\mathbf{1}_{M\times M}\otimes\mathbf{I}_2)+q_2\mathbf{I}_{2M} & \mathbf{0}_{2M\times 2M}\\ \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M}\end{bmatrix} \tag{A.77}$$

$$= \mathbf{\Phi}_{\ell+1}\begin{bmatrix}q_1(\mathbf{1}_{M\times M}\otimes\mathbf{I}_2)+q_2\mathbf{I}_{2M} & \mathbf{0}_{2M\times 2M}\\ \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M}\end{bmatrix}\mathbf{\Phi}_{\ell+1}^T \tag{A.78}$$

$$= \mathbf{\Phi}_{\ell+1}\mathbf{G}\mathbf{Q}_u\mathbf{G}^T\mathbf{\Phi}_{\ell+1}^T \tag{A.79}$$

Substitution from (A.75) into (A.76), yields:

$$\begin{aligned}\mathbf{P}_{\ell+1}^{u} &= \mathbf{\Phi}_{\ell+1}\mathbf{C}_{\ell}\left(\mathbf{P}_{\ell}' - \mathbf{P}_{\ell}'\mathbf{C}_{\ell}^{T}\mathbf{H}^{T}\left(\mathbf{H}\mathbf{C}_{\ell}\mathbf{P}_{\ell}'\mathbf{C}_{\ell}^{T}\mathbf{H}^{T} + \mathbf{R}_{u}\right)^{-1}\mathbf{H}\mathbf{C}_{\ell}\mathbf{P}_{\ell}' + \mathbf{G}\mathbf{Q}_{u}\mathbf{G}^{T}\right)\mathbf{C}_{\ell}^{T}\mathbf{\Phi}_{\ell+1}^{T} \\ &= \mathbf{C}_{\ell+1}\left(\mathbf{P}_{\ell}' - \mathbf{P}_{\ell}'\mathbf{C}_{\ell}^{T}\mathbf{H}^{T}\left(\mathbf{H}\mathbf{C}_{\ell}\mathbf{P}_{\ell}'\mathbf{C}_{\ell}^{T}\mathbf{H}^{T} + \mathbf{R}_{u}\right)^{-1}\mathbf{H}\mathbf{C}_{\ell}\mathbf{P}_{\ell}' + \mathbf{G}\mathbf{Q}_{u}\mathbf{G}^{T}\right)\mathbf{C}_{\ell+1}^{T} \quad \text{(A.80)}\end{aligned}$$

At this point, we employ the following relations, which can be easily verified:

$$\mathbf{C}_{\ell}^{T}\mathbf{H}^{T} = \mathbf{H}^{T}\mathbf{C}_{\ell}^{T} \quad \text{(A.81)}$$

$$\mathbf{R}_{u} = \mathbf{C}_{\ell}\mathbf{R}_{u}\mathbf{C}_{\ell}^{T} \quad \text{(A.82)}$$

Substitution in (A.80) yields

$$\begin{aligned}\mathbf{P}_{\ell+1}^{u} &= \mathbf{C}_{\ell+1}\left(\mathbf{P}_{\ell}' - \mathbf{P}_{\ell}'\mathbf{H}^{T}\left(\mathbf{H}\mathbf{P}_{\ell}'\mathbf{H}^{T} + \mathbf{R}_{u}\right)^{-1}\mathbf{H}\mathbf{P}_{\ell}' + \mathbf{G}\mathbf{Q}_{u}\mathbf{G}^{T}\right)\mathbf{C}_{\ell+1}^{T} \\ &= \mathbf{C}_{\ell+1}\mathbf{P}_{\ell+1}'\mathbf{C}_{\ell+1}^{T} \quad \text{(A.83)}\end{aligned}$$

We have thus shown that if the property of (A.75) holds for time index ℓ, it then also holds for time index $\ell+1$. For $\ell = 0$, the property can be easily shown to hold, since $\mathbf{C}_0 = \mathbf{I}_{2M}$. Thus, the proof by induction is complete.

A.11 Steady-state solution of the Riccati recursion for the dual map

To determine the asymptotic solution of (2.111), we employ the following lemma, which has been adapted from [58]:

Lemma 13 *Suppose $\mathbf{P}_k'^{(0)}$ is the solution to the discrete-time Riccati recursion in (2.111) with initial value $\mathbf{P}_0' = \mathbf{0}_{4M\times 4M}$. Then the solution with the initial condition given in (2.108) is determined by the identity*

$$\mathbf{P}_k' - \mathbf{P}_k'^{(0)} = \mathbf{T}_k\left(\mathbf{I}_{4M} + \mathbf{P}_0'\mathbf{J}_k\right)^{-1}\mathbf{P}_0'\mathbf{T}_k^{T} \quad \text{(A.84)}$$

where $\mathbf{T}_k$ *is given by*

$$\mathbf{T}_k = (\mathbf{I}_{4M} - \mathbf{K}_p\mathbf{H})^k (\mathbf{I}_{4M} + \mathbf{P}\mathbf{J}_k) \tag{A.85}$$

In these expressions, $\mathbf{P}$ *is any solution to the Discrete Algebraic Riccati Equation (DARE):*

$$\mathbf{P} = \mathbf{P} - \mathbf{P}\mathbf{H}^T(\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}_u)^{-1}\mathbf{H}\mathbf{P} + \mathbf{G}\mathbf{Q}_u\mathbf{G}^T \tag{A.86}$$

and $\mathbf{K}_p = \mathbf{P}\mathbf{H}^T \left(\mathbf{R}_u + \mathbf{H}\mathbf{P}\mathbf{H}^T\right)^{-1}$. $\mathbf{J}_k$ *denotes the solution to the* dual *Riccati recursion:*

$$\mathbf{J}_{k+1} = \mathbf{J}_k - \mathbf{J}_k\mathbf{G}(\mathbf{Q}_u^{-1} + \mathbf{G}^T\mathbf{J}_k\mathbf{G})^{-1}\mathbf{G}^T\mathbf{J}_k + \mathbf{H}^T\mathbf{R}_u^{-1}\mathbf{H} \tag{A.87}$$

with zero initial condition, $\mathbf{J}_0 = \mathbf{0}_{4M\times 4M}$.

The derivations comprise three intermediate results:

Intermediate Result 1

We first derive the solution to (2.111) with zero initial condition. When the initial covariance for the map coordinates is zero, the submatrix of $\mathbf{P}'_k$ corresponding to the global map will remain equal to zero for all k, since no uncertainty is ever added to the static landmarks' positions. Therefore, the only submatrix of $\mathbf{P}'_k$ with nonzero value is the submatrix ${}^R\mathbf{P}'_k$, which corresponds to the relative map. To simplify the derivations, we introduce the eigenvalue decomposition of the matrix $\mathbf{Q}_u$, which we denote as

$$\mathbf{Q}_u = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T = \mathbf{U}\,\mathrm{diag}(\lambda_i)\mathbf{U}^T$$

Substitution of the values of the matrices $\mathbf{H}$, $\mathbf{G}$, $\mathbf{R}_u$ and $\mathbf{Q}_u$ in (2.111), leads to the following recursion for ${}^R\mathbf{P}'_k$:

$$\begin{aligned}
{}^R\mathbf{P}'_{k+1} &= {}^R\mathbf{P}'_k - {}^R\mathbf{P}'_k \left({}^R\mathbf{P}'_k + r\mathbf{I}_{2M}\right)^{-1} {}^R\mathbf{P}'_k + \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \Rightarrow & \text{(A.88)}\\
\mathbf{U}^T{}^R\mathbf{P}'_k\mathbf{U} &= \mathbf{U}^T{}^R\mathbf{P}'_k\mathbf{U} - \mathbf{U}^T{}^R\mathbf{P}'_k\mathbf{U}\left(\mathbf{U}^T{}^R\mathbf{P}'_k\mathbf{U} + r\mathbf{I}_{2M}\right)^{-1}\mathbf{U}^T{}^R\mathbf{P}'_k\mathbf{U} + \mathbf{\Lambda} \Rightarrow & \text{(A.89)}\\
\mathbf{P}'_{n_{k+1}} &= \mathbf{P}'_{n_k} - \mathbf{P}'_{n_k}\left(\mathbf{P}'_{n_k} + r\mathbf{I}_{2M}\right)^{-1}\mathbf{P}'_{n_k} + \mathbf{\Lambda} & \text{(A.90)}
\end{aligned}$$

where we have denoted

$$\mathbf{P}'_{n_k} = \mathbf{U}^{TR}\mathbf{P}'_k\mathbf{U} \tag{A.91}$$

We note that since $\mathbf{P}'_{n_k}$ is initially zero, and the matrix coefficients in the above recursion are diagonal, $\mathbf{P}'_{n_k}$ will retain a diagonal structure for all time. The steady-state value of $\mathbf{P}'_{n_k}$, which we denote as $\mathbf{P}'_{n_\infty} = \text{diag}(p_{\infty_i})$, is found by solving the equations:

$$p_{\infty_i} \quad = \quad p_{\infty_i} - \frac{p_{\infty_i}^2}{p_{\infty_i} + r} + \lambda_i, i = 1, ..., 2M \tag{A.92}$$

Solving these equations and substituting in (A.91), we obtain the following steady-state solution for ${}^R\mathbf{P}'_k$:

$${}^R\mathbf{P}'_\infty = \mathbf{U}\,\text{diag}\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4} + \lambda_i r}\right)\mathbf{U}^T \tag{A.93}$$

and therefore the steady state solution to the Riccati in (2.111) with zero initial condition is given by

$$\mathbf{P}'^{(0)}_\infty = \begin{bmatrix} \mathbf{U}\,\text{diag}\left(\frac{\lambda_i}{2} + \sqrt{\frac{\lambda_i^2}{4} + \lambda_i r}\right)\mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \end{bmatrix} \tag{A.94}$$

Intermediate Result 2

We next derive the steady-state solution to the dual Riccati in (A.87). Substituting the values of the matrices $\mathbf{H}$, $\mathbf{G}$, $\mathbf{R}_u$ and $\mathbf{Q}_u$ in this recursion, and studying the block structure of the matrices that appear in it, leads to the observation that all block submatrices of $\mathbf{J}_k$, except for the one corresponding to the relative map, remain zero. The time evolution of this submatrix is described by the recursion:

$$\begin{aligned} {}^R\mathbf{J}_{k+1} &= {}^R\mathbf{J}_k + \frac{1}{r}\mathbf{I}_{2M} - {}^R\mathbf{J}_k\left({}^R\mathbf{J}_k + \mathbf{U}\,\text{diag}\left(\frac{1}{\lambda_i}\right)\mathbf{U}^T\right)^{-1}{}^R\mathbf{J}_k \Rightarrow \\ \mathbf{U}^{TR}\mathbf{J}_{k+1}\mathbf{U} &= \mathbf{U}^{TR}\mathbf{J}_k\mathbf{U} + \frac{1}{r}\mathbf{I}_{2M} - \mathbf{U}^{TR}\mathbf{J}_k\mathbf{U}\left(\mathbf{U}^{TR}\mathbf{J}_k\mathbf{U} + \text{diag}\left(\frac{1}{\lambda_i}\right)\right)^{-1}\mathbf{U}^{TR}\mathbf{J}_k\mathbf{U} \Rightarrow \end{aligned}$$

$$\mathbf{J}_{n_{k+1}} = \mathbf{J}_{n_k} + \frac{1}{r}\mathbf{I}_{2M} - \mathbf{J}_{n_k}\left(\mathbf{J}_{n_k} + \operatorname{diag}\left(\frac{1}{\lambda_i}\right)\right)^{-1}\mathbf{J}_{n_k}$$

where we have defined

$$\mathbf{J}_{n_k} = \mathbf{U}^{TR}\mathbf{J}_k\mathbf{U} \tag{A.95}$$

Similarly to the case of $\mathbf{P}'_{n_k}$, we observe that $\mathbf{J}_{n_k}$ remains diagonal for all time. Its asymptotic value is found by setting $\mathbf{J}_{n_k} = \mathbf{J}_{n_{k+1}} = \mathbf{J}_{n_\infty}$, and is equal to

$$\mathbf{J}_{n_\infty} = \operatorname{diag}\left(\frac{1}{2r} + \sqrt{\frac{1}{4r^2} + \frac{1}{\lambda_i r}}\right)$$

Therefore, the steady-state value of ${}^R\mathbf{J}_k$ is

$${}^R\mathbf{J}_\infty = \mathbf{U}\operatorname{diag}\left(\frac{1}{2r} + \sqrt{\frac{1}{4r^2} + \frac{1}{\lambda_i r}}\right)\mathbf{U}^T = \mathbf{U}\operatorname{diag}(J_{n_\infty})\mathbf{U}^T \tag{A.96}$$

and the asymptotic solution of the dual Riccati with zero initial condition is given by

$$\mathbf{J}_\infty = \begin{bmatrix} {}^R\mathbf{J}_\infty & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \end{bmatrix} = \begin{bmatrix} \mathbf{U}\operatorname{diag}\left(\frac{1}{2r} + \sqrt{\frac{1}{4r^2} + \frac{1}{\lambda_i r}}\right)\mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \end{bmatrix} \tag{A.97}$$

Intermediate Result 3

The solution requires computation of the asymptotic value of the right-hand side of (A.84). For this purpose, we now compute the asymptotic value of the matrix $\mathbf{T}_k$ (cf. (A.85)). We first note that $\mathbf{P}'^{(0)}_\infty$ is a solution to the DARE in (A.86) (this can be verified by substitution), and thus

$$\begin{aligned}
\mathbf{T}_k &= (\mathbf{I}_{4M} - \mathbf{K}_p\mathbf{H})^k(\mathbf{I}_{4M} + \mathbf{P}\mathbf{J}_k) \\
&= \left(\mathbf{I}_{4M} - \mathbf{P}'^{(0)}_\infty\mathbf{H}^T\left(\mathbf{R}_u + \mathbf{H}\mathbf{P}'^{(0)}_\infty\mathbf{H}^T\right)^{-1}\mathbf{H}\right)^k\left(\mathbf{I}_{4M} + \mathbf{P}'^{(0)}_\infty\mathbf{J}_k\right) \\
&= \begin{bmatrix} \left(\mathbf{I}_{2M} - {}^R\mathbf{P}'^{(0)}_\infty\left(r\mathbf{I}_{2M} + {}^R\mathbf{P}'^{(0)}_\infty\right)^{-1}\right)^k & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix}\left(\mathbf{I}_{4M} + \mathbf{P}'^{(0)}_\infty\mathbf{J}_k\right)
\end{aligned}$$

$$= \begin{bmatrix} \mathbf{U}\,\mathrm{diag}\left(1-\frac{p_\infty}{r+p_\infty}\right)^k \mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \left(\mathbf{I}_{4M} + \mathbf{P}_\infty'^{(0)}\mathbf{J}_k\right)$$

At this point we note that

$$1 - \frac{p_\infty}{r+p_\infty} < 1$$

and thus

$$\lim_{k\to\infty}\left(1-\frac{p_\infty}{r+p_\infty}\right)^k = 0$$

Therefore, we obtain

$$\lim_{k\to\infty}\mathbf{T}_k = \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \tag{A.98}$$

Steady-state solution

To compute the steady-state solution of (2.111), we evaluate the right-hand side of (A.84) as $k\to\infty$. Substitution from (2.108), (A.97) and (A.98) yields:

$$\begin{aligned}
&\mathbf{P}'_\infty - \mathbf{P}'^{(0)}_\infty = \mathbf{T}_\infty\left(\mathbf{I}_{4M}+\mathbf{P}'_0\mathbf{J}_\infty\right)^{-1}\mathbf{P}'_0\mathbf{T}_\infty^T \\
&= \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{2M}+(\mathbf{Q}_u+r\mathbf{I}_{2M})^R\mathbf{J}_\infty & \mathbf{0}_{2M\times 2M} \\ r^R\mathbf{J}_\infty\mathbf{I}_{2M} & \mathbf{I}_{2M} \end{bmatrix}^{-1} \\
&\times \begin{bmatrix} (\mathbf{Q}_u+r\mathbf{I}_{2M}) & r\mathbf{I}_{2M} \\ r\mathbf{I}_{2M} & r\mathbf{I}_{2M} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \begin{bmatrix} \mathbf{U}\,\mathrm{diag}\left(1+(\lambda_i+r)J_{\infty_i}\right)\mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ -\mathbf{U}\,\mathrm{diag}\left(rJ_{\infty_i}\right)\mathbf{U}^T & \mathbf{I}_{2M} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0}_{2M\times 2M} & r\mathbf{I}_{2M} \\ \mathbf{0}_{2M\times 2M} & r\mathbf{I}_{2M} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{I}_{2M} \end{bmatrix} \begin{bmatrix} \mathbf{U}\,\mathrm{diag}\left(\frac{1}{1+(\lambda_i+r)J_{\infty_i}}\right)\mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ -\mathbf{U}\,\mathrm{diag}\left(\frac{rJ_{\infty_i}}{1+(\lambda_i+r)J_{\infty_i}}\right)\mathbf{U}^T & \mathbf{I}_{2M} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{2M\times 2M} & r\mathbf{I}_{2M} \\ \mathbf{0}_{2M\times 2M} & r\mathbf{I}_{2M} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{U}\,\mathrm{diag}\left(r-\frac{r^2J_{\infty_i}}{1+(\lambda_i+r)J_{\infty_i}}\right)\mathbf{U}^T \end{bmatrix}
\end{aligned} \tag{A.99}$$

Substitution for the values of J_{∞_i} from (A.96) in the last expression, and simple algebraic manipulation, yields

$$\mathbf{P}'_\infty - \mathbf{P}'^{(0)}_\infty = \begin{bmatrix} \mathbf{0}_{2M\times 2M} & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{U}\,\mathrm{diag}\left(-\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i r}\right)\mathbf{U}^T \end{bmatrix} \tag{A.100}$$

Combining the last result with that of (A.94), we obtain

$$\mathbf{P}'_\infty = \begin{bmatrix} \mathbf{U}\,\mathrm{diag}\left(\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i r}\right)\mathbf{U}^T & \mathbf{0}_{2M\times 2M} \\ \mathbf{0}_{2M\times 2M} & \mathbf{U}\,\mathrm{diag}\left(-\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i r}\right)\mathbf{U}^T \end{bmatrix} \tag{A.101}$$

This expression provides an upper bound for the covariance of the augmented state vector after every EKF *propagation* step. In order to derive a bound for the covariance immediately after the *update* step of the EKF, we note that during propagation, the absolute map covariance remains unchanged, while the uncertainty of the relative map increases. Using this observation, we can show that an upper bound on the steady-state covariance matrix of the relative map, immediately after every update step, is given by

$${}^R\mathbf{P}'_\infty = \mathbf{U}\,\mathrm{diag}\left(-\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i r}\right)\mathbf{U}^T \tag{A.102}$$

while the asymptotic uncertainty of the absolute positions of the landmarks in SLAM is bounded above by the matrix

$${}^G\mathbf{P}'_\infty = \mathbf{U}\,\mathrm{diag}\left(-\frac{\lambda_i}{2}+\sqrt{\frac{\lambda_i^2}{4}+\lambda_i r}\right)\mathbf{U}^T \tag{A.103}$$

We now employ the special structure of the matrix $\mathbf{Q}_u$ (cf. (2.107)), in order to compute its eigenvalues in closed form. The following result can be easily verified:

Lemma 14 *The singular value decomposition of the matrix* $\mathbf{A} = a_1\left(\mathbf{1}_{m\times m}\otimes\mathbf{I}_2\right) + a_2\mathbf{I}_{2m}$, *where* $a_1, a_2 > 0$ *is given by:*

$$\mathbf{A} = \mathbf{U}_A \begin{bmatrix} (ma_1+a_2)\mathbf{I}_2 & \mathbf{0}_{2\times 2m} \\ \mathbf{0}_{2m\times 2} & a_2\mathbf{I}_{2m-2} \end{bmatrix} \mathbf{U}_A^T \tag{A.104}$$

where $\mathbf{U}_A$ *is a unitary matrix. Moreover, if the singular value decomposition of a matrix* $\mathbf{A}'$ *is:*

$$\mathbf{A}' = \mathbf{U}_A \begin{bmatrix} s_1\mathbf{I}_2 & \mathbf{0}_{2\times 2m} \\ \mathbf{0}_{2m\times 2} & s_2\mathbf{I}_{2m-2} \end{bmatrix} \mathbf{U}_A^T \tag{A.105}$$

then this matrix is equal to

$$\mathbf{A}' = \frac{s_1 - s_2}{m} \left(\mathbf{1}_{m\times m} \otimes \mathbf{I}_2\right) + s_2\mathbf{I}_{2m} \tag{A.106}$$

By application of this lemma, for $\mathbf{U}_A \leftarrow \mathbf{U}$, $a_1 \leftarrow q_1$, $a_2 \leftarrow q_2$, $m \leftarrow M$, we obtain the eigenvalues of $\mathbf{Q}_u$ as $\lambda_i = Mq_1 + q_2$, for $i = 1, 2$, and $\lambda_i = q_2$, for $i = 3, \ldots, 2M$. As a result, the singular values of the upper bounds shown in (A.102) and (A.103) are given by:

$$\lambda_{p_i} = -\frac{Mq_1 + q_2}{2} + \sqrt{\frac{(Mq_1 + q_2)^2}{4} + (Mq_1 + q_2)r}, \quad i = 1, 2 \tag{A.107}$$

$$\lambda_{p_i} = -\frac{q_2}{2} + \sqrt{\frac{q_2^2}{4} + q_2 r}, \quad i = 3, \ldots, 2M \tag{A.108}$$

The desired result follows directly from application of the second result of Lemma 14.

A.12 Upper bounds on robot pose uncertainty in SLAM

In this appendix, we derive some intermediate results that are used in the computation of upper bounds on the robot pose uncertainty in SLAM. Using the asymptotic results from (2.114) and the values of the Jacobian $\mathbf{H}_{X_k}$ from (2.121), we obtain:

$$\mathbf{H}_{X_k}\mathbf{P}_\infty^u\mathbf{H}_{X_k}^T = 2b_1(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) + 2b_2\mathbf{I}_{2M} \Rightarrow \tag{A.109}$$

$$\left(\mathbf{H}_{X_k}\mathbf{P}_\infty^u\mathbf{H}_{X_k}^T\right)^{-1} = \underbrace{\frac{1}{2b_2}}_{\alpha}\mathbf{I}_{2M} - \underbrace{\frac{b_1}{b_2(2b_2 + 2b_1M)}}_{\beta}(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) \tag{A.110}$$

$$= \alpha\mathbf{I}_{2M} - \beta(\mathbf{1}_{M\times M} \otimes \mathbf{I}_2) \tag{A.111}$$

where we have used the result of Appendix D. Substitution in (2.124) yields the following asymptotic value for $\mathbf{P}^{u}_{\boldsymbol{\theta\theta}}$:

$$\begin{bmatrix} (\alpha M - \beta M^2)\mathbf{I}_2 & (\alpha - \beta M)\sum_{i=1}^{M} \breve{\mathbf{p}}_i \\ (\alpha - \beta M)\sum_{i=1}^{M} \breve{\mathbf{p}}_i^T & \alpha \sum_{i=1}^{M} \left(\breve{\mathbf{p}}_i^T \breve{\mathbf{p}}_i\right) - \beta \left(\sum_{i=1}^{M} \breve{\mathbf{p}}_i\right)^T \left(\sum_{i=1}^{M} \breve{\mathbf{p}}_i\right) \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{P}_{pp} & \mathbf{P}_{P\phi} \\ \mathbf{P}_{P\phi}^T & P_{\phi\phi} \end{bmatrix} \tag{A.112}$$

Employing the formula for the inversion of a partitioned matrix (cf. Appendix E), we obtain the following expression for $P_{\phi\phi}$, which is an upper bound of the asymptotic orientation variance:

$$P_{\phi\phi} = \frac{1}{\alpha} \frac{1}{\sum_{i=1}^{M} \left(\breve{\mathbf{p}}_i^T \breve{\mathbf{p}}_i\right) - \frac{1}{M} \left(\sum_{i=1}^{M} \breve{\mathbf{p}}_i^T\right) \left(\sum_{i=1}^{M} \breve{\mathbf{p}}_i\right)} \tag{A.113}$$

For any i, j, the property $\breve{\mathbf{p}}_i^T \breve{\mathbf{p}}_j = {}^{R}\mathbf{p}_i^{TR}\mathbf{p}_j$ holds, and thus we can re-write the denominator of the expression for $P_{\phi\phi}$ as

$$D = \frac{1}{M} \left(M \sum_{i=1}^{M} \left({}^{R}\mathbf{p}_i^{TR}\mathbf{p}_i\right) - \left(\sum_{i=1}^{M} {}^{R}\mathbf{p}_i^T\right) \left(\sum_{i=1}^{M} {}^{R}\mathbf{p}_i\right) \right)$$

Moreover, if we denote the distance between landmarks i and j as ρ_{ij}, we obtain

$$\begin{aligned} \sum_{i=1}^{M}\sum_{j=1}^{M} \rho_{ij}^2 &= \sum_{i=1}^{M}\sum_{j=1}^{M} ({}^{R}\mathbf{p}_i - {}^{R}\mathbf{p}_j)^T ({}^{R}\mathbf{p}_i - {}^{R}\mathbf{p}_j) \\ &= \sum_{i=1}^{M} \left(\sum_{j=1}^{M} \left({}^{R}\mathbf{p}_i^{TR}\mathbf{p}_i\right) + \sum_{j=1}^{M} \left({}^{R}\mathbf{p}_j^{TR}\mathbf{p}_j\right) - 2\sum_{j=1}^{M} \left({}^{R}\mathbf{p}_j^{TR}\mathbf{p}_i\right) \right) \\ &= \sum_{i=1}^{M} \left(M\,{}^{R}\mathbf{p}_i^{TR}\mathbf{p}_i + \sum_{j=1}^{M} \left({}^{R}\mathbf{p}_j^{TR}\mathbf{p}_j\right) - 2\,{}^{R}\mathbf{p}_i^T \left(\sum_{j=1}^{M} {}^{R}\mathbf{p}_j\right) \right) \\ &= M\sum_{i=1}^{M} \left({}^{R}\mathbf{p}_i^{TR}\mathbf{p}_i\right) + M\sum_{j=1}^{M} \left({}^{R}\mathbf{p}_j^{TR}\mathbf{p}_j\right) - 2\left(\sum_{i=1}^{M} {}^{R}\mathbf{p}_i^T\right)\left(\sum_{j=1}^{M} {}^{R}\mathbf{p}_j\right) \\ &= 2\left(M\sum_{i=1}^{M} \left({}^{R}\mathbf{p}_i^{TR}\mathbf{p}_i\right) - \left(\sum_{i=1}^{M} {}^{R}\mathbf{p}_i^T\right)\left(\sum_{i=1}^{M} {}^{R}\mathbf{p}_i\right) \right) \\ &= 2MD \end{aligned} \tag{A.114}$$

Using this result, the upper bound on the robot's orientation uncertainty is written as:

$$P_{\phi\phi} = \frac{1}{\alpha}\frac{2M}{\sum_{i=1}^{M}\sum_{j=1}^{M}\rho_{ij}^2} = \frac{4Mb_2}{\sum_{i=1}^{M}\sum_{j=1}^{M}\rho_{ij}^2} \tag{A.115}$$

We now show how an upper bound on the covariance matrix of the robot's position estimates can be determined. From (A.112) we obtain:

$$\mathbf{P}_{pp} = \left((\alpha M - \beta M^2)\mathbf{I}_2 - \frac{(\alpha-\beta M)^2}{\alpha\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \beta\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)^T\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\right)^{-1}$$

which, by application of the matrix inversion lemma (cf. Appendix D) and simple algebraic manipulation, yields:

$$\begin{aligned}\mathbf{P}_{pp} &= \frac{1}{\alpha M - \beta M^2}\mathbf{I}_2 + \frac{\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)}{\alpha M^2\left(\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\right)} \\ &= \frac{2b_2 + 2Mb_1}{M}\mathbf{I}_2 + \underbrace{\frac{2b_2\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)}{M^2\left(\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\right)}}_{\mathbf{T}_2}\end{aligned}$$

We now compute the trace of $\mathbf{T}_2$, which will be useful in determining an upper bound on $\mathbf{P}_{pp}$:

$$\begin{aligned}\text{trace}(\mathbf{T}_2) &= \frac{2b_2}{M}\,\text{trace}\left(\frac{\frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)}{\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)}\right) \\ &= \frac{2b_2}{M}\left(\frac{\frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)}{\sum_{i=1}^{M}\left(\breve{\mathbf{p}}_i^T\breve{\mathbf{p}}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i^T\right)\left(\sum_{i=1}^{M}\breve{\mathbf{p}}_i\right)}\right) \\ &= \frac{2b_2}{M}\left(\frac{\frac{1}{M}\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i^T\right)\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i\right)}{\sum_{i=1}^{M}\left({}^R\mathbf{p}_i^T{}^R\mathbf{p}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i^T\right)\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i\right)}\right) \\ &= \frac{2b_2}{M}\left(\frac{\sum_{i=1}^{M}\left({}^R\mathbf{p}_i^T{}^R\mathbf{p}_i\right)}{\sum_{i=1}^{M}\left({}^R\mathbf{p}_i^T{}^R\mathbf{p}_i\right) - \frac{1}{M}\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i^T\right)\left(\sum_{i=1}^{M}{}^R\mathbf{p}_i\right)} - 1\right)\end{aligned}$$

$$= \frac{4b_2}{M} \frac{\sum_{i=1}^{M} \left({}^{R}\mathbf{p}_i^{T\,R}\mathbf{p}_i \right)}{\sum_{i=1}^{M} \sum_{j=1}^{M} \rho_{ij}^2} - \frac{2b_2}{M} \tag{A.116}$$

Appendix B

Appendix for Chapter 3

B.1 Proof of SDP equivalence

In this appendix, we prove that the objective value corresponding to the point $Y^\star = (f_1^\star, \ldots, f_M^\star, \mathbf{P}_{ss}^\star)$, with $\mathbf{P}_{ss}^\star$ defined in (3.41), is equal to the optimal objective value for problem (3.40), i.e., that

$$\mathbf{W}_p \mathbf{P}_{ss}^\star \mathbf{W}_p^T \;=\; \mathbf{W}_p \mathcal{P}^\star \mathbf{W}_p^T \tag{B.1}$$

To simplify the notation, in the following derivations we employ the substitutions $\mathbf{C} = \sum_{i=1}^M f_i \mathbf{C}_i$ and $\mathbf{C}^\star = \sum_{i=1}^M f_i^\star \mathbf{C}_i$. In order to prove (B.1) we will employ three intermediate results:

Intermediate Result 1

Pre- and post-multiplying (3.39) by $\mathbf{J}^{-1}$ results in the equivalent matrix inequality:

$$-\mathbf{F}_c \mathbf{J}^{-1} - \mathbf{J}^{-1}\mathbf{F}_c^T - \mathbf{Q}_c + \mathbf{J}^{-1}\mathbf{C}\mathbf{J}^{-1} \;\succeq\; \mathbf{0}$$

Thus, at the optimal solution, we obtain

$$\mathbf{F}_c \mathbf{J}^{\star -1} + \mathbf{J}^{\star -1}\mathbf{F}_c^T + \mathbf{Q}_c - \mathbf{J}^{\star -1}\mathbf{C}^\star \mathbf{J}^{\star -1} = \mathbf{A}$$

where $\mathbf{A} \preceq \mathbf{0}$. If we denote $\mathbf{Q}_c' = \mathbf{Q}_c - \mathbf{A}$, then it is $\mathbf{Q}_c' \succeq \mathbf{Q}_c$, and we see that $\mathbf{J}^{\star -1}$ satisfies an ARE given by

$$\mathbf{F}_c \mathbf{J}^{\star -1} + \mathbf{J}^{\star -1} \mathbf{F}_c^T + \mathbf{Q}_c' - \mathbf{J}^{\star -1} \mathbf{C}^\star \mathbf{J}^{\star -1} = \mathbf{0}$$

It can be shown, that the solution of an algebraic Riccati equation is a monotonically increasing function of $\mathbf{Q}_c$ [45]. Therefore, by comparison of the last ARE to the ARE in (3.41), we conclude that

$$\mathbf{J}^{\star -1} \succeq \mathbf{P}_{ss}^\star \Rightarrow \mathbf{J}^\star \preceq \mathbf{P}_{ss}^{\star -1} \tag{B.2}$$

Additionally, from the property $\mathbf{J}^{\star -1} \succeq \mathbf{P}_{ss}^\star$ we derive the first intermediate result:

$$\mathbf{W}_p \mathbf{J}^{\star -1} \mathbf{W}_p^T \succeq \mathbf{W}_p \mathbf{P}_{ss}^\star \mathbf{W}_p^T \tag{B.3}$$

Intermediate Result 2

The Karush-Kuhn-Tucker (KKT) optimality conditions [12] for problem (3.40) include the following "complementary slackness" conditions:

$$\operatorname{trace}(\mathbf{\Lambda}_1^\star (\mathbf{J}^{\star -1} - \mathcal{P}^\star)) = 0 \tag{B.4}$$

$$\operatorname{trace}\left(\mathbf{\Lambda}_2^\star \left(\mathbf{J}^\star \mathbf{F}_c + \mathbf{F}_c^T \mathbf{J}^\star - \mathbf{C}^\star + \mathbf{J}^\star \mathbf{Q}_c \mathbf{J}^\star \right) \right) = 0 \tag{B.5}$$

$$\lambda_i^\star f_i^\star = 0, \quad i = 1 \ldots M$$

$$\mu_i^\star (f_i^\star - f_{i_{\max}}) = 0, \quad i = 1 \ldots M$$

$$\nu^\star \left(\sum_{i=1}^{M} f_i^\star - f_{\text{total}} \right) = 0$$

$$\xi_i^\star \left(\mathbf{e}_{3i}^T \mathcal{P}^\star \mathbf{e}_{3i} - \epsilon_\phi \right) = 0, \quad i = 1 \ldots N$$

as well as the "stationarity" condition:

$$\nabla \operatorname{trace}(\mathbf{W}_p \mathcal{P} \mathbf{W}_p^T) + \nabla \operatorname{trace}\left(\mathbf{\Lambda}_1^\star \left(\mathbf{J}^{-1} - \mathcal{P} \right) \right) + \nabla \operatorname{trace}\left(\mathbf{\Lambda}_2^\star \left(\mathbf{J}\mathbf{F}_c + \mathbf{F}_c^T \mathbf{J} - \mathbf{C} + \mathbf{J}\mathbf{Q}_c\mathbf{J} \right) \right)$$

$$-\sum_{i=1}^{M}\nabla\lambda_i^\star f_i+\sum_{i=1}^{M}\nabla\mu_i^\star(f_i-f_{i_{\max}})+\nabla\nu^\star\left(\sum_{i=1}^{M}f_i-f_{\text{total}}\right)+\sum_{i=1}^{N}\nabla\xi_i^\star\left(\mathbf{e}_{3i}^T\mathcal{P}\mathbf{e}_{3i}-\epsilon_\phi\right)=0 \tag{B.6}$$

where $\mathbf{\Lambda}_1,\mathbf{\Lambda}_2\in\mathbf{S}_+^{3N}$, and $\lambda_i,\ \mu_i,\ \nu,\ \xi_i\geq 0$ are the variables of the dual problem, and the superscript * indicates the value of a variable at the optimal solution. In (B.6) differentiation is with respect to the primal variables $\mathcal{P}$, $\mathbf{J}$, f_i, and the derivatives are computed at the optimal solution. Applying the derivative with respect to $\mathcal{P}$, and evaluating at the optimal point, yields:

$$\begin{aligned}\mathbf{0}&=\mathbf{W}_p\mathbf{W}_p^T-\mathbf{\Lambda}_1^\star+\sum_{i=1}^{N}\xi_i^\star\mathbf{e}_{3i}\mathbf{e}_{3i}^T\Rightarrow\\ \mathbf{\Lambda}_1^\star&=\mathbf{W}_p\mathbf{W}_p^T+\sum_{i=1}^{N}\xi_i^\star\mathbf{e}_{3i}\mathbf{e}_{3i}^T\ \Rightarrow\ \mathbf{\Lambda}_1^\star=\mathbf{W}_p'\mathbf{W}_p'^T\end{aligned} \tag{B.7}$$

where $\mathbf{W}_p'$ is a diagonal matrix, whose diagonal elements corresponding to the robots' positions are equal to 1, while the elements corresponding to the robots' orientation are equal to $\sqrt{\xi_i^\star}$, $i=1\ldots N$.

We now employ the KKT complementary slackness condition with respect to the dual variable $\mathbf{\Lambda}_1$ (cf. (B.4)), to obtain:

$$\begin{aligned}\text{trace}(\mathbf{\Lambda}_1^\star(\mathcal{P}^\star-\mathbf{J}^{\star-1}))&=0\Rightarrow\\ \text{trace}(\mathbf{W}_p'^T(\mathcal{P}^\star-\mathbf{J}^{\star-1})\mathbf{W}_p')&=0\Rightarrow\\ \mathbf{W}_p'^T(\mathcal{P}^\star-\mathbf{J}^{\star-1})\mathbf{W}_p'&=\mathbf{0}\end{aligned} \tag{B.8}$$

This result follows from the fact that for any symmetric (positive or negative) semidefinite matrix A,

$$\text{trace}(\mathbf{A})=0\Rightarrow\mathbf{A}=\mathbf{0}$$

Pre- and post-multiplying (B.8) by $\mathbf{W}_p=\mathbf{W}_p^T$, and using the fact that $\mathbf{W}_p\mathbf{W}_p'=\mathbf{W}_p$, we obtain the second intermediate result:

$$\mathbf{W}_p\mathcal{P}^\star\mathbf{W}_p^T=\mathbf{W}_p\mathbf{J}^{\star-1}\mathbf{W}_p^T \tag{B.9}$$

Intermediate Result 3

Applying (B.6) for the derivative with respect to $\mathbf{J}$, and evaluating at the optimal solution, yields

$$-\mathbf{J}^{\star-1}\mathbf{\Lambda}_1^\star\mathbf{J}^{\star-1} + \mathbf{F}_c^T\mathbf{\Lambda}_2^\star + \mathbf{\Lambda}_2^\star\mathbf{F}_c + \mathbf{\Lambda}_2^\star\mathbf{J}^\star\mathbf{Q}_c + \mathbf{Q}_c\mathbf{J}^\star\mathbf{\Lambda}_2^\star = \mathbf{0} \tag{B.10}$$

We now pre-multiply (3.41) by $\mathbf{\Lambda}_2^\star\mathbf{P}_{ss}^{\star-1}$, post-multiply by $\mathbf{P}_{ss}^{\star-1}$, and apply the trace operator, to obtain the identity

$$\text{trace}\left(\mathbf{\Lambda}_2^\star\left(\mathbf{P}_{ss}^{\star-1}\mathbf{F}_c + \mathbf{F}_c^T\mathbf{P}_{ss}^{\star-1} - \mathbf{C}^\star + \mathbf{P}_{ss}^{\star-1}\mathbf{Q}_c\mathbf{P}_{ss}^{\star-1}\right)\right) = 0$$

Subtracting this equation from the second complementary slackness condition (cf. (B.5)), and rearranging terms, we find

$$\text{trace}\left(\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\left(\mathbf{\Lambda}_2^\star\mathbf{F}_c + \mathbf{F}_c^T\mathbf{\Lambda}_2^\star + \mathbf{\Lambda}_2^\star\mathbf{J}^\star\mathbf{Q}_c + \mathbf{Q}_c\mathbf{P}_{ss}^{\star-1}\mathbf{\Lambda}_2^\star\right)\right) = 0 \tag{B.11}$$

Using the result of (B.10) to simplify this expression, and separating terms, yields

$$\text{trace}\left(\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{J}^{\star-1}\mathbf{\Lambda}_1^\star\mathbf{J}^{\star-1}\right) = \text{trace}\left(\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{Q}_c(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1})\mathbf{\Lambda}_2^\star\right) \tag{B.12}$$

At this point, we note that the right-hand side of this equation is a nonnegative quantity, since the matrices $\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{Q}_c(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1})$ and $\mathbf{\Lambda}_2^\star$ are symmetric positive semidefinite. We now show that the left hand side of (B.12) is nonpositive. Using the expression of (B.7), as well as the property $\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1} \preceq \mathbf{0}$ (cf. (B.2)), we obtain

$$\begin{aligned}
\alpha &= \text{trace}\left(\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{J}^{\star-1}\mathbf{\Lambda}_1^\star\mathbf{J}^{\star-1}\right) \\
&= \text{trace}\left(\mathbf{W}_p'^T\mathbf{J}^{\star-1}\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{J}^{\star-1}\mathbf{W}_p'\right) \preceq 0
\end{aligned}$$

Combining this last result and the fact that the right-hand side of (B.12) is a nonnegative quantity, we conclude that both sides must be equal to zero. Consequently,

$$\begin{aligned}
\mathbf{W}_p'^T\mathbf{J}^{\star-1}\left(\mathbf{J}^\star - \mathbf{P}_{ss}^{\star-1}\right)\mathbf{J}^{\star-1}\mathbf{W}_p' &= \mathbf{0} \Rightarrow \\
\mathbf{W}_p'^T\mathbf{J}^{\star-1}\mathbf{W}_p' - \mathbf{W}_p'^T\mathbf{J}^{\star-1}\mathbf{P}_{ss}^{\star-1}\mathbf{J}^{\star-1}\mathbf{W}_p' &= \mathbf{0}
\end{aligned} \tag{B.13}$$

We now consider the following matrix:

$$\mathbf{E} \;=\; \begin{bmatrix} \mathbf{P}^{\star}_{ss} & -\mathbf{J}^{\star -1}\mathbf{W}'_p \\ -\mathbf{W}'^{T}_p\mathbf{J}^{\star -1} & \mathbf{W}'^{T}_p\mathbf{J}^{\star -1}\mathbf{W}'_p \end{bmatrix}$$

This matrix will be used in conjunction with the following lemma:

Lemma 15 *If* $\mathbf{A} \succ \mathbf{0}$*, and* $\mathbf{D}$ *is symmetric, then for any vector* $\mathbf{y}$ *of appropriate dimensions, the minimum of*

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}^T \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

with respect to $\mathbf{x}$ *is equal to* $\mathbf{y}^T\left(\mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}\right)\mathbf{y}$ *and is attained for* $\mathbf{x} = -\mathbf{A}^{-1}\mathbf{B}\mathbf{y}$.

Applying this result, we see that the minimum value of the quadratic product $[\mathbf{u}^T \;\; \mathbf{v}^T]\mathbf{E}[\mathbf{u}^T \;\; \mathbf{v}^T]^T$ over all vectors $[\mathbf{u}^T \;\; \mathbf{v}^T]^T$ is equal to

$$\mathbf{v}^T\left(\mathbf{W}'^{T}_p\mathbf{J}^{\star -1}\mathbf{W}'_p - \mathbf{W}'^{T}_p\mathbf{J}^{\star -1}\mathbf{P}^{\star -1}_{ss}\mathbf{J}^{\star -1}\mathbf{W}'_p\right)\mathbf{v}$$

Using the result of (B.13) we conclude that the minimum value of the quadratic product $[\mathbf{u}^T \;\; \mathbf{v}^T]\mathbf{E}[\mathbf{u}^T \;\; \mathbf{v}^T]^T$ equals zero, and thus $\mathbf{E}$ is positive semidefinite. Therefore

$$\begin{aligned} \begin{bmatrix} \mathbf{W}_p & \mathbf{W}_p \end{bmatrix} \mathbf{E} \begin{bmatrix} \mathbf{W}_p & \mathbf{W}_p \end{bmatrix}^T \;&\succeq\; \mathbf{0} \Rightarrow \\ \mathbf{W}_p\mathbf{P}^{\star}_{ss}\mathbf{W}^T_p - \mathbf{W}_p\mathbf{J}^{\star -1}\mathbf{W}^T_p \;&\succeq\; \mathbf{0} \\ \mathbf{W}_p\mathbf{P}^{\star}_{ss}\mathbf{W}^T_p \;&\succeq\; \mathbf{W}_p\mathbf{J}^{\star -1}\mathbf{W}^T_p \end{aligned} \tag{B.14}$$

where we have used the fact that $\mathbf{W}'_p\mathbf{W}_p = \mathbf{W}_p$. Equation (B.14) is the third intermediate result.

Proof of (B.1)*:* Substituting from (B.9) in (B.3) and (B.14) we obtain

$$\mathbf{W}_p\mathcal{P}^{\star}\mathbf{W}^T_p \succeq \mathbf{W}_p\mathbf{P}^{\star}_{ss}\mathbf{W}^T_p$$

and

$$\mathbf{W}_p \mathbf{P}_{ss}^{\star} \mathbf{W}_p^T \succeq \mathbf{W}_p \mathcal{P}^{\star} \mathbf{W}_p^T$$

respectively. The desired result of (B.1) follows directly from the last two relations.

Appendix C

Appendices for Chapter 4

C.1 IMU Propagation

The filter propagation equations are derived by discretization of the continuous-time IMU system model, as described in the following:

Continuous-time system modeling

The time evolution of the IMU state is described by [19]:

$$\begin{gathered} {}^{I}_{G}\dot{\bar{q}}(t) = \tfrac{1}{2}\mathbf{\Omega}\big(\boldsymbol{\omega}(t)\big){}^{I}_{G}\bar{q}(t), \quad \dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t) \\ {}^{G}\dot{\mathbf{v}}_I(t) = {}^{G}\mathbf{a}(t), \quad \dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t), \quad {}^{G}\dot{\mathbf{p}}_I(t) = {}^{G}\mathbf{v}_I(t) \end{gathered} \tag{C.1}$$

In these expressions ${}^{G}\mathbf{a}$ is the body acceleration in the global frame, $\boldsymbol{\omega} = [\omega_x \;\; \omega_y \;\; \omega_z]^T$ is the rotational velocity expressed in the IMU frame, and

$$\mathbf{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor \boldsymbol{\omega} \times \rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix}, \quad \lfloor \boldsymbol{\omega} \times \rfloor = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$$

The gyroscope and accelerometer measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$ respectively, are given by [19]:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{C}({}^{I}_{G}\bar{q})\boldsymbol{\omega}_G + \mathbf{b}_g + \mathbf{n}_g \tag{C.2}$$

$$\mathbf{a}_m = \mathbf{C}({}^I_G\bar{q})({}^G\mathbf{a} - {}^G\mathbf{g} + 2\lfloor\boldsymbol{\omega}_G\times\rfloor\,{}^G\mathbf{v}_I + \lfloor\boldsymbol{\omega}_G\times\rfloor^2\,{}^G\mathbf{p}_I) + \mathbf{b}_a + \mathbf{n}_a \tag{C.3}$$

where $\mathbf{C}(\cdot)$ denotes a rotational matrix, and $\mathbf{n}_g$ and $\mathbf{n}_a$ are zero-mean, white Gaussian noise processes modeling the measurement noise. It is important to note that the IMU measurements incorporate the effects of the planet's rotation, $\boldsymbol{\omega}_G$. Moreover, the accelerometer measurements include the gravitational acceleration, ${}^G\mathbf{g}$, expressed in the local frame.

Applying the expectation operator in the state propagation equations (C.1) we obtain the continuous-time first-order approximation equations for propagating the estimates of the evolving IMU state:

$$\dot{{}^I_G\hat{q}} = \tfrac{1}{2}\boldsymbol{\Omega}(\hat{\boldsymbol{\omega}})\,{}^I_G\hat{q}, \quad \dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3\times1},$$

$$\dot{{}^G\hat{\mathbf{v}}}_I = \mathbf{C}({}^I_G\hat{q})^T\hat{\mathbf{a}} - 2\lfloor\boldsymbol{\omega}_G\times\rfloor\,{}^G\hat{\mathbf{v}}_I - \lfloor\boldsymbol{\omega}_G\times\rfloor^2\,{}^G\hat{\mathbf{p}}_I + {}^G\mathbf{g} \tag{C.4}$$

$$\dot{\hat{\mathbf{b}}}_a = \mathbf{0}_{3\times1}, \quad \dot{{}^G\hat{\mathbf{p}}}_I = {}^G\hat{\mathbf{v}}_I$$

where $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$ and $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g - \mathbf{C}({}^I_G\hat{q})\boldsymbol{\omega}_G$.

From (C.1) and (C.4) we obtain the linearized continuous-time model for the propagation of the IMU error-state:

$$\dot{\widetilde{\mathbf{X}}}_{\text{IMU}} = \mathbf{F}\widetilde{\mathbf{X}}_{\text{IMU}} + \mathbf{G}\mathbf{n}_{\text{IMU}} \tag{C.5}$$

where $\mathbf{n}_{\text{IMU}} = \begin{bmatrix}\mathbf{n}_g^T & \mathbf{n}_{wg}^T & \mathbf{n}_a^T & \mathbf{n}_{wa}^T\end{bmatrix}^T$ is the system noise. The covariance matrix of $\mathbf{n}_{\text{IMU}}$, $\mathbf{Q}_{\text{IMU}}$, depends on the IMU noise characteristics and is computed off-line during sensor calibration [3]. Finally, the matrices $\mathbf{F}$ and $\mathbf{G}$ that appear in (C.5) are given by:

$$\mathbf{F} = \begin{bmatrix} -\lfloor\hat{\boldsymbol{\omega}}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\mathbf{C}({}^I_G\hat{q})^T\lfloor\hat{\mathbf{a}}\times\rfloor & \mathbf{0}_{3\times3} & -2\lfloor\boldsymbol{\omega}_G\times\rfloor & -\mathbf{C}({}^I_G\hat{q})^T & -\lfloor\boldsymbol{\omega}_G\times\rfloor^2 \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}$$

and

$$\mathbf{G} = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\mathbf{C}({}^{I}_{G}\hat{\bar{q}})^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}$$

Discrete-time implementation

The IMU samples the signals $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$ with a period T, and these measurements are used for state propagation in the EKF. Every time a new IMU measurement is received, the IMU state estimate is propagated using 5th order Runge-Kutta numerical integration of (C.4). Moreover, the covariance matrix of the IMU state estimates is propagated using numerical integration of the following Lyapunov equation:

$$\dot{\mathbf{P}}_{II} = \mathbf{F}\mathbf{P}_{II} + \mathbf{P}_{II}\mathbf{F}^T + \mathbf{G}\mathbf{Q}_{\text{IMU}}\mathbf{G}^T \tag{C.6}$$

which describes the time evolution of the covariance in the linearized error-state model of (C.5). Using numerical integration in the time interval $[t_k, t_{k+1}]$, with initial condition $\mathbf{P}_{II_{k|k}}$, we obtain the term $\mathbf{P}_{II_{k+1|k}}$ (cf. (4.7) and (4.8)).

Finally, for propagating the covariance matrix of the entire EKF state vector, the state-transition matrix of the IMU errors, $\boldsymbol{\Phi}(k+1, k)$, is required. This matrix is similarly computed by numerical integration of the differential equation

$$\dot{\boldsymbol{\Phi}}(t_k + \tau, t_k) = \mathbf{F}\boldsymbol{\Phi}(t_k + \tau, t_k), \quad \tau \in [0, T] \tag{C.7}$$

in the time interval $[t_k, t_{k+1}]$, with initial condition $\boldsymbol{\Phi}(t_k, t_k) = \mathbf{I}_{15}$.

C.2 Feature initialization

To compute an estimate of the position of a tracked feature f_j we employ *intersection* [162]. To avoid local minima, and for better numerical stability, during this process we use an

inverse-depth parametrization of the feature position [99]. In particular, if $\{C_n\}$ is the camera frame in which the feature was observed for the first time, then the feature coordinates with respect to the camera at the i-th time instant are:

$$^{C_i}\mathbf{p}_{f_j} = \mathbf{C}(^{C_i}_{C_n}\bar{q})\,^{C_n}\mathbf{p}_{f_j} + {}^{C_i}\mathbf{p}_{C_n}, \quad i \in \mathcal{S}_j \tag{C.8}$$

In this expression $\mathbf{C}(^{C_i}_{C_n}\bar{q})$ and $^{C_i}\mathbf{p}_{C_n}$ are the rotation and translation between the camera frames at time instants n and i, respectively. Equation (C.8) can be rewritten as:

$$^{C_i}\mathbf{p}_{f_j} = {}^{C_n}Z_j \left(\mathbf{C}(^{C_i}_{C_n}\bar{q}) \begin{bmatrix} \frac{^{C_n}X_j}{^{C_n}Z_j} \\ \frac{^{C_n}Y_j}{^{C_n}Z_j} \\ 1 \end{bmatrix} + \frac{1}{^{C_n}Z_j}\,^{C_i}\mathbf{p}_{C_n} \right) \tag{C.9}$$

$$= {}^{C_n}Z_j \left(\mathbf{C}(^{C_i}_{C_n}\bar{q}) \begin{bmatrix} \alpha_j \\ \beta_j \\ 1 \end{bmatrix} + \rho_j\ {}^{C_i}\mathbf{p}_{C_n} \right) \tag{C.10}$$

$$= {}^{C_n}Z_j \begin{bmatrix} h_{i1}(\alpha_j, \beta_j, \rho_j) \\ h_{i2}(\alpha_j, \beta_j, \rho_j) \\ h_{i3}(\alpha_j, \beta_j, \rho_j) \end{bmatrix} \tag{C.11}$$

In the last expression h_{i1}, h_{i2} and h_{i3} are scalar functions of the quantities $\alpha_j, \beta_j, \rho_j$, which are defined as:

$$\alpha_j = \frac{^{C_n}X_j}{^{C_n}Z_j}, \quad \beta_j = \frac{^{C_n}Y_j}{^{C_n}Z_j}, \quad \rho_j = \frac{1}{^{C_n}Z_j}, \tag{C.12}$$

Substituting from (C.11) into (4.13) we can express the measurement equations as functions of α_j, β_j and ρ_j only:

$$\mathbf{z}_i^{(j)} = \frac{1}{h_{i3}(\alpha_j, \beta_j, \rho_j)} \begin{bmatrix} h_{i1}(\alpha_j, \beta_j, \rho_j) \\ h_{i2}(\alpha_j, \beta_j, \rho_j) \end{bmatrix} + \mathbf{n}_i^{(j)} \tag{C.13}$$

Given the measurements $\mathbf{z}_i^{(j)}, i \in \mathcal{S}_j$, and the estimates for the camera poses in the state vector, we can obtain estimates for $\hat{\alpha}_j, \hat{\beta}_j$, and $\hat{\rho}_j$, using Gauss-Newton least-squares mini-

mization. Then, the global feature position is computed by:

$$^G\hat{\mathbf{p}}_{f_j} = \frac{1}{\hat{\rho}_j}\mathbf{C}^T(^{C_n}_{G}\hat{q}) \begin{bmatrix} \hat{\alpha}_j \\ \hat{\beta}_j \\ 1 \end{bmatrix} + {}^G\hat{\mathbf{p}}_{C_n} \tag{C.14}$$

We note that during the least-squares minimization process the camera pose estimates are treated as known constants, and their covariance matrix is ignored. As a result, the minimization can be carried out very efficiently, at the expense of the optimality of the feature position estimates. Recall, however, that up to a first-order approximation, the errors in these estimates do *not* affect the measurement residual (cf. (4.20)). Thus, no significant degradation of performance is inflicted.

C.3 MSC-KF optimality

We here prove that applying the MSC-KF measurement model results in optimal use of the feature information, except for the inaccuracies caused by linearization. The proof is carried out for the case of a single feature, but it can be trivially generalized to the case where more features are processed.

Given the measurement equation (4.21), the EKF computes the exact (up to linearization) covariance for the resulting state estimates. Thus, for the purposes of our proof it suffices to show that the EKF covariance is identical to that arising from a non-linear least-squares estimator, which is known to be optimal up to linearization.

The covariance of the updated state computed by the MSC-KF, when a single feature is used for the update, can be re-written as [92]:

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{P}^{-1}_{k+1|k} + \mathbf{H}_o^{(j)T}(\sigma^2_{\text{im}}\mathbf{I}_{2M_j-3})^{-1}\mathbf{H}_o^{(j)}\right)^{-1} \tag{C.15}$$

$$= \left(\mathbf{P}^{-1}_{k+1|k} + \frac{1}{\sigma^2_{\text{im}}}\mathbf{H}_o^{(j)T}\mathbf{H}_o^{(j)}\right)^{-1} \tag{C.16}$$

We now show that this is identical to the result arising from the use of non-linear least-squares to process the measurements of the feature. In this case, we formulate a state vector, $\mathbf{X}_{\text{LS}}$ comprising both the MSC-KF state (which consists of the IMU state and the

sliding window of camera poses), as well as the feature position. The *information matrix* for this state vector is given by [30]:

$$\mathcal{I}_{k+1|k+1} = \begin{bmatrix} \mathbf{P}_{k+1|k}^{-1} & \mathbf{0}_{\xi\times 3} \\ \mathbf{0}_{3\times\xi} & \mathbf{0}_{3\times 3} \end{bmatrix} + \frac{1}{\sigma_{\text{im}}^2} \begin{bmatrix} \mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)} & \mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{f}^{(j)} \\ \mathbf{H}_{f}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)} & \mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)} \end{bmatrix} \tag{C.17}$$

$$= \begin{bmatrix} \mathcal{I}_{\mathbf{X}\mathbf{X}} & \mathcal{I}_{\mathbf{X}f} \\ \mathcal{I}_{f\mathbf{X}} & \mathcal{I}_{ff} \end{bmatrix} \tag{C.18}$$

where the first term in (C.17) corresponds to the prior information, and the second to the information carried by the feature measurements. The covariance matrix of $\mathbf{X}_{\text{LS}}$ is equal to $\mathcal{I}_{k+1|k+1}^{-1}$, and therefore the covariance matrix corresponding to the MSC-KF state is the "top-right" submatrix of $\mathcal{I}_{k+1|k+1}^{-1}$. By employing the result of Appendix E, we obtain the following expression for it:

$$\begin{aligned} \mathbf{P}_{\mathbf{X}\mathbf{X}} &= \left(\mathcal{I}_{\mathbf{X}\mathbf{X}} - \mathcal{I}_{\mathbf{X}f}\mathcal{I}_{ff}^{-1}\mathcal{I}_{f\mathbf{X}}\right)^{-1} \\ &= \left(\mathbf{P}_{k+1|k}^{-1} + \frac{1}{\sigma_{\text{im}}^2}\mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)} - \frac{1}{\sigma_{\text{im}}^2}\left(\mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{f}^{(j)}\right)\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)}\right)^{-1}\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)}\right)\right)^{-1} \end{aligned}$$

By comparison of the last expression with that of (C.16), we conclude that for proving equality it suffices to show that the following holds:

$$\mathbf{H}_{o}^{(j)T}\mathbf{H}_{o}^{(j)} = \mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)} - \left(\mathbf{H}_{\mathbf{X}}^{(j)T}\mathbf{H}_{f}^{(j)}\right)\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)}\right)^{-1}\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{\mathbf{X}}^{(j)}\right) \Leftrightarrow \tag{C.19}$$

$$\mathbf{H}_{o}^{(j)T}\mathbf{H}_{o}^{(j)} = \mathbf{H}_{\mathbf{X}}^{(j)T}\left(\mathbf{I}_{2M_j} - \mathbf{H}_{f}^{(j)}\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)}\right)^{-1}\mathbf{H}_{f}^{(j)T}\right)\mathbf{H}_{\mathbf{X}}^{(j)} \tag{C.20}$$

If we let $\mathbf{H}_{f}^{(j)} = \mathbf{Q}\mathbf{R}$ denote the thin QR factorization of $\mathbf{H}_{f}^{(j)}$, then we obtain:

$$\mathbf{I}_{2M_j} - \mathbf{H}_{f}^{(j)}\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)}\right)^{-1}\mathbf{H}_{f}^{(j)T} = \mathbf{I}_{2M_j} - \mathbf{Q}\mathbf{R}\left(\mathbf{R}^T\mathbf{Q}^T\mathbf{Q}\mathbf{R}\right)^{-1}\mathbf{R}^T\mathbf{Q}^T \tag{C.21}$$

Using the fact that $\mathbf{Q}$ is unitary, we obtain:

$$\begin{aligned} \mathbf{I}_{2M_j} - \mathbf{H}_{f}^{(j)}\left(\mathbf{H}_{f}^{(j)T}\mathbf{H}_{f}^{(j)}\right)^{-1}\mathbf{H}_{f}^{(j)T} &= \mathbf{I}_{2M_j} - \mathbf{Q}\mathbf{R}\left(\mathbf{R}^T\mathbf{R}\right)^{-1}\mathbf{R}^T\mathbf{Q}^T \\ &= \mathbf{I}_{2M_j} - \mathbf{Q}\mathbf{Q}^T \end{aligned} \tag{C.22}$$

However, since the columns of $\mathbf{Q}$ form a basis for the range of $\mathbf{H}_f^{(j)}$, we obtain $\mathbf{Q}\mathbf{Q}^T + \mathbf{U}\mathbf{U}^T = \mathbf{I}_{2M_j}$, where we remind that $\mathbf{U}$ is the basis of the left nullspace of $\mathbf{H}_f^{(j)}$ (cf. (4.21)). Thus:

$$\mathbf{I}_{2M_j} - \mathbf{H}_f^{(j)} \left(\mathbf{H}_f^{(j)T} \mathbf{H}_f^{(j)} \right)^{-1} \mathbf{H}_f^{(j)T} = \mathbf{I}_{2M_j} - \mathbf{Q}\mathbf{Q}^T = \mathbf{U}\mathbf{U}^T \tag{C.23}$$

We proceed to substitute this result in (C.20):

$$\mathbf{H}_o^{(j)T} \mathbf{H}_o^{(j)} = \mathbf{H}_{\mathbf{X}}^{(j)T} \mathbf{U}\mathbf{U}^T \mathbf{H}_{\mathbf{X}}^{(j)} \tag{C.24}$$

$$= \left(\mathbf{U}^T \mathbf{H}_{\mathbf{X}}^{(j)} \right)^T \left(\mathbf{U}^T \mathbf{H}_{\mathbf{X}}^{(j)} \right) \tag{C.25}$$

However, from (4.21) we see that $\mathbf{H}_o^{(j)} = \mathbf{U}^T \mathbf{H}_{\mathbf{X}}^{(j)}$, and thus the above relation is a tautology. This completes the proof.

Appendix D

Matrix Inversion Lemma

If $\mathbf{A}$ is an $n \times n$ invertible matrix, $\mathbf{B}$ is an $n \times m$ matrix, $\mathbf{C}$ is an $m \times m$ invertible matrix and $\mathbf{D}$ is an $m \times n$ matrix, then:

$$(\mathbf{A}^{-1}+\mathbf{B}\mathbf{C}^{-1}\mathbf{D})^{-1} = \mathbf{A} - \mathbf{A}\mathbf{B}(\mathbf{D}\mathbf{A}\mathbf{B} + \mathbf{C})^{-1}\mathbf{D}\mathbf{A} \tag{D.1}$$

Appendix E

Partitioned Matrix Inversion

Let a $(m+n)\times(m+n)$ matrix $\mathbf{K}$ be partitioned as

$$\mathbf{K} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}$$

Where the $m \times m$ matrix $\mathbf{A}$ and the $n \times n$ matrix $\mathbf{D}$ are invertible. Then the inverse matrix of $\mathbf{K}$ can be written as

$$\begin{bmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{Z} & \mathbf{U} \end{bmatrix} = \begin{bmatrix} (\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -\mathbf{D}^{-1}\mathbf{C}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} & (\mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix} \quad \text{(E.1)}$$

$$= \begin{bmatrix} (\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} & -(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\mathbf{B}\mathbf{D}^{-1} \\ -(\mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & (\mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix} \quad \text{(E.2)}$$

Bibliography

[1] The International Journal of Robotics Research Special Issue: 2nd Workshop on Integration of Vision and Inertial Sensors. June 2007, Volume 26, No. 6.

[2] J. Adams, A. Robertson, K. Zimmerman, and J. How. Technologies for spacecraft formation flying. In *Proceedings of ION-GPS 96*, pages 1321–1330, New Orleans, LA, Sept. 17-20 1996.

[3] R. O. Allen and D. H. Chang. Performance testing of the systron donner quartz gyro. JPL Engineering Memorandum, EM #343-1297, Jan. 5, 1993.

[4] J. Andade-Cetto, T. Vidal-Calleja, and A. Sanfeliu. Unscented transformation of vehicle states in SLAM. In *Proceedings of the IEEE International Conference of Robotics and Automation*, pages 323 – 328, Barcelona, Spain, Apr. 18-22 2005.

[5] G. Anousaki and K. J. Kyriakopoulos. A dead-reckoning scheme for skid-steered vehicles in outdoor environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 580–585, New Orleans, LA, Apr. 26 - May 1 2004.

[6] D. Avintzour and S. Rogers. Optimal measurement scheduling for prediction and estimation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(10):2017–2023, Oct. 1990.

[7] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. Consistency of the EKF-SLAM algorithm. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3562–3568, Beijing, China, Oct. 2006.

[8] T. Balch and R. C. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, Dec. 1998.

[9] J. S. Baras and A. Bensoussan. Optimal sensor scheduling in nonlinear filtering of diffusion processes. *SIAM Journal of Control and Optimization*, 27(4):786–813, July 1989.

[10] D. S. Bayard and P. B. Brugarolas. An estimation algorithm for vision-based exploration of small bodies in space. In *Proceedings of the American Control Conference*, pages 4589–4595, Portland, Oregon, June 2005.

[11] M. Bosse, R. Rikoski, J. Leonard, and S. Teller. Vanishing points and 3D lines from omnidirectional video. In *Proceedings of the International Conference on Image Processing*, pages 513–516, Rochester, NY, Sep. 2002.

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

[13] W. G. Breckenridge. Quaternions proposed standard conventions. Interoffice Memorandum IOM 343-79-1199, Jet Propulsion Laboratory, Pasadena, CA, 1999.

[14] W. L. Brogan. *Modern Control Theory*. Prentice Hall, Upper Saddle River, NJ, 1991.

[15] A. Broggi, M. Bertozzi, A. Fascioli, C. G. L. Bianco, and A. Piazzi. Visual perception of obstacles and vehicles for platooning. *IEEE Transactions on Intelligent Transportation Systems*, 1(3):164–176, Sep. 2000.

[16] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 476–481, San Francisco, CA, Apr. 24-28 2000.

[17] P. Cain and S. Malwal. Landmark use and development of navigation behaviour in the weakly electric fish gnathonemus petersii. *Journal of Experimental Biology*, 205(24):3915–3923, Dec. 2002.

[18] J. A. Castellanos, J. Neira, and J. D. Tardos. Limits to the consistency of EKF-based SLAM. In *Proceedings of the IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal, July 2004.

[19] A. B. Chatfield. *Fundamentals of High Accuracy Inertial Navigation*. American Institute of Aeronautics and Astronautics, Inc., Reston, VA, 1997.

[20] Y. Cheng, J. Goguen, A. Johnson, C. Leger, L. Matthies, M. S. Martin, and R. Willson. The Mars exploration rovers descent image motion estimation system. *IEEE Intelligent Systems*, 19(3):13–21, May/June 2004.

[21] A. Chiuso, P. Favaro, H. Jin, and S. Soatto. Structure from motion causally integrated over time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):523–535, Apr. 2002.

[22] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, Providence, RI, 1997.

[23] T. H. Chung, V. Gupta, B. Hassibi, J. W. Burdick, and R. M. Murray. Scheduling for distributed sensor networks with single sensor measurement per time step. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 187–192, New Orleans, LA, Apr. 26- May 1 2004.

[24] B. Cochran, H. Mouritsen, and M. Wikelski. Migrating songbirds recalibrate their magnetic compass daily from twilight cues. *Science*, 304:405–408, Apr. 2004.

[25] M. Csorba. *Simultaneous Localization and Map Building*. PhD thesis, University of Oxford, 1997.

[26] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, and C. Taylor. A vision-based formation control framework. *IEEE Transactions on Robotics and Automation*, 18(5):813 – 825, Oct. 2002.

[27] A. Das, J. Spletzer, V. Kumar, and C. Taylor. Ad hoc networks for localization and control. In *Proceedings of the IEEE Conference on Decision and Control*, pages 2978–2983, Las Vegas, NV, Dec. 2002.

[28] A. J. Davison and D. W. Murray. Simultaneous localisation and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):865–880, July 2002.

[29] M. C. Deans. Maximally informative statistics for localization and mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1824–1829, Washington D.C., May 2002.

[30] F. Dellaert and M. Kaess. Square root SAM: Simultaneous localization and mapping via square root information smoothing. *International Journal of Robotics Research*, 25(12):1181–1203, Dec. 2006.

[31] D. D. Diel. Stochastic constraints for vision-aided inertial navigation. Master's thesis, Massachusetts Institute of Technology, Jan. 2005.

[32] D. D. Diel, P. DeBitetto, and S. Teller. Epipolar constraints for vision-aided inertial navigation. In *Proceedings of the IEEE Workshop on Motion and Video Computing*, pages 221–228, Breckenridge, CO, Jan. 2005.

[33] G. Dissanayake, P. Newman, H. Durrant-Whyte, S. Clark, and M. Csorba. A Solution to the Simultaneous Localization and Map Building (SLAM) Problem. *IEEE Transactions of Robotics and Automation*, 17(3):229–241, June 2001.

[34] T. Duckett, S. Marsland, and J. Shapiro. Fast, on-line learning of globally consistent maps. *Autonomous Robots*, 12(3):287–300, May 2002.

[35] H. Durrant-Whyte and T. Bailey. Simultaneous Localisation and Mapping (SLAM): Part I the essential algorithms. *Robotics and Automation Magazine*, 13(2):108 – 117, June 2006.

[36] E. Eade and T. Drummond. Scalable monocular SLAM. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 469 – 476, New York, NY, June 17-26 2006.

[37] C. Engels, H. Stewenius, and D. Nister. Bundle adjustment rules. In *Proceedings of the Photogrammetric Computer Vision Conference*, pages 266–271, Bonn, Germany, Sep. 20-22 2006.

[38] R. Eustice, O. Pizarro, and H. Singh. Visually augmented navigation in an unstructured environment using a delayed state history. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 25–32, New Orleans, LA, Apr. 2004.

[39] R. Eustice, H. Singh, and J. Leonard. Exactly sparse delayed-state filters. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2428–2435, Barcelona, Spain, Apr. 2005.

[40] R. M. Eustice, H. Singh, and J. J. Leonard. Exactly sparse delayed-state filters for view-based SLAM. *IEEE Transactions on Robotics*, 22(6):1100–1114, Dec. 2006.

[41] J. Fenwick, P. Newman, and J. Leonard. Cooperative concurrent mapping and localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1810–1817, Washington D.C., May 11-15 2002.

[42] J. W. Fenwick. Collaborative concurrent mapping and localization. Master's thesis, Massachusetts Institute of Technology, June 2001.

[43] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. Collaborative multi-robot localization. In *Proceedings of the Annual German Conference on Artificial Intelligence*, pages 255–266, Bonn, Germany, Sep. 13-15 1999.

[44] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 8(3):325–344, June 2000.

[45] G. Freiling and V. Ionescu. Monotonicity and convexity properties of matrix Riccati equations. *IMA Journal of Mathematical Control and Information*, 18(1):61–72, Mar. 2001.

[46] D. W. Gage. Minimum resource distributed navigation and mapping. In *Proceedings of SPIE Mobile Robots XV*, volume 4195, pages 96–103, 2000.

[47] R. Garcia, J. Puig, O. Ridao, and X. Cufi. Augmented state Kalman filtering for AUV navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4010–4015, Washington, DC, May 2002.

[48] A. Georgiev and P. K. Allen. Design and analysis of a sun sensor for planetary rover absolute heading detection. *IEEE Transactions on Robotics and Automation*, 17(6):939–947, Dec. 2001.

[49] A. Georgiev and P. K. Allen. Localization methods for a mobile robot in urban environments. *IEEE Transactions on Robotics*, 20(5):851–864, Oct. 2004.

[50] P. Gibbens, G. Dissanayake, and H. Durrant-Whyte. A closed form solution to the single degree of freedom simultaneous localisation and map building (SLAM) problem. In *Proceedings of the IEEE Conference on Decision and Control*, pages 191–196, Sydney, NSW, Australia, Dec. 12-15 2000.

[51] G. Golub and C. van Loan. *Matrix computations*. The Johns Hopkins University Press, London, 1996.

[52] R. Grabowski and P. Khosla. Localization techniques for a team of small robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1067–1072, Maui, HI, Oct.29-Nov.3 2001.

[53] J. E. Guivant and E. M. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. *IEEE Transactions on Robotics and Automation*, 17(3):242–257, June 2001.

[54] V. Gupta, T. Chung, B. Hassibi, and R. M. Murray. Sensor scheduling algorithms requiring limited computation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 825–828, Montreal, Canada, May 2004.

[55] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, Feb. 2006.

[56] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, Manchester, UK, Aug. 31 - Sep. 2 1988.

[57] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2000.

[58] B. Hassibi. *Indefinite Metric Spaces in Estimation, Control and Adaptive Filtering*. PhD thesis, Stanford University, Aug. 1996.

[59] J. A. Hesch and S. I. Roumeliotis. An indoor localization aid for the visually impaired. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3545–3551, Rome, Italy, Apr. 10 - 15 2007.

[60] Y. S. Hidaka, A. I. Mourikis, and S. I. Roumeliotis. Optimal formations for cooperative localization of mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4137–4142, Barcelona, Spain, Apr. 18-22 2005.

[61] A. Howard. Multi-robot mapping using manifold representations. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4198–4203, New Orleans, LA, Apr. 2004.

[62] A. Howard, M. Mataric, and G. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 434–59, Lauzanne, Switzerland, Sep.30-Oct.4 2002.

[63] A. Howard, M. Mataric, and G. Sukhatme. Putting the 'I' in 'team': an ego-centric approach to cooperative localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 868–874, Taipei, Taiwan, Sep. 14-19 2003.

[64] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. Analysis and improvement of the consistency of extended Kalman filter-based SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 473–479, Pasadena, CA, May 2008.

[65] S. Huang and G. Dissanayake. Convergence analysis for extended Kalman filter based SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

[66] S. Huang and G. Dissanayake. Convergence and consistency analysis for extended Kalman filter based SLAM. *IEEE Transactions on Robotics*, 23(5):1036–1049, Oct. 2007.

[67] T. L. Huntsberger, A. Trebi-Ollennu, H. Aghazarian, P. S. Schenker, P. Pirjanian, and H. D. Nayar. Distributed control of multi-robot systems engaged in tightly coupled tasks. *Autonomous Robots*, 17(1):79–92, July 2004.

[68] A. Huster. *Relative position sensing by fusing monocular vision and inertial rate sensors*. PhD thesis, Department of Electrical Engineering, Stanford University, 2003.

[69] A. E. Johnson and L. H. Matthies. Precise image-based motion estimation for autonomous small body exploration. In *Proceedings of the International Symposium On Artificial Intelligence, Robotics and Automation in Space*, pages 627–634, Noordwijk, The Netherlands, June 1999.

[70] S. Julier and J. K. Uhlmann. A counter example to the theory of simultaneous localization and map building. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4238–4243, Seoul, Korea, May 2001.

[71] N. Karlsson, E. D. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich. The vSLAM algorithm for robust localization and mapping. In *Proceedings of the International Conference of Robotics and Automation*, Barcelona, Spain, 2005.

[72] A. Kelly. Linearized error propagation in odometry. *International Journal of Robotics Research*, 23(2):179–218, Feb 2004.

[73] J. Kim and S. Sukkarieh. Autonomous airborne navigation in unknown terrain environments. *IEEE Transactions on Aerospace and Electronic Systems*, 40(3):1031–1045, July 2004.

[74] J. Kim and S. Sukkarieh. Real-time implementation of airborne inertial-SLAM. *Robotics and Autonomous Systems*, 55(1), Jan. 2007.

[75] K. Konolige, M. Agrawal, and J. Sola. Large-scale visual odometry for rough terrain. In *Proceedings of the International Symposium on Research in Robotics*, Hiroshima, Japan, Nov. 26-29 2007.

[76] D. Kumar and H. G. Tanner. How sensor graph topology affects localization accuracy. In *Proceedings of the European Control Conference*, pages 868–873, Kos, Greece, July 2-5 2007.

[77] R. Kurazume and S. Hirose. Study on cooperative positioning system: optimum moving strategies for CPS-III. In *Proceedings of the IEEE International Conference in Robotics and Automation*, pages 2896–2903, Leuven, Belgium, May 16-20 1998.

[78] R. Kurazume, S. Nagata, and S. Hirose. Cooperative positioning with multiple robots. In *Proceedings of the IEEE International Conference in Robotics and Automation*, pages 1250–1257, Los Alamitos, CA, May 8-13 1994.

[79] J. Langelaan and S. Rock. Passive GPS-free navigation for small UAVs. In *Proceedings of the IEEE Aerospace Conference*, pages 1–9, Big Sky, MT, Mar. 2005.

[80] J. W. Langelaan. *State estimation for autonomous flight in cluttered environments.* PhD thesis, Stanford University, Department of Aeronautics and Astronautics, 2006.

[81] H. Lee, K. Teo, and A. E. Lim. Sensor scheduling in continuous time. *Automatica*, 37(12):2017–2023, Dec. 2001.

[82] J. Leonard, R. Rikoski, P. Newman, and M. Bosse. Mapping partially observable features from multiple uncertain vantage points. *The International Journal of Robotics Research*, 21(10-11):943–975, 2002.

[83] D. Lerro and Y. Bar-Shalom. Tracking with debiased consistent converted measurements versus EKF. *IEEE Transactions on Aerospace and Electronic Systems*, 29(3):1015–1022, July 1993.

[84] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 260(2):91–110, Nov. 2004.

[85] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, Oct. 1997.

[86] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 674–679.

[87] M. D. Marco, A. Garulli, A. Giannitrapani, and A. Vicino. Simultaneous localization and map building for a team of cooperating robots: a set membership approach. *IEEE Transactions of Robotics and Automation*, 19(2):238–248, Dec. 2003.

[88] A. Martinelli. Improving the precision on multi robot localization by using a series of filters hierarchically distributed. In *Proceedings of the IEEE/RJS International Conference on Intelligent Robots and Systems*, pages 1053–1058, San Diego, CA, Oct. 29-Nov. 2 2007.

[89] E. B. Martinson and F. Dellaert. Marco polo localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1960–2345, Taipei, Taiwan, Sep. 14-19 2003.

[90] L. H. Matthies. *Dynamic stereo vision.* PhD thesis, School of Computer Science, Carnegie Mellon University, 1989.

[91] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141-1 of *Mathematics in Science and Engineering.* Academic Press, London, 1979.

[92] P. S. Maybeck. *Stochastic Models, Estimation and Control*, volume 141-2 of *Mathematics in Science and Engineering*. Academic Press, London, 1982.

[93] P. McLauchlan. The variable state dimension filter applied to surface-based structure from motion. Technical Report VSSP-TR-4/99, School of Electrical Engineering, Information Technology and Mathematics, University of Surrey, UK, 1999.

[94] L. Meier, J. Peschon, and R. M. Dressler. Optimal control of measurement subsystems. *IEEE Transactions on Automatic Control*, 12(5):528–536, Oct. 1967.

[95] U. Miekkala. Graph properties for splitting with grounded laplacian matrices. *BIT Numerical Mathematics*, 33(2):485–495, Sep. 1993.

[96] F. M. Mirzaei and S. I. Roumeliotis. A Kalman filter-based algorithm for IMU-camera calibration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2427–2434, San Diego, CA, Oct. 29 - Nov. 2 2007.

[97] B. Mohar. The laplacian spectrum of graphs. *Graph Theory, Combinatorics, and Applications*, 2:871–898, 1991.

[98] M. Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem with Unknown Data Association*. PhD thesis, Robotics Institute, Carnegie Mellon University, July 2003.

[99] J. Montiel, J. Civera, and A. Davison. Unified inverse depth parametrization for monocular SLAM. In *Proceedings of Robotics: Science and Systems*, pages 81–88, Philadelphia, PA, Aug. 16-19 2006.

[100] A. I. Mourikis and S. I. Roumeliotis. Analysis of positioning uncertainty in reconfigurable networks of heterogeneous mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 572–579, New Orleans, LA, Apr. 26 - May 1 2004.

[101] A. I. Mourikis and S. I. Roumeliotis. Analysis of positioning uncertainty in simultaneous localization and mapping (SLAM). In *Proceedings of the IEEE/RSJ International Conference on Robotics and Intelligent Systems (IROS)*, pages 13–20, Sendai, Japan, Sep. 28 - Oct. 2 2004.

[102] A. I. Mourikis and S. I. Roumeliotis. Performance bounds for cooperative simultaneous localization and mapping (C-SLAM). Technical report, Dept. of Computer Science and Engineering, University of Minnesota, 2004. http://www.cs.umn.edu/∼mourikis/TR_MultiSLAM.pdf.

[103] A. I. Mourikis and S. I. Roumeliotis. Optimal sensing strategies for mobile robot formations: Resource-constrained localization. In *Proceedings of Robotics: Science and Systems*, pages 281–288, Cambridge, MA, June 8-11 2005.

[104] A. I. Mourikis and S. I. Roumeliotis. Performance bounds for cooperative simultaneous localization and mapping (C-SLAM). In *Proceedings of Robotics: Science and Systems*, pages 73–80, Cambridge, MA, June 8-11 2005.

[105] A. I. Mourikis and S. I. Roumeliotis. Analytical characterization of the accuracy of SLAM without absolute orientation measurements. In *Proceedings of Robotics: Science and Systems*, pages 215–222, Philadelphia, PA, Aug. 2006.

[106] A. I. Mourikis and S. I. Roumeliotis. Optimal sensor scheduling for resource-constrained localization of mobile robot formations. *IEEE Transactions on Robotics*, 22(5):917–931, Oct. 2006.

[107] A. I. Mourikis and S. I. Roumeliotis. Performance analysis of multirobot cooperative localization. *IEEE Transactions on Robotics*, 22(4):666–681, Aug. 2006.

[108] A. I. Mourikis and S. I. Roumeliotis. Predicting the accuracy of Cooperative Simultaneous Localization and Mapping (C-SLAM). *International Journal of Robotics Research*, 25(12):1273–1286, Dec. 2006.

[109] A. I. Mourikis and S. I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3565–3572, Rome, Italy, Apr. 2007.

[110] A. I. Mourikis and S. I. Roumeliotis. A dual-layer estimator architecture for long-term localization. In *Proceedings of the Workshop on Visual Localization for Mobile Platforms*, Anchorage, AK, June 28 2008.

[111] A. I. Mourikis, S. I. Roumeliotis, and J. W. Burdick. SC-KF mobile robot localization: A stochastic cloning-Kalman filter for processing relative-state measurements. *IEEE Transactions on Robotics*, 23(3):717–730, Aug. 2007.

[112] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, D. M. Helmick, and L. Matthies. Autonomous stair climbing for tracked vehicles. *International Journal of Computer Vision & International Journal of Robotics Research - Joint Special Issue on Vision and Robotics*, 26(7):737–758, July 2007.

[113] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, and L. H. Matthies. Vision-aided inertial navigation for precise planetary landing: Analysis and experiments. In *Proceedings of Robotics: Science and Systems*, Atlanta, GA, June 26-30 2007.

[114] H. Mouritsen and B. J. Frost. Virtual migration in tethered flying monarch butterflies reveals their orientation mechanisms. *Proceedings of the National Academy of Sciences*, 99(15):10162–10166, July 2002.

[115] P. Moutarlier and R. Chatila. Stochastic multisensory data fusion for mobile robot location and environment modeling. In H. Miura and S. Arimoto, editors, *Fifth International Symposium of Robotics Research*, pages 85–94, Tokyo, Japan, 28-31 Aug. 1989.

[116] J. Neira and J. D. Tardos. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, Dec. 2001.

[117] E. W. Nettleton, P. W. Gibbens, and H. F. Durrant-Whyte. Closed form solutions to the multiple platform simultaneous localization and map building (SLAM) problem. In B. V. Dasarathy, editor, *Proceedings of the SPIE, Sensor Fusion: Architectures, Algorithms, and Applications IV*, volume 4051, pages 428–437, 2000.

[118] P. Newman. *On the structure and solution of the simultaneous localisation and map building problem.* PhD thesis, University of Sydney, Mar. 1999.

[119] P. Newman, J. Leonard, J. Tardos, and J. Neira. Explore and return: experimental validation of real-time concurrent mapping and localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1802–1809, Washington, DC, May 11-15 2002.

[120] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1):3–20, Jan. 2006.

[121] J. Oliensis. A new structure-from-motion ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):685–700, July 2000.

[122] C. F. Olson and L. H. Matthies. Maximum likelihood rover localization by matching range maps. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 272–277, Leuven, Belgium, 16-20 May 1998.

[123] C. F. Olson, L. H. Matthies, H. Schoppers, and M. W. Maimone. Robust stereo ego-motion for long distance navigation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 453–458, Hilton Head, SC, June 2000.

[124] C. F. Olson, L. H. Matthies, M. Schoppers, and M. W. Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215–229, June 2003.

[125] L. L. Ong, M. Ridley, J. H. Kim, E. Nettleton, and S. Sukkarieh. Six DoF decentralised SLAM. In *Proceedings of the Australasian Conference on Robotics and Automation*, pages 10–16, Brisbane, Australia, Dec. 2003.

[126] T. Oskiper, Z. Zhiwei, S. Samarasekera, and R. Kumar. Visual odometry system using multiple stereo cameras and inertial measurement unit. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, June 17-22 2007.

[127] C. Parker, H. Zhang, and C. R. Kube. Blind buldozing: Multiple robot nest construction. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2010–15, Las Vegas, NV, Oct. 27-31 2003.

[128] M. Paskin. *Thin Junction Tree Filters for Simultaneous Localization and Mapping.* PhD thesis, University of California, Berkeley, 2002.

[129] S. T. Pfister, K. L. Kriechbaum, S. I. Roumeliotis, and J. W. Burdick. Weighted range sensor matching algorithms for mobile robot displacement estimation. In *Proceedings*

of the IEEE International Conference on Robotics and Automation, pages 1667–1674, Washington D.C., May 11-15 2002.

[130] P. Pinies, T. Lupton, S. Sukkarieh, and J. Tardos. Inertial aiding of inverse depth SLAM using a monocular camera. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2797–2802, Rome, Italy, Apr. 2007.

[131] R. J. Prazenica, A. Watkins, and A. J. Kurdila. Vision-based Kalman filtering for aircraft state estimation and structure from motion. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, number AIAA 2005-6003, San Fransisco, CA, Aug. 15-18 2005.

[132] I. M. Rekleitis, G. Dudek, and E. Milios. Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence*, 31(1-4):7–40, Oct. 2001.

[133] I. M. Rekleitis, G. Dudek, and E. Milios. Multi-robot cooperative localization: A study of trade-offs between efficiency and accuracy. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2690–2695, Lausanne, Switzerland, Sep.30-Oct.4 2002.

[134] I. M. Rekleitis and S. I. Roumeliotis. Analytical expressions for positioning uncertainty propagation in networks of robots. In *Proceedings of the IEEE Mediterranean Conference on Control and Automation*, pages 131–136, Rhodes, Greece, 2003.

[135] B. Ronacher and R. Wehner. Desert ants cataglyphis fortis use self-induced optic flow to measure distances travelled. *Journal of Comparative Physiology A*, 177(1):71–75, July 1995.

[136] S. I. Roumeliotis and G. A. Bekey. SEGMENTS: A layered, dual-Kalman filter algorithm for indoor feature extraction. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 454–461, Takamatsu, Japan, Oct. 30 - Nov. 5 2000.

[137] S. I. Roumeliotis and G. A. Bekey. Distributed multirobot localization. *IEEE Transactions on Robotics and Automation*, 18(5):781–795, Oct. 2002.

[138] S. I. Roumeliotis and J. W. Burdick. Stochastic cloning: A generalized framework for processing relative state measurements. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1788–1795, Washington, DC, May 11-15 2002.

[139] S. I. Roumeliotis, A. E. Johnson, and J. F. Montgomery. Augmenting inertial navigation with image-based motion estimation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4326–33, Washington D.C, May 2002.

[140] S. I. Roumeliotis and I. M. Rekleitis. Analysis of multirobot localization uncertainty propagation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1763–1770, Las Vegas, NV, Oct. 27-31 2003.

[141] S. I. Roumeliotis and I. M. Rekleitis. Propagation of uncertainty in cooperative multi-robot localization: Analysis and experimental results. *Autonomous Robots*, 17(1):41–54, July 2004.

[142] A. C. Sanderson. A distributed algorithm for cooperative navigation among multiple mobile robots. *Advanced Robotics*, 12(4):335–349, 1998.

[143] A. V. Savkin, R. J. Evans, and E. Skafidas. The problem of optimal robust sensor scheduling. *Systems and Control Letters*, 43(2):149–157, June 2001.

[144] B. Schumitsch, S. Thrun, L. Guibas, and K. Olukotun. The identity management Kalman filter (IMKF). In *Proceedings of Robotics: Science and Systems*, pages 223–230, Philadelphia, PA, Aug. 2006.

[145] S. Se, D. G. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735–758, Aug. 2002.

[146] J. A. Simmons. Perception of echo phase information in bat sonar. *Science*, 204(4399):1336–1338, June 1979.

[147] E. Skafidas and A. Nerode. Optimal measurement scheduling in linear quadratic gaussian control problems. In *Proceedings of the IEEE International Conference on Control Applications*, pages 1225 – 1229, Trieste, Italy, Sept. 1998.

[148] R. C. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Proceedings of the Workshop on Uncertainty in Artificial Intelligence (AAAI)*, pages 1–21, Philadelphia, PA, Aug. 1986.

[149] R. C. Smith, M. Self, and P. Cheeseman. *Autonomous Robot Vehicles*, chapter Estimating Uncertain Spatial Relationships in Robotics, pages 167–193. Springer-Verlag, 1990.

[150] S. Soatto, R. Frezza, and P. Perona. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3):393–413, Mar. 1996.

[151] S. Soatto and P. Perona. Recursive 3-D visual motion estimation using subspace constraints. *International Journal of Computer Vision*, 22(3):235–259, 1997.

[152] J. Spletzer, A. Das, R. Fierro, C. Taylor, V. Kumar, and J. Ostrowski. Cooperative localization and control for multi-robot manipulation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 631 – 636, Wailea, HI, Nov. 2001.

[153] D. Strelow. *Motion estimation from image and inertial measurements*. PhD thesis, Carnegie Mellon University, Nov. 2004.

[154] E. Stump, B. Grocholsky, and V. Kumar. Extensive representations and algorithms for nonlinear filtering and estimation. In *Proceedings of International Workshop on the Algorithmic Foundations of Robotics*, New York, NY, July 16-18 2006.

[155] H. G. Tanner, G. J. Pappas, and V. Kumar. Leader-to-formation stability. *IEEE Transactions on Robotics and Automation*, 20(3):433–455, June 2004.

[156] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. The MIT Press, Cambridge, MA, 2005.

[157] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, 23(7-8):693–716, Aug. 2004.

[158] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley, the robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692, June 2006.

[159] N. Trawny, A. I. Mourikis, S. I. Roumeliotis, A. E. Johnson, and J. Montgomery. Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks. *Journal of Field Robotics*, 24(5):357–378, May 2007.

[160] N. Trawny and S. I. Roumeliotis. Indirect Kalman filter for 3D pose estimation. Technical Report 2005-002, Dept. of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, Mar. 2005.

[161] N. Trawny and S. I. Roumeliotis. A unified framework for nearby and distant landmarks in bearing-only SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1923–1929, Orlando, FL, May 2006.

[162] B. Triggs, P. McLauchlan, R. Hartley, and Fitzgibbon. Bundle adjustment – a modern synthesis. In *Vision Algorithms: Theory and Practice*, pages 298–375. Springer Verlag, 2000.

[163] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, J. Dolan, D. Duggins, D. Ferguson, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. Howard, A. Kelly, D. Kohanbash, M. Likhachev, N. Miller, K. Peterson, R. Rajkumar, P. Rybski, B. Salesky, S. Scherer, Y. Woo-Seo, R. Simmons, S. Singh, J. Snider, A. Stentz, W. R. Whittaker, J. Ziglar, H. Bae, B. Litkouhi, J. Nickolaou, V. Sadekar, S. Zeng, J. Struble, M. Taylor, and M. Darms. Tartan Racing: A multi-modal approach to the DARPA Urban Challenge. Technical report, Defense Advanced Research Projects Agency (DARPA), Apr. 2007.

[164] A. Vedaldi, G. Guidi, and S. Soatto. Moving forward in structure from motion. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–7, June 17-22 2007.

[165] S. Venkataramanan and A. Dogan. Nonlinear control for reconfiguration of UAV formation. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Austin, TX, 2003.

[166] R. Volpe. Mars rover navigation results using sun sensor heading determination. In *Proceedings of the IEEE/RSJ International Conference on Robotics and Intelligent Systems*, pages 469–467, Kyingju, Korea, Oct. 17-21 1999.

[167] Z. Wang, Y. Hirata, and K. Kosuge. Control a rigid caging formation for cooperative object transportation by multiple mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1580–1585, New Orleans, LA, Apr. 26 - May 1 2004.

[168] J. Wawerla, G. S. Sukhatme, and M. J. Mataric. Collective construction with multiple robots. In *Proceedings of the IEEE/RSJ International Conference on Robotics and Intelligent Systems*, pages 2696 – 2701, Lausanne, Switzerland, Sep. 30 - Oct. 4 2002.

[169] Y. Xiong and L. Matthies. Vision-guided autonomous stair climbing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1842–1847, San Francisco, CA, Apr. 2000.

[170] A. Yamashita, M. Fukuchi, J. Ota, T. Arai, and H. Asama. Motion planning for cooperative transportation of a large object by multiple mobile robots in a 3D environment. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3144–51, San Francisco, CA, Apr. 24-28 2000.

[171] F. Zhang, B. Grocholsky, and V. Kumar. Formations for localization of robot networks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3369–3374, New Orleans, LA, Apr. 26 - May 1 2004.