Cooperative Multi-Robot Belief Space Planning for Visual-Inertial Navigation and Online Sensor Calibration

Yair Ben-Elisha

Cooperative Multi-Robot Belief Space Planning for Visual-Inertial Navigation and Online Sensor Calibration

Research Thesis

Submitted in partial fulfillment of the requirements
for the degree of Master of Science

Yair Ben-Elisha

Submitted to the Senate
of the Technion — Israel Institute of Technology
Tishrei 5778 Haifa September 2017

This research was carried out under the supervision of Asst. Prof. Vadim Indelman from the Faculty of Aerospace Engineering at the Technion - Israel Institute of Technology.

List of Publications

Some results in this thesis have been published as articles by the author and research collaborators in conferences and journals during the course of the author's research period, the most up-to-date versions of which being:

[1] Y. Ben-Elisha and V. Indelman, Active online visual-inertial navigation and sensor calibration via belief space planning and factor graph based incremental smoothing. In IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), 2017.

[2] Y. Ben-Elisha and V. Indelman, Active online self-calibration and accurate navigation via belief space planning and factor graph based incremental smoothing. In 57th Israel Annual Conference on Aerospace Sciences (IACAS), 2017.

Acknowledgements

I would like to hugely thank my advisor Prof. Vadim Indelman for all his guidance, help, support and patience throughout my research. Thank you for the detailed explanations and the high-standard demands.

For my great family, thank you for the encouragement and patience. Special thanks to my wife Moran for the unlimited support.

Contents

List of Figures

Abstract

High accuracy navigation in GPS-deprived uncertain environments is of prime importance to various robotics applications. In such scenarios, it has been recently shown that online sensor calibration and multi-robot collaboration, whereby robots make mutual observations of the environment or perform relative observations of each other, can significantly enhance navigation accuracy. However, these approaches typically consider a passive setting, where robot actions are externally determined. On the other hand, belief space planning (BSP) approaches account for different sources of uncertainty, thus identifying actions that improve certain aspects in inference, such as accuracy. Yet, existing BSP approaches typically do not consider sensor calibration in the mentioned problem setting, nor a visual-inertial SLAM setup.

In this research, we contribute single-robot and multi-robot BSP approaches for active sensor calibration considering a visual-inertial SLAM system. To that end, we maintain a belief over both robot's pose and sensor calibration, and reason how that belief would evolve for different actions while considering partially unknown and uncertain environments. In particular, we leverage the recently developed concept of IMU pre-integration and develop appropriate factor graph formulation for future beliefs to facilitate computationally efficient inference within BSP.

Another key aspect of our approach are indirect multi-robot observation updates given the states of different robots are correlated. This concept allows for a subset of robots to carry on their individual (possibly time-critical) tasks while preserving high accuracy estimation by relying on other expendable robots to make appropriate observations of the environment. We study our approach in high-fidelity synthetic simulation and show the determined actions can lead to significantly improved estimation accuracy.

Abbreviations and Notations

ANPL	:	Autonomous Navigation and Perception Lab
BSP	:	Belief Space Planning
DOF	:	Degree Of Freedom
FG	:	Factor Graph
GPS	:	Global Positioning System
IMU	:	Inertial Measurement Unit
INS	:	Inertial Navigation System
iSAM	:	Incremental Smoothing And Mapping
MAP	:	Maximum A Posteriori
MR	:	Multi Robot
RMSE	:	Root Mean Square Error
SFM	:	Structure From Motion
SLAM	:	Simultaneous Localization And Mapping
PDF	:	Probability Density Function
PRM	:	Probabilistic Road-map
POMDP	:	Partially Observable Markov Decision Process
RRT	:	Rapidly-exploring Random Tree
VINS	:	Vision-Inertial Navigation System

Chapter 1

Introduction

Autonomous navigation in unknown or uncertain environments has been extensively investigated over the past two decades with numerous applications in robotics, including aerial GPS-denied navigation, indoor navigation and autonomous driving. Highly accurate online navigation in these, and many other, applications is of prime importance. Modern navigation systems no longer rely solely on inertial measurement units that are suspect to drift and on GPS that may be unreliable or unavailable, but calculate the navigation solution by fusing measurements captured by different on-board sensors (e.g. camera, laser sensors). When the environment is unknown or uncertain, robot localization (and navigation) involves also mapping the environment, a problem known in the navigation context as visual-inertial SLAM.

The corresponding inference problem involves tracking the probability density function (pdf) over variables of interest given available information. These variables often include navigation state, landmarks representing the mapped environment thus far and sensor calibration parameters. The latter can represent extrinsic calibration, such as relative pose between different sensors, and also intrinsic parameters such as IMU bias and camera focal length. While some calibration parameters can be recovered in an offline fashion, due to stochasticity, some sensors (e.g. IMU) require also online calibration, without which navigation accuracy will be compromised. Online IMU calibration, however, is traditionally done considering GPS availability with pre-defined trajectories (i.e. actions) that were specifically calculated for such a setting.

Yet, actual performance depends, among other factors, on robot actions - different robot actions can often result in different estimation accuracies, especially in lack of external information (such as GPS). Thus, attaining high-accuracy navigation involves deep intertwining between inference and planning, requiring the latter to account for different sources of uncertainty. The corresponding planning approaches are known as belief space planning (BSP) methods, which have received considerable attention in recent years in the context of autonomous navigation in known, and more recently, unknown environments. However, existing BSP approaches typically do not consider sensor calibration aspects in the context of visual-inertial SLAM, while hard-coded actions that were suitable for GPS setting may perform poorly in scenarios considered herein.

Moreover, we leverage recent work that addressed visual-inertial SLAM using factor graphs

and incremental smoothing [20], and use these techniques also within belief space planning. Finally, within this framework we extend the recently-developed concept of IMU pre-integration [31], that was used thus far only for information fusion and visual-inertial SLAM [20], to BSP so that longer planning horizons can be efficiently considered in presence of high-rate IMU measurements.

In addition, an extension of the BSP problem to a multi-robot framework has been investigated over the last few years. Collaboration between multiple robots can significantly improves performance of both inference and planning phases. In particular, by sharing relevant information between robots, estimation quality can substantially improve, while by appropriately coordinating actions the robots can often finish a task in a shorter time. However, existing multi-robot BSP approaches consider that the cooperation between the robots is done by observing mutual scenes (even at different times) or by observing the other robot. In addition, recent planning approaches consider an objective function, used for calculating the optimal paths, where each robot aims to improve its estimation accuracy only.

In this research we develop two main approaches: (a) single-robot BSP approach for active sensor calibration and accurate autonomous navigation considering a visual-inertial SLAM setting, and (b) extension of the single-robot BSP approach to a multi-robot framework for efficient active cooperation between the robots.

The first approach is capable of calculating optimal actions for reducing estimation error within inference, reducing estimation error growth rate via IMU sensor calibration, or a combination of both. Additional typical costs in the objective function, such as reaching a goal and control effort, are naturally supported as well.

The second approach further advances the state of the art by defining two collaborative robots subgroups, the "strategic group" and the "expendable group". The key idea is to have robots from the "expendable robots" improve performance, in terms of estimation accuracy, of robots from the "strategic group" while the latter execute time-critical missions. Robots from the former group are envisioned to be cheap and therefore expandable, with their sole purpose to boost the performance of the "strategic group". In particular, the objective function for the "expendable group" aims to improve the "strategic group's" estimation accuracy. This approach leverages the concept of indirect multi-robot updates that we introduce herein, according to which, in presence of prior correlation between states of different robots, an informative observation captured by one robot impacts also other robots that can be in different regions of the environments. Leveraging this concept, the expandable robots in our multi-robot BSP approach have enhanced flexibility to aid the strategic robots by traveling to areas where informative observations can (or likely to) be captured, without constraining the trajectories of the strategic robots.

1.1 Related Work

Traditional inertial navigation systems are based on the strapdown mechanism [40, 8], in which IMU measurements are integrated into a navigation solution. Typically, navigation aiding meth-

ods apply filtering approaches for fusing measurements from other available sensors with the inertial solution. More recent approaches for information fusion in inertial navigation systems are calculating the optimal solution based on a non-linear optimization involving all the unknown variables (and using all the available measurements). These approaches, also directly related to bundle adjustment (BA), are commonly used in the robotics community for solving the full simultaneous localization and mapping (SLAM) problem (see e.g. [3, 7]). In particular, [7] represents the posterior probability density function (pdf) using the factor graph graphical model [28], which naturally encodes the inherent sparsity of the underlying (square root) information matrices. Kaess et al. [25, 24] develop incremental smoothing and mapping (iSAM) approaches that, utilizing graphical models, allow to efficiently update the posterior with each new incoming measurement. Indelman et al. [18] introduced a related concept in the context of multi-robot collaborative navigation, however, considering the covariance form and not the information form as in [25, 24].

While iSAM is a general approach for efficient maximum a posteriori (MAP) inference given available measurements, possibly from different sensors, recent work [19, 20] examined its specific use for inertial navigation systems in presence of high rate sensors such as IMU. The recently developed concept of IMU pre-integration [31, 20, 9] allows to avoid the issue of adding factor nodes to the graph at IMU rate, by summarizing consecutive IMU measurements, and incorporating into the factor graph only the summarized quantities while still supporting real time performance (see [20] for details).

While the above approaches focus on inference (e.g. estimating robot poses) given actions, different actions can lead to substantially different estimation performance, for example, in terms of accuracy. The corresponding problem of finding an optimal action(s) is an instantiation of a partially observable Markov decision process (POMDP) problem, which is computationally intractable [34]. Numerous approaches that trade-off optimality with computational complexity have been developed in recent years. These approaches are often known as belief space planning (BSP). These approaches can be segmented into several categories: point-based value iteration methods [29, 35], simulation based approaches [39, 41], sampling based approaches [12, 13, 38] and direct trajectory optimization approaches [16, 17, 42].

Yet, only few BSP approaches consider sensor calibration aspects [43, 11, 33]. For example, Hausman et al. [11] consider IMU calibration (in terms of bias), optimizing a nominal trajectory for self-calibration. However, that approach assumes GPS availability, in contrast to the problem setup considered herein. Webb et al. [43] develop a method for active calibration of extrinsic system parameters using continuous POMDP. Active calibration of extrinsic system parameters (e.g. transformation between frames) is also considered in [33]. However, neither of these works considers visual-inertial navigation systems, nor active calibration of intrinsic parameters such as IMU bias in unknown GPS-deprived environment.

A multi-robot belief space framework has been also investigated in recent years. Developed approaches focus on different aspects, including multi-robot tracking, active SLAM and autonomous navigation in unknown environments and informative planning (see e.g. [30, 2, 14, 13, 36, 21]). In particular, existing approaches [38, 5, 18] typically focus on the inference part, con-

sidering robot actions to be determined externally. Active multi-robot approaches [4] typically focus on the trade-off between exploring new regions and reducing uncertainty by re-observing previously mapped areas. Recent works, [27, 13, 14, 21], consider the problem of collaborative multi-robot BSP while operating in unknown environment. In particular, [13, 21] introduced within the belief reasoning regarding future mutual observations of environments that are unknown at planning time. However, these multi-robot BSP approaches consider that the cooperation between the robots is done by observing mutual scenes (even at different times) or by observing the other robot. Additionally, these approaches typically consider an objective function where each robot aims to improve its own estimation accuracy. In contrast, we consider an objective function for the "expendable group" that aims to improve the "strategic group's" estimation accuracy. In addition, we introduce a concept of indirect multi-robot updates, that in presence of prior correlation between states of different robots, an informative observation captured by one robot impacts also other robots that can be in different regions of the environments.

1.2 Contributions

In this research we develop a BSP approach for active sensor calibration and accurate autonomous navigation considering a visual-inertial SLAM setting. Our approach is capable of calculating optimal actions for reducing estimation error within inference, reducing estimation error growth rate via IMU sensor calibration, or a combination of both. Additional typical costs in the objective function, such as reaching a goal and control effort, are naturally supported as well.

We also contribute a collaborative multi-robot approach for expendable indirect multi-robot cooperation given the states of different robots are correlated. This concept allows for a subset of robots to carry on their individual (possibly time-critical) tasks while preserving high accuracy estimation by relying on other expendable robots to make appropriate observations of the environment and indirectly update the other robots state estimation.

The combination of these two approaches improve the strategic robots performance, in terms of reducing estimation error, error growth rate and time for reaching the goal, by relying on the expendable robots that focus on updating the estimation accuracy of the former group.

In addition, we leverage recent work that addressed visual-inertial SLAM using factor graphs and incremental smoothing [20], and use these techniques also within belief space planning. Finally, within this framework we extend the recently-developed concept of IMU pre-integration [31], that was used thus far only for information fusion and visual-inertial SLAM [20], to BSP so that longer planning horizons can be efficiently considered in presence of high-rate IMU measurements.

Chapter 2

Background

2.1 Inertial Navigation

Inertial navigation is a dead reckoning system that computes the platform position, velocity and attitude by integrating the linear acceleration and angular rate measured by the Inertial Measurement Unit (IMU). Every dead reckoning system requires an initialization stage, the initial navigation solution, that mostly achieved by an external source, such as a human interface, GPS or a more accurate Inertial Navigation System (INS).

The inertial navigation is based on the high rate measurements from the IMU sensor, angular rate measurements of the gyroscopes and linear acceleration measurements of the accelerometers. The navigation solution (position, velocity and attitude) of the platform is calculated by using the "Strapdown Equations". The main principles for the definition of the strapdown equations are:

- Determination of the angular motion of a vehicle using gyroscopic sensors, from which its attitude relative to a reference frame may be derived;
- Measure specific force using accelerometers;
- Resolve the specific force measurements into the reference frame using the knowledge of attitude derived from the information provided by the gyroscopes;
- Evaluate the force resulting from the gravitational field - the gravitational attraction of the Earth in the case of systems operating in the vicinity of the Earth;
- Integrate the resolved specific force measurements to obtain estimates of the velocity and position of the vehicle.

Unfortunately, all the inertial sensors are subject to errors which limit the accuracy to which the angle of rotation or applied accelerate can be measured. These errors are propagating through time and can be modeled using a complex error model, usually provided by the sensor's manufacturer. In this work we assume a simple IMU error estimation vector as,

$$c^{IMU} = \begin{bmatrix} d^T & b^T \end{bmatrix}^T \tag{2.1}$$

where b and d are the accelerometers and the gyroscopes errors respectively, usually referred to as bias and drift.

More information about inertial navigation and technology can be found in [40].

2.2 Probabilistic Inference

Throughout this work, we use Bayesian inference to recover the posterior probability distribution function over variables of interest, such as robot poses, sensors calibration and landmark locations, based on available information such as features from camera-captured images and IMU measurements. In this section, however, we first provide the necessary background in probabilistic inference considering the random variables X and Y.

The joint probability density function (pdf) of the variables X and Y is defined as $p(x, y)$ and, if the two variables are independent, is given by,

$$p(x, y) = p(x)p(y) \tag{2.2}$$

Variables often carry information about other variables, in which case, using Bayes rule, this conditional probability is written as,

$$p\left(x|y\right) = \frac{p\left(x, y\right)}{p\left(y\right)} = \frac{p\left(y|x\right)p\left(x\right)}{p\left(y\right)}. \tag{2.3}$$

In probabilistic robotics, Bayes rule allows to compute the conditional probability $p(x|y)$, called the posterior probability distribution, as a function of the inverse conditional probability $p(y|x)$ and the prior $p(x)$. Using the fact that $p(y)$ is being independent of x, it can be considered as constant and thus, Eq. (2.3) can be written as,

$$p(x|y) = \eta p(y|x)p(x) \ \ , \ \ \eta \doteq p(y)^{-1} \tag{2.4}$$

Given the conditional probability, the calculation of the maximum a posteriori (MAP) estimate x^* is define as,

$$x^* = \underset{x}{argmax}\, p\left(x|y\right) \tag{2.5}$$

For our case, it is convenient to assume that the joint pdf can be parametrized by a Gaussian distribution

$$p\left(x|y\right) \sim N\left(\hat{X}, \Lambda\right) \tag{2.6}$$

where Λ is the information matrix (inverse of the covariance matrix).

The maximum a posteriori (MAP) estimate of X is then given by

$$\hat{X}^* = \underset{x}{argmin} \ -log\left\{p\left(x|y\right)\right\} \tag{2.7}$$

This optimization problem lies at the core of the inner inference layer of our planning ap-

proach. In principle, solving (2.7) involves iterative nonlinear optimization. A standard way to solve the minimization problem is the Gauss-Newton method, where a single iteration involves linearizing the above equation about the current estimate $\bar{X}_k$, calculating the delta vector ΔX_k and updating the estimate $\bar{X}_k \leftarrow \bar{X}_k + \Delta X_k$. This process should be repeated until convergence. See [7, 24, 16] for more information.

2.3 Factor Graph

Factor graphs are graphical models that are well suited to modeling complex estimation problems, such as Simultaneous Localization and Mapping (SLAM) or Structure from Motion (SFM) [28, 7]. A factor graph is a bipartite graph, $G_k = (\mathcal{F}_k, \mathcal{V}_k, \mathcal{E}_k)$ with two types of nodes: factor nodes $f_i \in \mathcal{F}_k$ and variable nodes $v_j \in \mathcal{V}_k \equiv \Theta_k$. Edges $e_{ij} \in \mathcal{E}_k$ can exist only between factor nodes and variable nodes, and are present if and only if the factor f_i involves a variable v_j.

The variables nodes represent the unknown random variables, while edges represent the constrains between appropriate variables, constraints that correspond to motion and measurement models and to prior knowledge. Each such constraint is called a factor. Each factor represents an individual term in the factorization of a joint pdf $p(\Theta)$, and therefore one can write,

$$p\left(\Theta_k\right) \propto \prod_i f_i\left(\mathcal{V}_k^i\right), \tag{2.8}$$

where $\mathcal{V}_k^i$ represents an appropriate subset of variable nodes ($\mathcal{V}_k^i \subseteq \mathcal{V}_k$) involved in the factor f_i, such as motion models, measurement models and priors. See illustration in Figure 2.1 and [7] for further details.

For a multi-robot scenario we can extend the factor graph definition in Eq. (2.8) to the following definition,

$$p\left(\Theta_k\right) \propto \prod_r \prod_i f_i^r\left(\mathcal{V}_k^{r,i}\right), \tag{2.9}$$

where $\mathcal{V}_k^{r,i}$ represents an appropriate subset of variable nodes ($\mathcal{V}_k^{r,i} \subseteq \mathcal{V}_k^r$) of robot r.

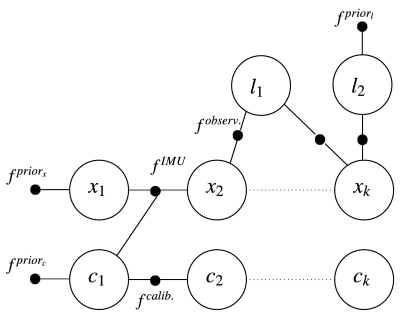


Figure 2.1: Example for a general factor graph for a problem with IMU and camera measurements

The factors f_i represent error functions that should be minimized. The explicit expression of such a function depends on the specific term in the factorization (2.8) that is represented by the factor f_i. Denoting this error function by $err\left(\mathcal{V}_k^i, z_i\right)$, the factor f_i is defined as

$$f_i\left(\mathcal{V}_k^i\right) = d\left(err\left(\mathcal{V}_k^i, z_i\right)\right) \tag{2.10}$$

where the operator $d(.)$ denotes a certain cost function. For Gaussian noise distributions, the general factor f_i (2.10) assumes the following form:

$$f_i\left(\mathcal{V}_k^i\right) = exp\left(-\frac{1}{2}\left\|err_i\left(\mathcal{V}_k^i, z_i\right)\right\|_{\Sigma_i}^2\right), \tag{2.11}$$

which defines the cost function $d(.)$, and calculating the MAP estimate $\mathcal{V}_k^* \doteq \underset{\mathcal{V}_k}{argmax}(\mathcal{V}_k|Z_k)$ becomes equivalent to performing inference over the factor graph and minimizing the following nonlinear least-squares function:

$$\sum_i \left\|err_i\left(\mathcal{V}_k^i, z_i\right)\right\|_{\Sigma_i}^2 \tag{2.12}$$

Here $\|a\|_{\Sigma_i}^2 \doteq a^T\Sigma^{-1}a$ is the squared Mahalanobis distance and Σ is the covariance matrix.

While, in general, this can be an expensive operation, it has been shown in previous works (see e.g. [20]) that using incremental smoothing, a recently developed approach [24] in the SLAM community, the involved computational complexity is small and high-rate performance is possible in typical navigation applications.

2.4 Pre-Integrated IMU

Incorporating high-rate IMU measurements into the posterior, Eq. (2.4), and the underlying inference, Eq. (2.5), which involves solving a nonlinear least squares (NLS) problem, does not provide real time performance in general. To avoid doing so, we adopt a recently developed concept of IMU pre-integration [31], and its application to sensor fusion in modern navigation systems [20].

The key idea is to integrate consecutive IMU measurements $\left(z_i^{IMU}, \ldots, z_j^{IMU}\right)$, between two lower-frequency measurements from other sensors, such as cameras, obtained at some times t_i and t_j, while expressing the resulting relative motion $\Delta x_{i\to j}$ in the body frame of x_i. Importantly, such a concept supports re-linearization, as part of the NLS iterative optimization, without recalculating $\Delta x_{i\to j}$ [31, 20]. Moreover, real time performance can be achieved by using intermediate $\Delta_{i\to t}$ with $t \leq j$.

We assume the starting pre-integration time is t_i and use the notations $\Delta p_{i\to j}^{b_i}, \Delta v_{i\to j}^{b_i}, R_{b_j}^{b_i}$ to represent the position, velocity and orientation components, respectively, calculated by preintegrating the IMU measurements from time t_i to some time t_j. In order to avoid recalculating these components when re-linearizing, Lupton and Sukkarieh [31] perform the integration in the body frame of the starting pre-integration time t_i, rather than in the global frame. The body frame at t_i is denoted by b_i. Finally, R_a^b represents a rotation from system a to system b.

Let $\Delta x_{i\to j} \doteq \left\{ \Delta p_{i\to j}^{b_i}, \Delta v_{i\to j}^{b_i}, R_{b_j}^{b_i} \right\}$.

Given the previous pre-integrated components, $\Delta p_{i\to j}^{b_i}, \Delta v_{i\to j}^{b_i}, R_{b_j}^{b_i}$, with $t_j \geq t_i$, and the calibration parameters at the starting time t_i, denoted by c_i, the equations for adding a new IMU measurement at time $t_{j+1} \doteq t_j + \Delta t$, comprising the specific force f_j and the angular velocity ω_j, are given in the following equations,

$$\begin{aligned} \Delta p_{i\to j+1}^{b_i} &= \Delta p_{i\to j}^{b_i} + \Delta v_{i\to j}^{b_i} \Delta t \\ \Delta v_{i\to j+1}^{b_i} &= \Delta v_{i\to j}^{b_i} + R_{b_j}^{b_i} f_i \Delta t \\ R_{b_{j+1}}^{b_i} &= R_{b_j}^{b_i} Expmap\left(\omega_j\right) \end{aligned} \tag{2.13}$$

and we get $\Delta x_{i\to j+1} \doteq \left\{ \Delta p_{i\to j+1}^{b_i}, \Delta v_{i\to j+1}^{b_i}, R_{b_{j+1}}^{b_i} \right\}$.

To summarize, we replace multiple IMU terms in the posterior, Eq. (2.4), between any two time instances t_i and t_j with $p\left(x_j | x_i, \Delta x_{i\to j}, c_i\right)$.

Chapter 3

Problem Formulation and Notations

Let $x_i^r = \begin{bmatrix} p_i^r & v_i^r & q_i^r \end{bmatrix}$ and $c_i^r = \begin{bmatrix} d_i^r & b_i^r \end{bmatrix}$ represent the navigation state vector and sensors calibration state vector or robot r, respectively, at time t_i. Here, p_i^r, v_i^r and q_i^r are the position, velocity and orientation (represented as a quaternion) of the body in the world frame and d_i^r, b_i^r are the gyroscope drift and accelerometers bias. Similarly we can add to the calibration state vector of other sensors parameters, e.g. camera calibration parameters. Also denote by L_i^r the perceived environment by that robot, e.g. in terms of 3D points, by time t_i.

We define the joint state

$$\Theta_k^r \doteq \{X_k^r, C_k^r, L_k^r\} \tag{3.1}$$

where $X_k^r \doteq \{x_0^r, \ldots, x_k^r\}$ and $C_k^r \doteq \{c_0^r, \ldots, c_k^r\}$ represent, respectively, navigation and calibration states up to time t_k.

Similarly we define $Z_{0:k}^r \doteq \{z_0^r, \ldots, z_k^r\}$ and $U_{0:k}^r \doteq \{u_0^r, \ldots, u_k^r\}$, where u_i^r and z_i^r represent control and obtained measurements of robot r at time t_i. In the following, we shall use the notation $z_i^{r,IMU} \in Z_i^r$ to denote IMU measurements and $z_i^{r,cam.} \in Z_i^r$ to denote camera observations.

We consider now a group of R collaborating robots, and denote by Θ_k the corresponding joint state

$$\Theta_k \doteq X_k \cup C_k \cup L_k \tag{3.2}$$

where $X_k \doteq \{X_k^r\}_{r=1}^R$, $C_k \doteq \{C_k^r\}_{r=1}^R$ and L_k represents the perceived environment by the entire group. Assuming a common reference frame between the robots is established, L_k includes all the 3D points in L_k^r for each r, expressed in that reference frame.

Given these definitions, the probability density function (pdf) at time t_k, also referred as the belief, is given by

$$b\left(\Theta_k\right) \doteq p\left(\Theta_k | Z_{0:k}, U_{0:k-1}\right) \propto \prod_{r=1}^{R} p\left(\Theta_k^r | Z_{0:k}^r, U_{0:k-1}\right), \tag{3.3}$$

from which the maximum a posteriori estimate

$$\Theta^{\star} = \underset{\Theta_k}{argmax}\, b\left(\Theta_k\right), \tag{3.4}$$

can be efficiently calculated while exploiting sparsity and re-using calculations (see e.g. [24]).

The belief (3.3) can be explicitly written in terms of individual motion, calibration and observation models as (see e.g. [16, 13])

$$b\left(\Theta_k\right) \propto \prod_{r=1}^{R} \left\{ priors^r \prod_{i=1}^{k} p\left(x_i^r | x_{i-1}^r, c_{i-1}^r, z_{i-1}^{r,IMU}\right) \cdot p\left(c_i^r | c_{i-1}^r\right) p\left(z_i^{r,cam.} | \Theta_i^{r,o}\right) \right\} \tag{3.5}$$

3.1 Motion Model

The probabilistic motion model, $p\left(x_i^r | x_{i-1}^r, c_{i-1}^r, z_{i-1}^{r,IMU}\right)$, is represented by the underlying nonlinear inertial navigation equations and the captured IMU measurements z^{IMU}

$$x_{i+1}^r = f\left(x_i^r, c_i^r, z_i^{r,IMU}\right) + w_i^r \;\;,\;\; w_i^r \sim N\left(0, \Sigma_w^r\right), \tag{3.6}$$

where w_i^r is an additive Gaussian noise with noise covariance matrix Σ_w^r, such that

$$p\left(x_i^r | x_{i-1}^r, c_{i-1}^r, z_{i-1}^{r,IMU}\right) \propto \exp\left[-\frac{1}{2}\| x_i^r - f\left(x_{i-1}^r, c_{i-1}^r, z_{i-1}^{r,IMU}\right)\|_{\Sigma_w^r}^2\right] \tag{3.7}$$

.

The function $f\left(x_i^r, c_i^r, z_i^{r,IMU}\right)$ is governed by the common differential strapdown equations (see e.g. [8, 20]). Assuming some general frame a (such as inertial frame, tangent frame, etc.) and denoting by b and i the body and inertial frames, the time derivative of the velocity v, expressed in frame a, is given by

$$\begin{aligned}
\dot{p}^a &= v^a \\
\dot{v}^a &= R_b^a f^b + g^a - 2\Omega_{ia}^a v^a - \left(\Omega_{ia}^a \Omega_{ia}^a + \dot{\Omega}_{ia}^a\right) p^a \\
\dot{R}_b^a &= R_b^a \Omega_{ab}^b \\
\Omega_{ia}^a &= [\omega_{ia}^a]_{\times} \;\;,\;\; \Omega_{ab}^b = \left[\omega^b\right]_{\times},
\end{aligned} \tag{3.8}$$

where R_b^a is the rotation matrix transforming from body frame to frame a, f^b is the specific force measured by the accelerometers and p^a is the position vector. The vector g^a is the position-dependent gravity acceleration.

The matrix Ω^a_{ia} is defined as

$$\Omega^a_{ia} = [\omega^a_{ia}]_\times \doteq \begin{bmatrix} 0 & -(\omega^a_{ia})_3 & (\omega^a_{ia})_2 \\ (\omega^a_{ia})_3 & 0 & -(\omega^a_{ia})_1 \\ -(\omega^a_{ia})_2 & (\omega^a_{ia})_1 & 0 \end{bmatrix} \tag{3.9}$$

with ω^a_{ia} being the rotational rate of frame a with respect to the inertial frame i, expressed in frame a, and $[.]_\times$ is the skew-symmetric operator, defined for any two vectors q_1 and q_2 as $[q_1]_\times q_2 = q_1 \times q_2$.

Similarly, $\Omega^b_{ab} = \left[\omega^b\right]_\times$ with ω^b denoting the rotation rate measured by the gyroscopes.

The IMU measurements errors Δf^b and $\Delta\omega^b$ include both random walk processes terms, n^g and n^a, and instrumentation calibration factors, d_{gyro} and $b_{acc.}$. The instrument calibration factors can be estimated and compensated as we show in our work. Given the measurements $f^b_{meas.}$, $\omega^b_{meas.}$ and estimation of the calibration factors d_{gyro}, $b_{acc.}$, the specific force and angular rate vectors for use in the navigation equations (3.8) are computed as

$$f^b = f^b_{meas.} - b_{acc.} \tag{3.10}$$

$$\omega^b = \omega^b_{meas.} - d_{gyro} \tag{3.11}$$

Observe that the true (noise free) acceleration and angular velocity of the robot are a function of the controls u^r_i; hence, we refer to the above IMU model as motion model.

In our work, we adopt a simplified model, commonly used in robotics [20, 11], called simple Euler integration prediction function. We could use different numerical schemes, ranging from a simple Euler integration to high-order Runge–Kutta integration, for solving these navigation equations. However, the factor graph framework allows the adoption of a simple Euler integration prediction function with an associated integration uncertainty.

3.2 Calibration Model

The IMU calibration parameters represented by $c = [b_{acc.}, d_{gyro}]$ are used for correcting the IMU measurement f^b, ω^b according to the assumed IMU error model. This model of IMU errors is usually estimated in conjunction with the estimation of the navigation state.

In the general case, the time propagation of c can be described according to some non-linear model of its own (e.g. random walk):

$$\dot{c} = g_c(c) \tag{3.12}$$

The calibration model $p\left(c^r_k | c^r_{k-1}\right)$ is assumed to be with additive Gaussian noise with the process noise covariance matrix Σ^r_e, i.e.

$$c^r_{i+1} = g(c^r_i) + e^r_i \ , \ e^r_i \sim N(0, \Sigma^r_e) \, , \tag{3.13}$$

In this work we assume a random constant model for the calibration model,

$$c_{i+1}^r = g\left(c_i^r\right) + e_i^r = c_i^r + e_i^r \tag{3.14}$$

If desired, more sophisticated schemes can be used as well.

3.3 Observation Model

The observation model $p\left(z_k^{r,L}|\Theta_k^{r,o}\right)$ at any time t_k can be written as

$$p\left(z_k^{r,cam.}|\Theta_k^{r,o}\right) = \prod_{l_j \in \Theta_k^{r,o}} p\left(z_{k,j}^{r,cam.}|x_k^r, l_j^r\right), \tag{3.15}$$

where $\Theta_k^{r,o} \subseteq \Theta_k^r$ are the overall involved states, $z_{k,j}^r$ is an (image) observation of landmark l_j, and the product accounts for all landmark observations captured at time t_k.

The observation model for each such observation is given by

$$z_{i,j}^r = h\left(x_i^r, l_j\right) + v_i^r \ , \ v_i^r \sim N\left(0, \Sigma_v^r\right), \tag{3.16}$$

where v_i^r is an additive Gaussian noise with noise covariance matrix Σ_v^r. Note that the landmark l_j can be either a random variable (SLAM framework) or a priori known (but uncertain) landmark at planning time.

Chapter 4

Approach

Our work can be divided into two separated approaches: single-robot and multi-robot. The main principle for both of the approaches is to improve the navigation accuracy of vision-aided inertial navigation systems using designated methods, such as active online sensors calibration or special cooperation between several robots.

4.1 Single-robot Approach

In the single-robot approach we incorporate into the belief space planning equations sensors calibration aspects such that navigation accuracy is significantly improved while using vision-aided inertial navigation systems. Our approach extends the state of the art by incorporating into the belief (3.3) sensor calibration model and incorporating into the objective function, $J\left(U_{k:k+L-1}\right)$ a cost function that quantifies calibration quality, possibly in combination with pose uncertainty reduction. We note that in lack of sources of absolute information (such as reliable GPS or known 3D points), accurate sensor calibration is of great importance for improving navigation accuracy. This is particularly true in GPS-deprived scenarios that involve segments of inertial navigation without vision-aiding. Moreover, we specifically consider high-rate IMU sensors and use the principle of the pre-integrated IMU factor also within the planning problem to avoid performing MAP inference (which required high computational power) at IMU rate instead of the (much lower) camera rate.

Specifically, we consider the following objective function over a planning horizon of L steps

$$J_k\left(b\left(\Theta_{k+L}\right), U_{k:k+L-1}\right) \doteq \sum_{l=1}^{L} \mathbb{E}\left(cf_l\left(b\left(\Theta_{k+l}\right), u_{k+l-1}\right)\right), \tag{4.1}$$

where cf_l represents an immediate cost function and the expectation accounts for all possible realizations of the (unknown) future sensor observations. The optimal control is defined as

$$U^{\star}_{k:k+L-1} = \underset{U_{k:k+L-1}}{argmin}\, J_k\left(b\left(\Theta_{k+L}\right), U_{k:k+L-1}\right). \tag{4.2}$$

One can identify best robot actions (or motion plans), among those generated by existing motion

planning approaches (e.g. sampling based approaches), or resort to direct optimization techniques to obtain locally optimal solutions in a timely manner [17].

4.1.1 Inference and Recursive Formulation of the Belief

The belief at the l^{th} look ahead time is defined, similarly to Eq. (3.3), as

$$b\left(\Theta_{k+l}\right) \doteq p\left(\Theta_{k+l}|Z_{0:k+l}, U_{0:k+l-1}\right), \tag{4.3}$$

where $Z_{0:k+l}$ and $U_{0:k+l-1}$ represent, respectively, all the measurements and controls up to (future) time t_{k+l}, while present time is t_k.

Determining optimal actions according to Eq. (4.2) involves evaluating the objective function J_k for different candidate actions $U_{k:k+L-1}$,given immediate costs cf_l, that we discuss in Section . This requires performing maximum a posteriori inference over the belief (4.3) such that

$$b\left(\Theta_{k+l}\right) = \mathcal{N}\left(\Theta^{\star}_{k+l}, \Lambda^{-1}_{k+l}\right), \tag{4.4}$$

where $\Theta^{\star}_{k+l}$ and Λ_{k+l} are the mean vector and information matrix, respectively.

Remark: We note that Λ_{k+l} can be calculated even though the actual values of future observations $Z_{k+1:k+l}$ are unknown, while $\Theta^{\star}_{k+l}$ is determined solely by the motion model by taking the common measurement likelihood assumption. Further details are outside the scope of this paper and can be found in [17].

The belief (4.3) can be recursively written as (see e.g. [13])

$$\begin{aligned} b\left(\Theta_{k+l}\right) &= \eta b\left(\Theta_{k+l-1}\right) p\left(x_{k+l}|x_{k+l-1}, u_{k+l-1}, c_{k+l-1}\right) \\ &\quad p\left(c_{k+l}|c_{k+l-1}\right) p\left(z_{k+l}|\Theta^{o}_{k+l}\right), \end{aligned} \tag{4.5}$$

where η is a normalization constant and the other products are the belief at the previous step, the motion model, the calibration model and the observation model. Also, $\Theta^{o}_{k+l} \subseteq \Theta_{k+l}$ are the involved random variables in the measurement likelihood term $p\left(z_{k+l}|\Theta^{o}_{k+l}\right)$, which can be further expanded in terms of individual measurements $z_{k+l,j} \in z_{k+l}$ representing observations of 3D points l_j.

Similarly to Eq. (3.15), the measurement likelihood term can be written as,

$$p\left(z_{k+l}|\Theta^{o}_{k+l}\right) = \prod_{l_j \in \Theta^{o}_{k+l}} p\left(z_{k+l,j}|x_{k+l}, l_j\right), \tag{4.6}$$

where the product considers all the landmarks $l_j \in L_k$ expected to be observed from a future viewpoint x_{k+l}. In general, these could be landmarks mapped by planning time t_k or a priori known landmarks that correspond to known areas in the environment (if any).

4.1.2 Incorporating (Pre-Integrated) IMU Measurements

Further, since we are focusing on active visual-inertial SLAM, the motion model in Eq. (4.5) is actually represented by the nonlinear inertial equations and IMU measurements, see Section 3.1. Thus, the motion model term $p\left(x_{k+l}|x_{k+l-1}, u_{k+l-1}, c_{k+l-1}\right)$ should be replaced by the corresponding product of IMU models $\prod_i p\left(x_i|x_{i-1}, c_{i-1}, z_{i-1}^{IMU}\right)$, representing the expected IMU measurements to be obtained between the time instances t_{k+l-1} and t_{k+l} given action u_{k+l-1}, i.e. $t_i \in [t_{k+l-1}, t_{k+l}]$.

As in passive visual-inertial SLAM, this would involve adding numerous factors and variables into the factor graph and the underlying MAP inference (4.4), thereby negatively impacting computational performance. Instead, we propose to use the concept of IMU pre-integration (see Section 2.4 for more details) and add a single probabilistic term that represents the motion model between camera times t_{k+l-1} and t_{k+l}. In other words, the motion model $p\left(x_{k+l}|x_{k+l-1}, u_{k+l-1}, c_{k+l-1}\right)$ in Eq. (4.5) is replaced by $p\left(x_{k+l}|x_{k+l-1}, \Delta x_{k+l-1\to k+l}, c_{k+l-1}\right)$, where $\Delta x_{k+l-1\to k+l}$ is the relative motion computed via pre-integrating (future) IMU measurements in the time frame $[t_{k+l-1}, t_{k+l}]$.

Note these measurements are a function of the considered actions via the nonlinear inertial navigation equations (3.1). Although the actual values of these (future) IMU observations are unknown, it is not required for calculating the posterior information matrix.

4.1.3 Choice of Cost Functions

Thus far, we showed how to incorporate within a future belief $b\left(\Theta_{k+l}\right)$ IMU measurements and sensor calibration models (e.g. IMU bias) while considering general immediate cost functions. In this section we focus on a specific family of cost functions, cf, that utilize this new information within planning, considering different types of planning objectives. We define a general cost function [17]:

$$cf\left(b\left(\Theta_{k+l}\right), u_{k+l}\right) = \left\|E_{k+l}^{G}\Theta_{k+l}^{\star} - \Theta^{G}\right\|_{M_\Theta} + \left\|\zeta\left(u_{k+l}\right)\right\|_{M_u} + \underbrace{tr\left(M_\Sigma \Lambda_{k+l}^{-1} M_\Sigma^T\right)}_{\doteq cf^{\Sigma}\left(M_\Sigma, \Lambda_{k+l}^{-1}\right)}, \quad (4.7)$$

where M_Σ, M_u and M_Θ are given weight matrices that can be adjusted online [16, 17], and $\zeta(u)$ is some known function that, depending on the application, quantifies the usage of control u. Θ^G is a predefined goal and E_{k+l}^G is a selection matrix, such that the matrix $E_{k+l}^G\Theta_{k+l}^\star$ contains a subset of states for which we want to impose a goal. Similarly, the matrix M_Σ in term cf^Σ can be used to choose the covariance of chosen states (position, pose, calibration etc.) from the joint covariance Λ_{k+l}^{-1}, i.e. consider only uncertainty of these states. We can consider different variations for the uncertainty cost term cf^Σ, such as accurate navigation or sensors self-calibration by using M_Σ that choose only the pose or calibration states, respectively. In this work we consider the uncertainty cost term $cf^{\Sigma^{TO}}$ that presents a 'trade-off' between two planning objectives (concurrent, or following each other): reaching a target with minimum navigation errors (i.e. accurate navigation) and calibrated sensors (i.e. self-calibration). The corresponding

cost function is defined as

$$cf^{\Sigma^{TO}} \doteq tr\left(M_{\Sigma_c}\Lambda_{k+l}^{-1}M_{\Sigma_c}^T\right) + tr\left(M_{\Sigma_x}\Lambda_{k+l}^{-1}M_{\Sigma_x}^T\right), \tag{4.8}$$

where the terms penalize, respectively, pose uncertainty and quantify calibration quality. However, given the fact that there is a strong coupling between inertial sensors calibration to the pose uncertainty and that the belief includes both the pose and calibration states, the sensor calibration is still accounted for even if the uncertainty cost is only over position.

4.2 Multi-Robot Approach

Collaboration between robots can significantly improve performance of both inference and planning phases. In particular, by sharing relevant information between robots, navigation accuracy can substantially improves, while by appropriately coordinating actions the robots can often finish a task in a shorter time. Previous works present active collaboration multi-robot approaches where the robots cooperate only while being in the same region (observing same scene or each other) and planning that aims to reduce the estimation uncertainty of each robot. Our research further advances the state of the art by defining two collaborative robot subgroups, the "strategic group" and the "expendable group".

The key idea is to have robots from the "expendable robots" improve performance, in terms of estimation accuracy, of robots from the "strategic group" while the latter execute time-critical missions. Robots from the former group are envisioned to be cheap and therefore expandable, with their sole purpose to boost the performance of the "strategic group". Along this concept, which we refer to as "Expendable Cooperation", we also introduce the notion of indirect updates between different robots. The basic idea is to exploit prior correlation between states of different robots such that observation made by one robot (from the "expendable group") impacts another robot (from "strategic group"). Indirect updates thus allow multi-robot cooperative estimation even when the robots operate in separate regions of the environment, allowing robots from the "strategic group" to carry on their time-critical missions, while the "expendable group" robots are tasked with reducing the estimation uncertainty of the former group. We define and elaborate on these innovative concepts in the following sections.

4.2.1 Multi-Robot Notations

In this section we briefly recall the relevant notations and definitions from Section 3. We define the joint state

$$\Theta_k^r \doteq \{X_k^r, C_k^r, L_k^r\}, \tag{4.9}$$

where $X_k^r \doteq \{x_0^r, \ldots, x_k^r\}$ and $C_k^r \doteq \{c_0^r, \ldots, c_k^r\}$ represent, respectively, navigation and calibration states of robot r up to time t_k.

Similarly, we define $Z_{0:k}^r \doteq \{z_0^r, \ldots, z_k^r\}$ and $U_{0:k}^r \doteq \{u_0^r, \ldots, u_k^r\}$, where u_i^r and z_i^r represent control and obtained measurements of robot r at time t_i. In the following, we shall use

the notation $z_i^{r,IMU} \in Z_i^r$ to denote IMU measurements and $z_i^{r,cam.} \in Z_i^r$ to denote camera observations.

We consider now a group of R collaborating robots, and denote by Θ_k the corresponding joint state

$$\Theta_k \doteq X_k \cup C_k \cup L_k \tag{4.10}$$

where $X_k \doteq \{X_k^r\}_{r=1}^R$, $C_k \doteq \{C_k^r\}_{r=1}^R$ and L_k represents the perceived environment by the entire group.

Given these definitions, the multi-robot's belief at time t_k, is given by [13, 21]

$$b(\Theta_k) \doteq p(\Theta_k | Z_{0:k}, U_{0:k-1}) \tag{4.11}$$

where $U_{0:k-1} \doteq \left\{U_{0:k-1}^r\right\}_{r=1}^R$ and $Z_{0:k} \doteq \{Z_{0:k}^r\}_{r=1}^R$.

The explicit definition of the belief and the extension to the belief at the l^{th} look ahead time are presented in detail in Section 4.2.3.

4.2.2 Expendable Cooperation

Multi-robot belief space planning can be framed as finding the optimal control $U_{k:k+L-1}^\star = \underset{U}{argmin}\, J(U_{k:k+L-1})$ given a general objective function

$$J_k(b(\Theta_{k+L}), U_{k:k+L-1}) \doteq \sum_{l=1}^{L} \mathbb{E}\left(cf_l\left(b(\Theta_{k+l}), u_{k+l-1}\right)\right), \tag{4.12}$$

where the costs can in general involve the belief over the entire joint state of all robots, i.e. the belief for the lth look ahead step, $b(\Theta_{k+l})$.

A common instantiation of this general formulation is where the cost of each robot involves only its own belief from appropriate future time (see e.g. [13, 21, 37]),

$$J_k(b(\Theta_{k+L}), U_{k:k+L-1}) = \mathbb{E}\left[\sum_{l=1}^{L}\sum_{r=1}^{R} cf_l^r\left(b\left(\Theta_{k+l}^r\right), u_{k+l-1}^r\right)\right] \tag{4.13}$$

where $b\left(\Theta_{k+l}^r\right) = \int\limits_{\neg\Theta_{k+l}^r} b(\Theta_{k+l})$.

Such a form, supports collaborative active state estimation, where each robot aims to improve its estimation accuracy while considering additional terms in cf_l, if exist. This formulation is beneficial in order to improve the estimation accuracy in a collaborative multi-robot setting where reducing the uncertainty of all robots is of interest (see e.g. [13, 21, 37]).

In contrast, we consider an alternative formulation addressing a multi-objective setting where a pre-defined subset of robots (denoted as "strategic group") is to reach a goal in minimum time and uncertainty, while the remaining robots (denoted as "expendable group") are solely entrusted in improving the performance of the former subgroup of robots. Intuitively, we envision robots from the "extendable group" to be cheap and affordable robots that one could spare, and we ad-

vocate to appropriately use these robots to boost the estimation accuracy of the "strategic group" robots, which are envisioned to be expensive. In particular, such a concept enables the strategic robots to focus mainly on their time-critical tasks (e.g. reaching a goal in minimal time), while relying on the expendable robots to maximally reduce estimation uncertainty of the former.

Specifically, instead of (4.13), we consider the following definition of the cost function cf_l from Eq. (4.12):

$$cf_l \doteq \sum_{r^A \in R^A} cf_l^{r^A}\left(b\left(\Theta_{k+l}^{r^A}\right), u_{k+l-1}^{r^A}\right) + \sum_{r^B \in R^B} cf_l^{r^B}\left(b\left(\Theta_{k+l}^{R^A}\right), u_{k+l-1}^{R^A}\right) \tag{4.14}$$

where R^A and R^B denote the "strategic group" and the "expendable group", respectively. Similarly, $cf_l^{r^A}$ and $cf_l^{r^B}$ are the cost functions for the robots in the "strategic group" and the "expendable group", respectively. Also, $\Theta_{k+l}^{R^A}$ and $u_{k+l-1}^{R^A}$ are, respectively, the joint state and the control at time t_{k+l-1} for all the robots in the "strategic group", R^A. Compared to the corresponding cost formulation in (4.13), where the cost of each robot involves a belief over its own state, our formulation devotes all the resources of the expendable group to aid the strategic group.

For example, for two robots ($R^A = \{r^A\}$, $R^B = \{r^B\}$), the above cost function reduces to,

$$cf_l = cf_l^{r^A}\left(b\left(\Theta_{k+l}^{r^A}\right), u_{k+l-1}^{r^A}\right) + cf_l^{r^B}\left(b\left(\Theta_{k+l}^{r^A}\right), u_{k+l-1}^{r^A}\right) \tag{4.15}$$

where r^A is the "strategic robot" and r^B is the "expendable robot".

Note that both cost functions, $cf_l^{r^A}$ and $cf_l^{r^B}$, include a belief over the state of the strategic group, which is r^A in this case. For instance, r^B can aim to minimize uncertainty of r^A and not its own uncertainty. Typically, our approach improves the performance of the strategic robots, however there are few specific scenarios that we can get identical results by using the previous approach. A detailed analysis for a family of cost functions and the connection between the objective functions of the two methods, i.e. Eq. (4.13) and (4.14), is discussed in Section 5.2.2.

To summarize this section, the proposed concept supports collaborative active state estimation with expendable robots that aim to improve estimation accuracy of strategic robots. This concept enables the strategic robots to focus on their main task, e.g. reaching some goal, while their navigation accuracy is taking care by other robots. In addition, we can use this concept also to calibrate the strategic robots via expendable robots.

In the next section we discuss a key aspect in this concept - how the expendable robots can indeed be used to reduce the estimation uncertainty of the strategic robots.

4.2.3 Multi-Robot Belief Propagation and Indirect Cooperation

Collaboration multi-robot can be achieved by using several cooperation methods. Until our research, these methods [27, 13, 21] can be divided into two categories: cooperation via observing

other robots or cooperation via observing mutual landmark (see graphical illustration on Figure 4.1 (a) and (b)). In other words, as shown in the figures, these cooperation methods require direct connection between the robot. Specifically, method (b) is more flexible than method (a) because it enables observing mutual landmarks at different times. In our research we want to increase the contribution of the "expendable group" concept, and relax the cooperation requirement for a direct connection between the robots (even for method (b)), by enabling indirect observation updates given the states of different robots are correlated (see graphical illustration on Figure 4.1 (c)). More specifically, Figure 4.1 (c) shows that when robot r_1 observes a previously mapped landmark l_1, the estimation update for robot r_1 impacts the state estimation of robot r_2 due to the initial correlation between them. This indirect concept means that given prior correlation, informative observation made by one robot, impacts also the states of other robots.

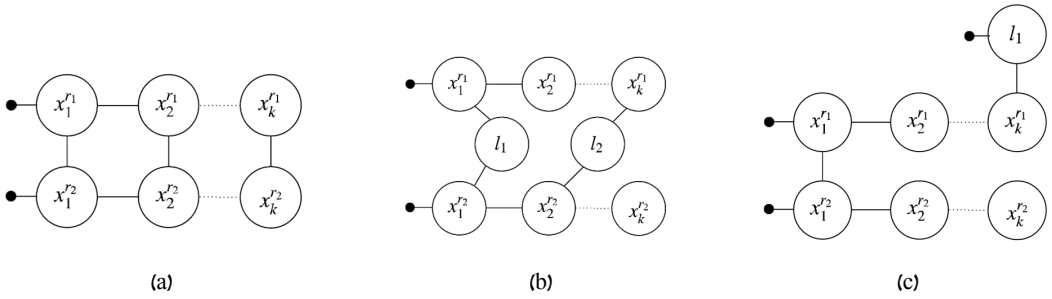


Figure 4.1: Factor graphs illustrations for multi-robot cooperation concepts. (a) Cooperation via observing other robots; (b) Cooperation via observing same landmark; (c) Indirect cooperation given past correlation . While (a) and (b) are direct cooperation concepts requiring direct connection between robots, (c) is the indirect cooperation concept given past correlation

In the following subsections, we present a theoretical analysis that demonstrates the indirect cooperation concept, and then elaborate on a multi-robot belief that includes all the described ingredients.

Theoretical Analysis

We present a simple theoretical analysis to demonstrate the indirect update concept. We assume a simple case of two robots, r_1 and r_2, starting with some initial correlation (e.g. mutual observation or prior knowledge). We will assume that robot r_1 observes an a priori known region and the other robot does not make any observation. Our goal is to show that, in presence of prior correlation, the covariance of r_2 will be updated as a result of an informative observation made by r_1, despite the fact that r_2 did not make any observations.

Let Σ and I represent the joint prior covariance and information matrix, respectively:

$$I \doteq A^T A = R^T R \ , \ \Sigma = I^{-1} \doteq \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}, \tag{4.16}$$

where A is the Jacobian matrix and R is the square root information matrix. The matrix R can

be calculated from the information matrix using Cholesky factorization.

For the simple case of our analysis the form of matrix R is defined as shown in figure 7.1, where the columns and rows represent the robots and r_{12} represent the correlation between them.

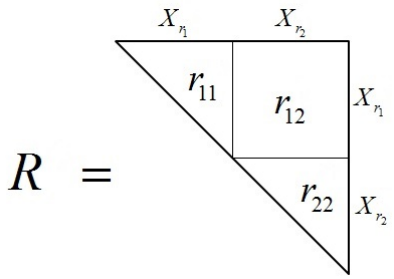


Figure 4.2: The square root information matrix R for the simple case of two robots

The Jacobian matrix A, is the Jacobian over the observation model $z = h(x) + v$ and it is calculated by,

$$A \doteq \nabla_X (h(X)) = \begin{bmatrix} a_1 & \dots & a_n \end{bmatrix} \tag{4.17}$$

where n is the number of robots.

For our case, only robot r_1 observe a landmark, so at the observation time the Jacobian matrix is,

$$A = \begin{bmatrix} a_1 & 0 \end{bmatrix} \tag{4.18}$$

To calculate R^k we need to update the matrix R^{k-1} with the Jacobian A^k, as shown in Figure 7.2.

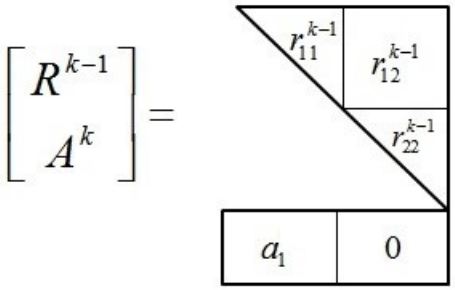


Figure 4.3: The matrix at time t_k after adding the Jacobian A^k. This matrix is updated using Given Rotation in order to calculate the a posteriori information matrix R^k

This update can be efficiently performed using Givens Rotations (See the Appendix for more details) that are used to nullify all entries of A^k and appropriately updating the entries of R^{k-1} (see Figure 7.2), yielding the a posteriori information matrix R^k,

$$R^k = \begin{bmatrix} r_{11}^k & r_{12}^k \\ 0 & r_{22}^k \end{bmatrix}$$

In our case a_1 is a very accurate measurement due to the a priori known region observation. We want to show that this measurement impact also robot r_2 estimation and covariance, i.e. Σ_{22}^k will be updated although robot r_2 did not measured anything.

In order to calculate the matrices R^k and Σ^k from $\begin{bmatrix} R^{k-1} & A^k \end{bmatrix}^T$ we used relations described in the literature [10, 22, 15] get the following expression for Σ_{22}^k (See the Appendix (Section 7) for the full proof),

$$\Sigma_{22}^k = \frac{\left(r_{11}^k\right)^2}{\left(r_{22}^{k-1}\right)^2 \left(r_{11}^k\right)^2 + \left(-a_1^{(0)} r_{12}^{k-1}\right)^2} \tag{4.19}$$

It is shown in Eq. (7.9) that in the case without correlation between the robots, i.e. $r_{12}^{k-1} = 0$, the covariance matrix is not updated

$$\Sigma_{22}^k = \left(r_{22}^{k-1}\right)^{-2}$$

On the other hand, with a correlation between the robots and observation made by robot r_1, the covariance of robot r_2 is updated - the stronger the initial correlation, the stronger the impact.

To conclude this section, we present a novel approach for indirect collaborative state estimation with expendable robots that can improves significantly the performance of the strategic robots in terms of navigation accuracy, calibration quality and overall path length/time.

Belief Space Planning Formulation

In this section we will present how all the multi-robot concepts we presented in the previous sections are integrated into the multi-robot belief space planning formulation.

The belief presented in Eq. (4.11) can be explicitly written in terms of individual motion, calibration and observation models as,

$$b\left(\Theta_k\right) = \prod_{r=1}^{R} \left\{ priors^r \prod_{i=1}^{k} p\left(x_k^r | x_{k-1}^r, c_{k-1}^r, u_{k-1}^r\right) p\left(c_k^r | c_{k-1}^r\right) p\left(z_k^{r,cam.} | \Theta_k^{r,o}\right) \right\} \tag{4.20}$$

where $p\left(x_k^r | x_{k-1}^r, c_{k-1}^r, u_{k-1}^r\right)$ is the motion model, $p\left(c_k^r | c_{k-1}^r\right)$ is the calibration model and $p\left(z_k^{r,cam.} | \Theta_k^{r,o}\right)$is the observation model. Also, $\Theta_k^{r,o} \subseteq \Theta_k^r$ are the involved random variables in the measurement likelihood term $p\left(z_k^{r,cam.} | \Theta_k^{r,o}\right)$, which can be further expanded in terms of individual measurements $z_{k,j}^{r,cam.} \in z_k^{r,cam.}$ representing observations of 3D points l_j.

Similarly to the single-robot definitions, the belief at the l^{th} look ahead step can be written recursively as follow (see e.g. [13, 21]),

$$b\left(\Theta_{k+l}\right) = \eta b\left(\Theta_{k+l-1}\right) \prod_{r=1}^{R} p\left(x_{k+l}^r | x_{k+l-1}^r, c_{k+l-1}^r, u_{k+l-1}^r\right) p\left(c_{k+l}^r | c_{k+l-1}^r\right) p\left(z_{k+l}^{r,cam} | \Theta_{k+l}^{r,o}\right) \tag{4.21}$$

where η is a normalization constant.

We now focus on the measurement likelihood term $p\left(z_{k+l}^{r,cam}|\Theta_{k+l}^{r,o}\right)$ that represents sensor observations of the environment (represented e.g. by 3D points). However, now, these are future observations of the environment to be made according to robot r's planned motion. As shown in [13, 21] this term can be explicitly written as,

$$\begin{aligned} p\left(z_{k+l}^{r,cam}|\Theta_{k+l}^{r,o}\right) &= \prod_{l_j \in \Theta_{k+l}^{r,o}} p\left(z_{k+l,j}^{r,cam}|x_{k+l}^{r}, l_j\right) \\ &\cdot \prod_{l_j^{Reg} \in \Theta_{k+l}^{r,o}} p\left(l_j^{Reg}\right) \cdot p\left(z_{k+l,j}^{r,cam}|x_{k+l}^{r}, l_j^{Reg}\right) \\ &\cdot \prod_{j} p\left(\xi_{k+l,k+j}^{r,r'}|x_{k+l}^{r}, x_{k+j}^{r'}\right) \end{aligned} \tag{4.22}$$

where l_j^{Reg} represents all the landmarks in the a priori known regions with uncertainty level represented by the prior $p\left(l_j^{Reg}\right)$ and $\xi_{k+l,k+j}^{r,r'}$ refers to a relative pose constraint calculated from raw measurements z_{k+l}^{r} and $z_{k+j}^{r'}$.

The first product represents observations of previously mapped 3D points $l_j \in L_k$ with $\Theta_{k+l}^{r,o}$ including those 3D points that are actually visible from x_{k+l}^{r}. The second product represents observations of landmarks within the a priori known regions that are actually visible from x_{k+l}^{r}. The last product represents future multi-robot constraints that correspond to different robots, r and r', mutually observing unknown scenes from different viewpoints.

Note, that all the three products do not always exist. The first two products exist when $l_j \in \Theta_{k+l}^{r,o}$ and $l_j^{Reg} \in \Theta_{k+l}^{r,o}$ exist. The last product exists only in the case where the robots are close enough to each other (defined by specific distance threshold).

This definition of the belief at the l^{th} look ahead step, includes the three cooperation method described before, the direct and the indirect (see Figure 4.1). The direct methods (a) and (b) are described by the last product in Eq. (4.22), while for method (a), robots see each other, the future times are equal, i.e. $l = j$. The indirect method (c) is described by the first and the second products in Eq. (4.22) in addition to the correlation between the robots encoded in the term $b\left(\Theta_{k+l-1}\right)$ in Eq. (4.21).

Chapter 5

Results

We study our proposed approaches in simulation considering autonomous navigation in partially unknown, uncertain, GPS-deprived environments. In the considered scenarios, an aerial robot (e.g. quadrotor) has to autonomously navigate to a pre-defined goals with minimum localization uncertainty while utilizing limited prior information regarding the environment and using its onboard sensors. The latter include a downward-facing monocular camera and an IMU.

Given a robot's trajectory, our simulation generates synthetic IMU measurements and camera observations, which are used to formulate appropriate pre-integrated IMU and projection factors, as described in Section 3. Image observations are corrupted by zero-mean Gaussian noise with standard deviation of 0.5 pixels. A simple IMU calibration model is considered, comprising accelerometers bias and gyroscopes drift of $10\,mg$ bias and $10\,^{\circ}/hr$ drift, respectively. Our Matlab implementation uses GTSAM library [6] for computationally efficient inference.

5.1 Single-Robot Results

We demonstrate our single-robot approach in comparison to other methods presented in previous works. We consider a scenario with environment comprises randomly scattered landmarks and an area without landmarks at all, where the goal is located (see Figure 5.6). Such a scenario was designed to highlight the importance of incorporating online sensor calibration aspects within planning, as in the area surrounding the goal no landmark observations can be obtained and inertial navigation is required; for example, this could model a dark or texture-less area. Navigation estimation error at the goal thus depends on IMU sensor calibration, in addition to the solution accuracy provided by visual SLAM while landmark observations are available. Clearly, navigation accuracy at the goal is expected to differ significantly for poorly- and well-calibrated IMU sensors.

For simplicity we assume the robot can only control its heading angle while keeping the velocity constant. The control effort, $\zeta(u)$ in Eq. (4.7), is therefore defined as the change in the heading angle. Such a setting induces single-axis maneuvers which are insufficient to yield a fully observable system with the considered visual-inertial SLAM framework [1, 32]; in particular, biases of accelerometer sensors cannot be calibrated in all axes.

For this reason, we consider some regions in the environment (in terms of landmarks within those regions) are a priori known to the robot with different levels of uncertainty (ranging between 10^{-5} and 10 meters) and can be used for updating the robot's belief. As will be seen in Section 5.1.1, regions that are known with high-accuracy can be used for accelerometer sensor calibration and for position updates, while those with lower level of accuracy (10 meters) can only be used for position updates. No other information is initially available, and thus the robot has to map the environment while navigating through it, i.e. perform SLAM, and consider appropriate aspects within the planning phase.

This section in organized as follows, first we demonstrate (using passive simulation) the influence of known regions over the robot's performance, then we present results from a simple case of active online calibration only and in the last two sections we demonstrate our approach contribution comparing to previous works.

5.1.1 Known Regions Influence - Study Case

We examine the influence of different uncertainty levels of priori known regions on the navigation and calibration performance. We simulate the same predefined trajectory considering identical environment in terms of landmarks, while changing only the uncertainty levels of landmarks within the a priori known region (green square). The results are presented in Figures 5.1-5.3.

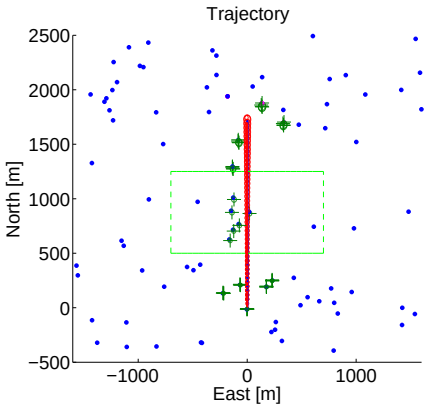


Figure 5.1: Straight forward trajectory with known region (green square). The unknown landmarks in the environment are the blue dots and the observed landmarks are shown with + on them.

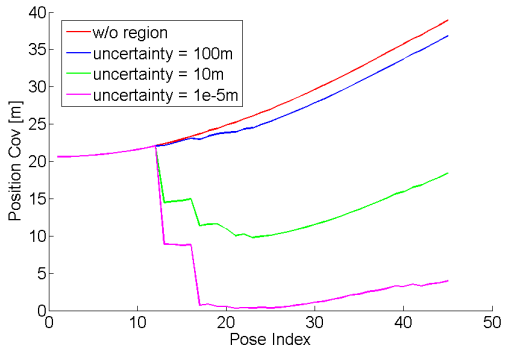


Figure 5.2: Position covariance results using different uncertainty levels of priori known regions

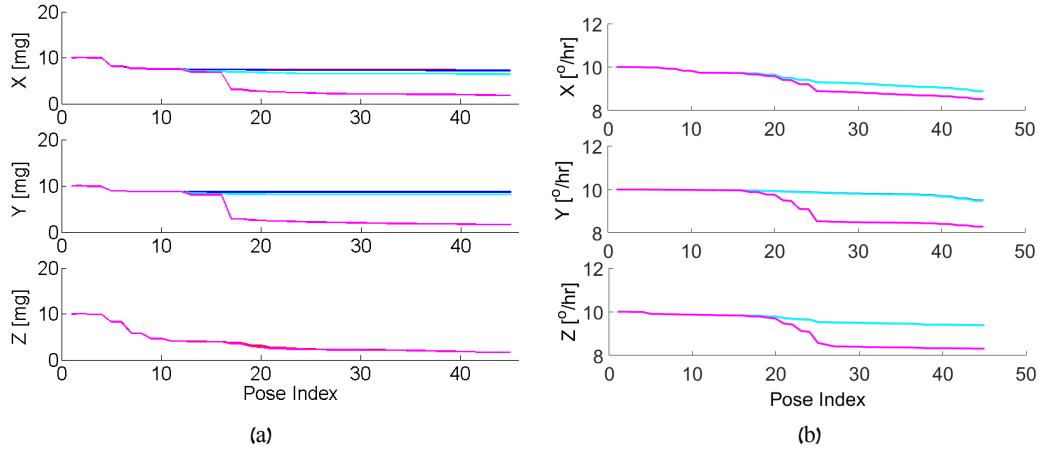


Figure 5.3: Calibration performance using different uncertainty levels of priori known regions, (a) Accelerometers calibration covariance results and (b) Gyroscopes calibration covariance results

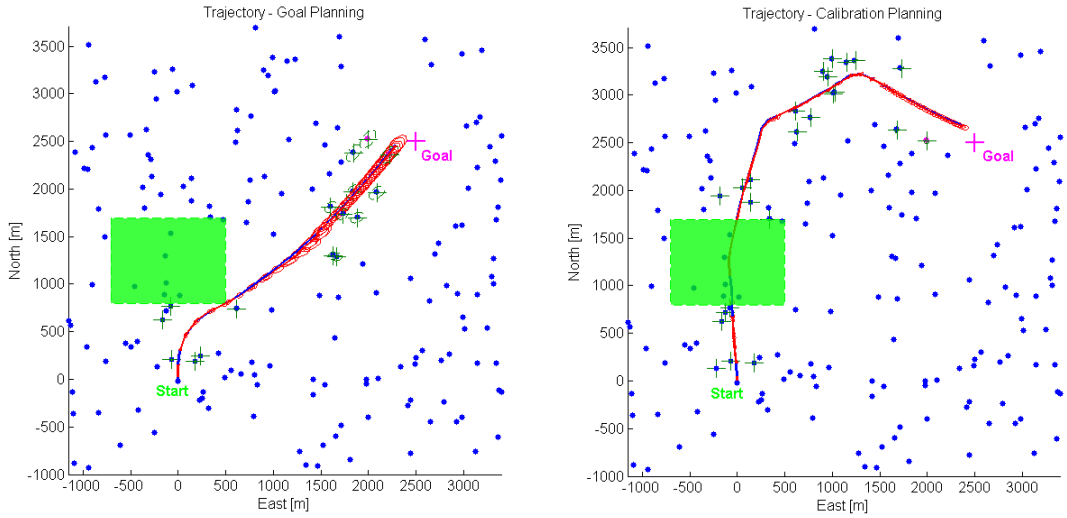
It is shown that it is possible to calibrate only the IMU's accelerometers using known region with very small uncertainty.

5.1.2 Active Online Sensors Calibration Results

In this section we present the influence on sensors calibration and navigation uncertainty by using 'reaching-goal' scenario comparing to 'online-calibration' scenario. These two scenarios are identical in terms of sensors parameters and environment (landmarks, known region and goal) and the only difference between them are the weight matrices, M_Θ and M_Σ, in Eq. (4.7). For 'reaching-goal' case, the weight matrix M_Σ in Eq. (4.7) is set to zero, meaning that the robot should plan for the shortest path to goal. For 'online-calibration' case, the weight matrix M_{Σ_x} is set to zero and the weight matrices M_Θ, M_{Σ_c} are adjustable in time according to the calibration level of the robot.

We use a gradient descent method for optimizing the outer layer and Gauss-Newton for calculating inference of the belief. And we define the initial trajectory as five steps ($\sim 200m$) of straight forward that don't pass through the known region.

The comparison between the two scenarios is presented in the following figures. The robot's trajectories are shown in Figure 5.4, where the green square is a known region with uncertainty of $10^{-5}m$. The unknown landmarks in the environment are the blue dots and the observed landmarks along the path are shown with '+' on them. Performance comparison between the two methods in term of estimation uncertainty (position and accelerometers calibration) is presented in Figure 5.5. The presented results are the SLAM results given the chosen path in each method.



(a) (b)

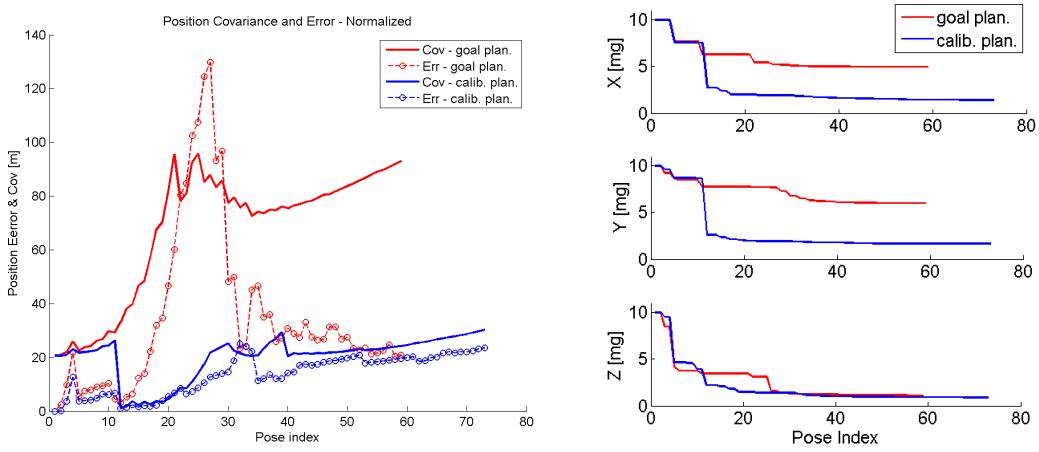
Figure 5.4: Simulation trajectories with known region (green square) using (a) 'reaching-goal' scenario or (b) 'online-calibration' scenario. The unknown landmarks in the environment are the blue dots and the observed landmarks are shown with + on them. These two scenarios are identical in terms of sensors parameters and environment (landmarks, known region and goal).

It is shown that the planning in our approach takes into account the known regions in the environment in order to keep the accelerometer's calibration minimum as possible. In addition, the known region drastically affects the navigation accuracy as shown in Figure 5.5.

5.1.3 Compared Approaches

We compare our approach, denoted as 'BSP-Calib', with two other methods: 'BSP' (previous approaches, [17, 16]) and 'Shortest-Path'. The objective function in all the three methods can be described via Eq. (4.7) with weights appropriately adapted on the fly, as discussed below.

The BSP method does not incorporate sensor calibration states within the belief, which therefore describes at time k the joint pdf over past and current robot poses X_k and mapped landmark L_k. Belief evolution along a given candidate trajectory thus considers perfectly calibrated sensors, both within inference and planning. In this method, similarly to [17], we balance between position uncertainty reduction versus goal achievement by defining a soft bound $tr\left(M_{\Sigma_x}\Lambda_{k+l}^{-1}M_{\Sigma_x}^T\right) \leq \beta^x$, and set the weights $M_\Theta = \alpha_k^x$ and $M_{\Sigma_x} \doteq \sqrt{1-\alpha_k^x}\bar{M}_{\Sigma_x}$, where $\alpha_k^x \doteq 1 - \frac{tr\left(M_{\Sigma_x}\Lambda_{k+l}^{-1}M_{\Sigma_x}^T\right)}{\beta^x} \in [0,1]$ and $\bar{M}_{\Sigma_x}$ is a selection matrix that extracts the appropriate



(a) (b)

Figure 5.5: Performance comparison in terms of estimation uncertainty: (a) position covariance and error and (b) accelerometers calibration covariance.

position covariance from Λ_{k+l}^{-1}. See [17] for further details.

In our approach, the BSP-Calib method, we consider a modification of the above to account also for the sensor calibration uncertainty while setting the weights. Specifically, we consider a soft bound β^c on sensor calibration uncertainty (accelerometer biases in our case), and let $\alpha_k^c \doteq 1 - \frac{tr\left(M_{\Sigma_c}\Lambda_{k+l}^{-1}M_{\Sigma_c}^T\right)}{\beta^c} \in [0,1]$ and $M_{\Sigma_c} \doteq \sqrt{1-\alpha_k^c}\bar{M}_{\Sigma_c}$, where $\bar{M}_{\Sigma_c}$ is defined similarly to $\bar{M}_{\Sigma_x}$. Since we now have two online-calculated parameters α_k^x and α_k^c, we set $M_\Theta = \min\left(\alpha_k^x, \alpha_k^c\right)$, and $M_{\Sigma_x} \doteq \sqrt{\alpha_k^c\left(1-\alpha_k^x\right)}\bar{M}_{\Sigma_x}$. The intuition here is as follows: as we approach any of the soft bounds β^x or β^c, i.e. $\alpha^x \to 0$ or $\alpha^c \to 0$, the goal attainment term is disabled to allow for active sensor calibration or position uncertainty reduction. However, since the environment is unknown and may include areas without landmarks (texture-less or dark areas) that require inertial navigation, we prioritize sensor calibration over reduction in position uncertainty, since this would yield slower uncertainty evolution. One can observe that when the sensor is sufficiently calibrated with respect to β^c, the method will aim to actively reduce position uncertainty in case $\alpha_k^x \to 0$, or invest efforts in goal attainment. In our implementation we use $\beta^x = 150^2\, m^2$ and $\beta^c = 2$ mg. Clearly, properly balancing the weights in a multi-objective function can be delicate and better approaches could exist. Further investigation of these aspects is left to future research.

In the Shortest-Path method we simply set all the weight matrices to zero except of M_Θ. This method will therefore choose the shortest path to the goal without considering uncertainty aspects.

5.1.4 Simulation Results

The comparison between the three methods is presented in the following figures. The robot's trajectories are shown in Figure 5.6, where the cyan square is a known region with uncertainty of $10^{-5}m$, and the green square is a known region with uncertainty of $10m$. The green square lo-

cated closer to start and goal locations. The unknown landmarks in the environment are the blue dots and the observed landmarks along the path are shown with '+' on them. Performance comparison between the three methods in terms of estimation uncertainty (position and accelerometers calibration) and paths length is presented in Figure 5.7. The presented results are the SLAM results given the chosen path in each method.

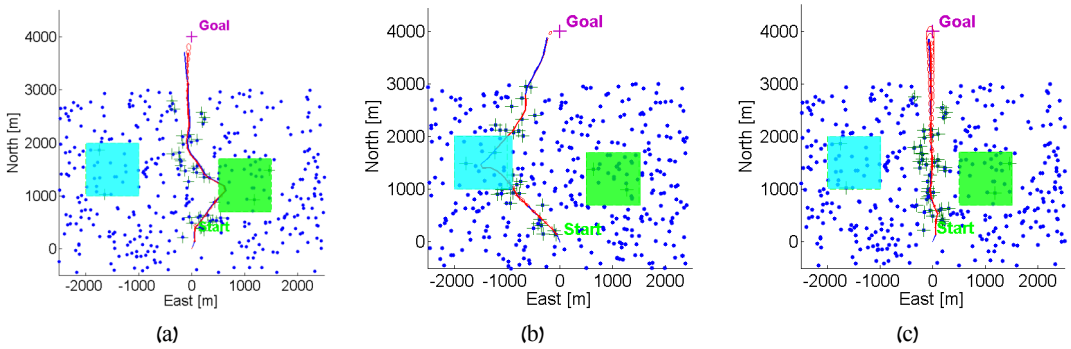


Figure 5.6: The robot's trajectories for the three methods: (a) 'BSP', (b) 'BSP-Calib' (our approach) and (c) 'Shortest-Path', where the cyan square is a known region with uncertainty of $10^{-5}m$, and the green square is a known region with uncertainty of $10m$. The unknown landmarks in the environment are the blue dots and the observed landmarks along the path are shown with + on them.

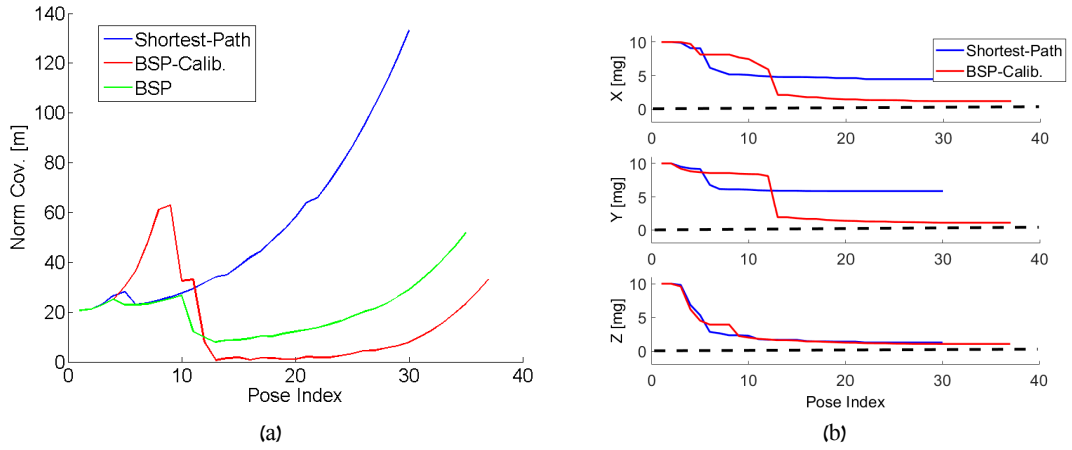


Figure 5.7: Performance comparison in terms of estimation uncertainty: (a) position covariance and (b) accelerometers calibration covariance. Calibration performance is not shown for 'BSP' as in this approach sensor calibration is not part of the belief, and is not considered in objective function. One can see that our approach, 'BSP-Calib', yields better performance despite the fact that it is the longest path to the goal.

One can see in the results that our approach, 'BSP-Calib', yields better performance, in terms of position uncertainty, despite the fact that it is the longest path to the goal. In our approach the planning chooses to go through the known region with the lower uncertainty (cyan region) in order to calibrate the robot's accelerometers and then proceed to the goal. On the other hand, the 'BSP' method does not consider the calibration state in the belief, and therefore chooses

a shorter path that reduces the pose uncertainty by reaching the closer known region but with higher uncertainty (green region).

We note that actual performance in inference, in terms of covariance uncertainty upon reaching the goal, may be actually different than the one which was calculated within planning. While the latter only considers already-mapped (or known) landmarks and dead reckoning otherwise, the inference process updates the belief based on incoming sensor observations also of new landmarks. It is thus conceptually possible that the path determined by our approach, 'BSP-Calib', will yield in inference inferior results to 'BSP' performance or 'Shortest-Path' methods. This, for example, could be the case in an environment full with (unknown) landmarks. One possible direction to approach this gap of belief evolution in planning and inference is to consider within planning some statistics regarding landmark distribution. This direction is left to future research.

5.2 Multi-Robot Results

In this study we consider a scenario with unknown environment comprising randomly scattered landmarks, an a priori known region and a goal located in a specific point (see Figure 5.11). We consider 2 aerial robots equipped with a downward-facing monocular camera and an IMU and assume the sensors for both robots are of identical quality. A probabilistic roadmap (PRM) [26] is used to discretize the (partially unknown) environment. Given a PRM, candidate paths are generated for each robot. The objective function, for each of the compared methods, is then evaluated for all combinations of candidate paths of both robots, and the optimal combination is then chosen.

This section is organized as follows. First, we provide a case study for indirect cooperation (see Section 5.2.1) in order to demonstrate the advantages of this concept. Then, in Sections 5.2.2 and 5.2.3 we demonstrate our multi-robot approach, denoted as 'Expendable-MR', which uses the cost function (4.14) and leverages indirect multi-robot updates. In these sections we also compare our approach to an alternative recent multi-robot BSP approach [13, 21]. The performance analysis in this section is in terms of position accuracy and path length.

Our multi-robot approach naturally applies also to active sensor calibration that was discussed in the single-robot case. However, while in our multi-robot simulation the belief includes sensor calibration variables, in this section we do not include in the objective function corresponding calibration costs.

5.2.1 Indirect Cooperation - Case Study

We evaluate the concept of indirect cooperation, as defined in Section 4.2.3 in a basic study, studying its feasibility in different scenarios. In all cases, we consider two robots operating in an unknown environment represented in landmarks and a priori known region. The a priori known region is used for more efficient indirectly update the strategic robot due to the significant influence on the expendable robot.

In this study we focus on the impact of indirect cooperation on the performance of both of the robots, in terms of estimation uncertainty; hence, the robot trajectories in this section are pre-defined and no planning is involved.

We consider the following three scenarios, denoted as 'Shortest-Path', 'Highest-Correlation' and 'Common-Cooperation', as shown in Figure 5.8. In that figure, the cyan square is the known region, the green dots represent the unknown landmarks and the ellipses along the robots' trajectories represent the position covariance. All the scenarios assume that the trajectory for the strategic robot, r_1, is te shortest path to the goal, while the trajectory of the expendable robot r_2 changes for each scenario, but always goes through the known region.

Specifically, in 'Shortest-Path' scenario, r_2 goes directly to the known region. In this scenario represents the shortest path for updating the strategic robot usually on account of low correlation at update time between the states of robots r_1 and r_2. The 'Highest-Correlation' scenario represents the path that gives the highest correlation at update time usually on account of update time. Finally, the 'Common-Cooperation' scenario represents the common multi-robot coop-

eration method which assumes the robots observe a mutual scene (even at different times) and does not rely on prior correlation between the robots at all. We present the robots performance in term of position accuracy (error and uncertainty).

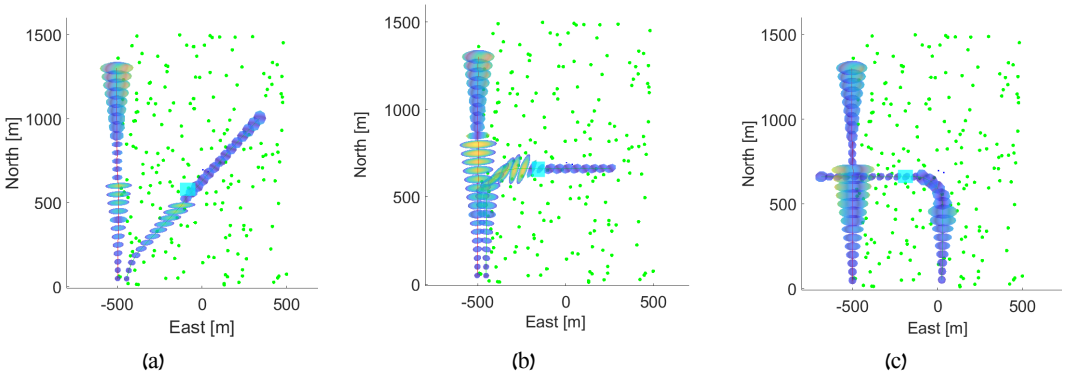


Figure 5.8: Robots' trajectories for the three scenarios: (a) 'Shortest-Path' where r_2 go directly to the known region in order to update r_1 at the shortest time, (b) 'Highest-Correlation' where r_2 go to the known region at the last time it can in order to achieve the highest correlation at update time and (c) 'Common-Cooperation' where r_2 observes a mutual scene after it passed through the known region. In all the graphs the cyan square represents the known region, the green dots represent the unknown landmarks and the ellipses along the robot's trajectories represent position covariance. In all cases, robot r_1 goes straight to goal, while robot r_2 goes through the known region.

The performance results for the three scenario are presented in Figure 5.9, where the strategic robot (r_1) is represented with red color and the expendable robot (r_2) with the green color. The shown results were generated by performing a Monte-Carlo simulation with 50 runs for each scenario. The plots show for each scenario the standard deviation (STD) over the runs results for the norm position error.

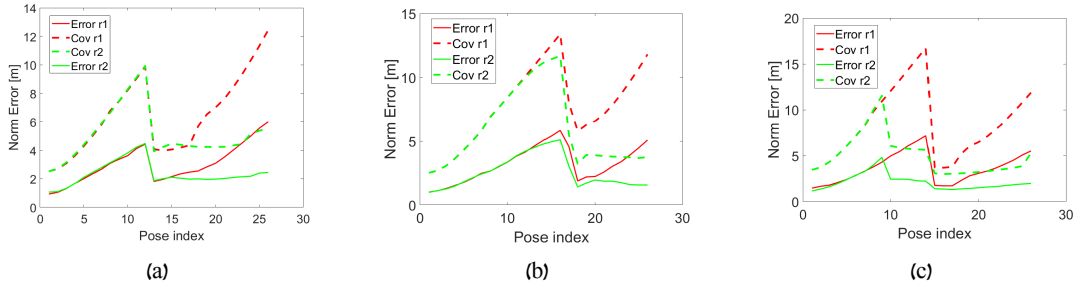


Figure 5.9: Performance comparison in terms of position estimation uncertainty and error: (a) 'Shortest-Path', (b) 'Highest-Correlation' and (c) 'Common-Cooperation'. Robot r_1 is the strategic robot (represented by red color); robot r_2 is the expendable robot (represented by green color). In scenarios (a) and (b) one can see that the state estimation of robot r_1 is indirectly updated when robot r_2 observes the known region. One can see the advantage of using the indirect cooperation concept (scenarios (a) and (b)) in terms of robot $r_1's$ update time compared to scenario (c).

One can see in Figure 5.9(a) and 5.9(b) that, given initial correlation between the states of both robots, the state estimation of robot r_1 is indirectly updated when robot r_2 observes the known region, despite the fact that r_1 is located in a different region and does not make any accurate observations on its own. Also, the results reflect the advantage using the indirect cooperation concept in terms of robot $r_1's$ update time, 12sec in Figure 5.9(a) compared to 15sec in Figure 5.9(c). To summarize this section, the indirect cooperation approach relaxes the cooperation requirement for observing mutual scene (5.9(c)) and enables the robots more flexibility in cooperation.

5.2.2 Expendable Cooperation - Compared Approaches

After demonstrating the advantages of the indirect cooperation concept we now proceeding to the results for the multi-robot expendable cooperation approach as described in Section 4.2.2. We compare our approach, denoted as 'Expendable-MR', with the previous approaches method (see e.g. [13, 21]), denoted as 'BSP-MR'. For each method we considered two scenarios: (i) the entire environment is scattered with landmarks and (ii) the goal is located within a region without landmarks at all.

The main objective for all the methods is that at least one (pre-defined) robot reaches the goal in minimum time and navigation error. In particular, in our approach only r_1 aims the goal while r_2 aims to reduce estimation uncertainty of r_1, while in BSP-MR both of the robots aim to reach the goal while minimizing their own uncertainty.

The general objective function for the two methods is given by Eqs. (4.13) and (4.15). In our simulation we considered the following specific costs for the two methods.

In our approach, the Expendable-MR method, we define robot r_1 as the strategic robot and robot r_2 as the expendable robot. For this method we use the following cost function,

$$\begin{aligned} cf_l^{Exp-MR} &= cf_l^{r_1} + cf_l^{r_2} \\ cf_l^{r_1} &= M_{\Theta}^{r_1} pathlen^{r_1} \\ cf_l^{r_2} &= M_{\Sigma_x}^{r_1} tr\left(\Sigma_{k+L}\right) \end{aligned} \tag{5.1}$$

where, $pathlen^{r_1}$ represents the path length of r_1 and $M_{\Theta}^{r_1}$ the weight matrix that sets the importance on this term. The matrix $M_{\Sigma_x}^{r_1}$ is used to choose the position covariance of robot $r_1's$ from the joint covariance Σ_{k+L} and determine the importance of it. The meaning of these cost functions is that r_1 plans to reach the goal with the shortest path while r_2 plans an optimal trajectory for reducing $r_1's$ position uncertainty using the expendable indirect cooperation concept we described in Section 4.2. In our simulation, robot $r_2's$ stopping condition is when the position uncertainty of robot r_1 is reduced below some threshold, i.e. r_2 stops after reaching the known region that indirectly updated robot $r_1's$ position state estimation.

In BSP-MR method, the objective function is as defined in Eq. (4.13), where each robot "cares" about its own state estimation. For this reason we define a similar cost function for both robots as follows,

$$\begin{aligned} cf_l^{BSP-MR} &= cf_l^{r_1} + cf_l^{r_2} \\ cf_l^{r_1} &= M_\Theta^{r_1} pathlen^{r_1} + M_{\Sigma_x}^{r_1} tr\left(\Sigma_{k+L}\right) \\ cf_l^{r_2} &= M_\Theta^{r_2} pathlen^{r_2} + M_{\Sigma_x}^{r_2} tr\left(\Sigma_{k+L}\right) \end{aligned} \tag{5.2}$$

The meaning of these cost functions is that each robot balances between reaching the goal with the shortest path while reducing its position uncertainty. In particular, as opposed to our approach, where only one robot has to reach the goal, here, both of the robots are tasked to reach the goal.

We observe that while the formulation (5.2) of BSP-MR method is common in literature and generally considered with non-zero weights, it can be reduced to the objective of our approach, i.e. Eq. (5.1), by setting $M_\Theta^{r_2} = M_{\Sigma_x}^{r_2} = 0$. As we shall see in the sequel, by assigning r_2 the role of an expendable robot, whose sole purpose is to aid r_1, and maintaining a joint multi-robot belief, which facilitates indirect multi-robot updates in presence of prior correlation, our approach is capable to improve the performance of robot r_1.

5.2.3 Expendable Cooperation - Simulation Results

We used a probabilistic roadmap (PRM) [26] to discretize the (partially unknown) environment and generate candidate paths over the roadmap. Figure 5.10 shows the considered scenarios for the two robots and the generated 20 candidate paths for each robot, 10 paths directly to the goal and 10 paths to the goal through the known region. The robots start near each other so there is a correlation between them from the beginning due to mutual landmark observation and a prior over the initial pose. Robot r_1 is represented with red color and robot r_2 is represented with green color.

Given the candidate paths, we calculate the optimal paths combination by using the objective functions defined in Eq. (5.1) and (5.2). The comparison between the two methods for each scenario is presented in the following figures. Note that the figure shows multi-robot SLAM results, i.e. the robots execute the optimal paths as determined by each of the methods, capture new sensor observations and use the latter to infer the posterior multi-robot belief. The robots' trajectories are shown in Figure 5.11, where the green square is a known region with uncertainty of $10^{-5}m$ and the unknown landmarks in the environment are the blue dots.

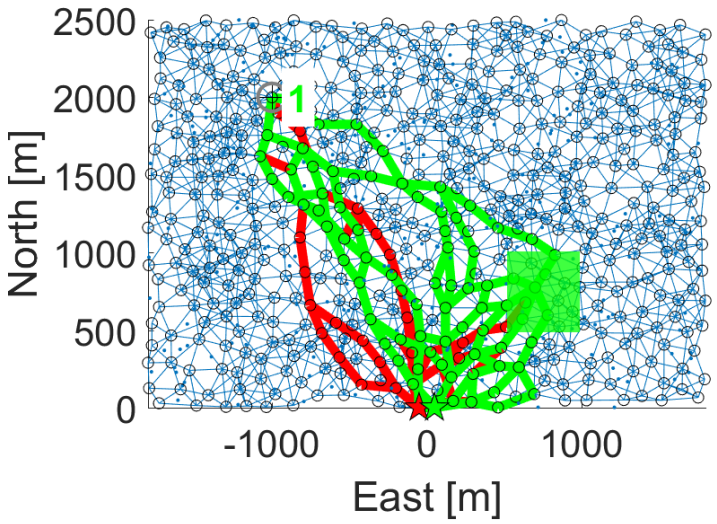


Figure 5.10: The environment discretization using probabilistic roadmap (PRM). The circles are the sampling points and the connections between the circles are the vertices that represent possible future robot locations. The generated candidate paths for the robots are the red (r_1) and green (r_2) trajectories.

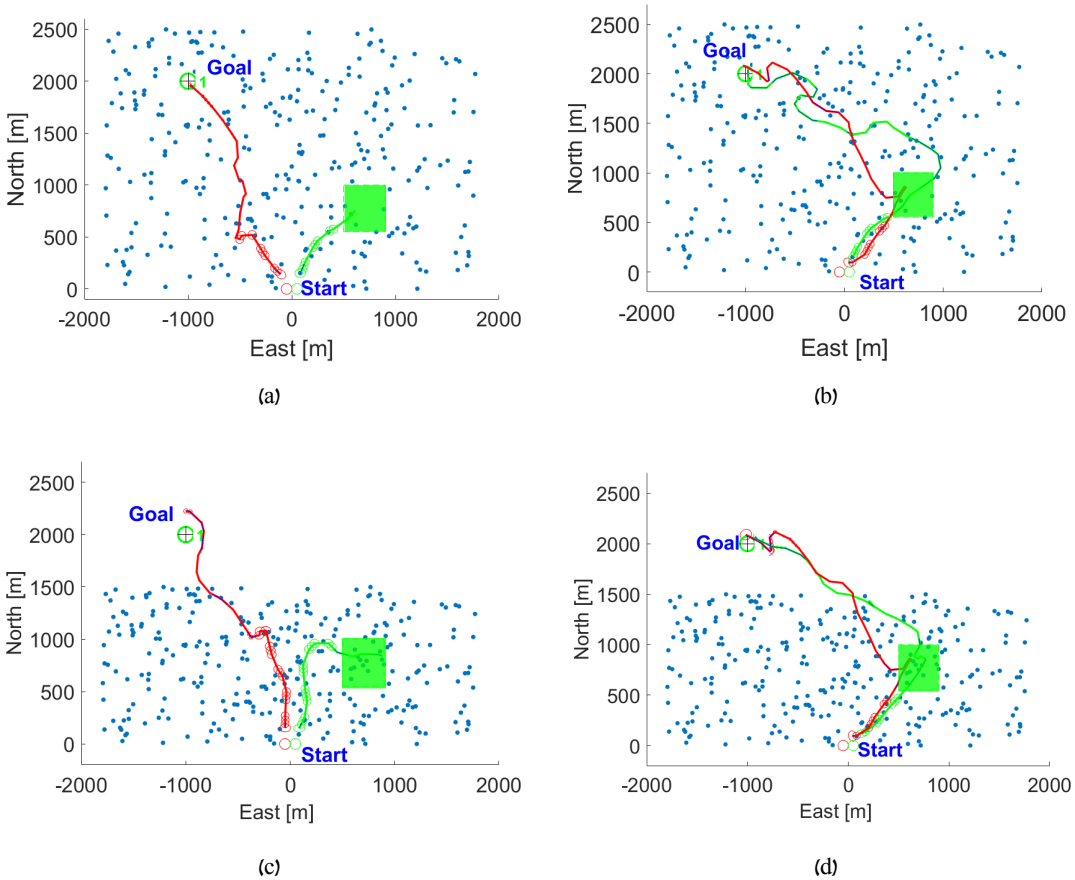


Figure 5.11: The robot's trajectories for the two methods considering the two scenarios: (a) 'Expendable-MR' with "full environment", (b) 'BSP-MR' with "full environment", (c)'Expendable-MR' with "dark-corridor environment" and (b) 'BSP-MR' with "dark-corridor environment". Where the green square is a known region with uncertainty of $10^{-5}m$ and the unknown landmarks in the environment are the blue dots. Robot r_1 is the strategic robot presented with red trajectory and robot r_2 is the expendable robot presented with green trajectory.

One can see that in the 'BSP-MR' method all the robots choose trajectories to the goal that pass through the known region. In contrast, in our approach, only the expendable robot plans to reach the known region and the strategic robot plans the shortest path to the goal while relying on the expendable robot to reduce its estimation uncertainty.

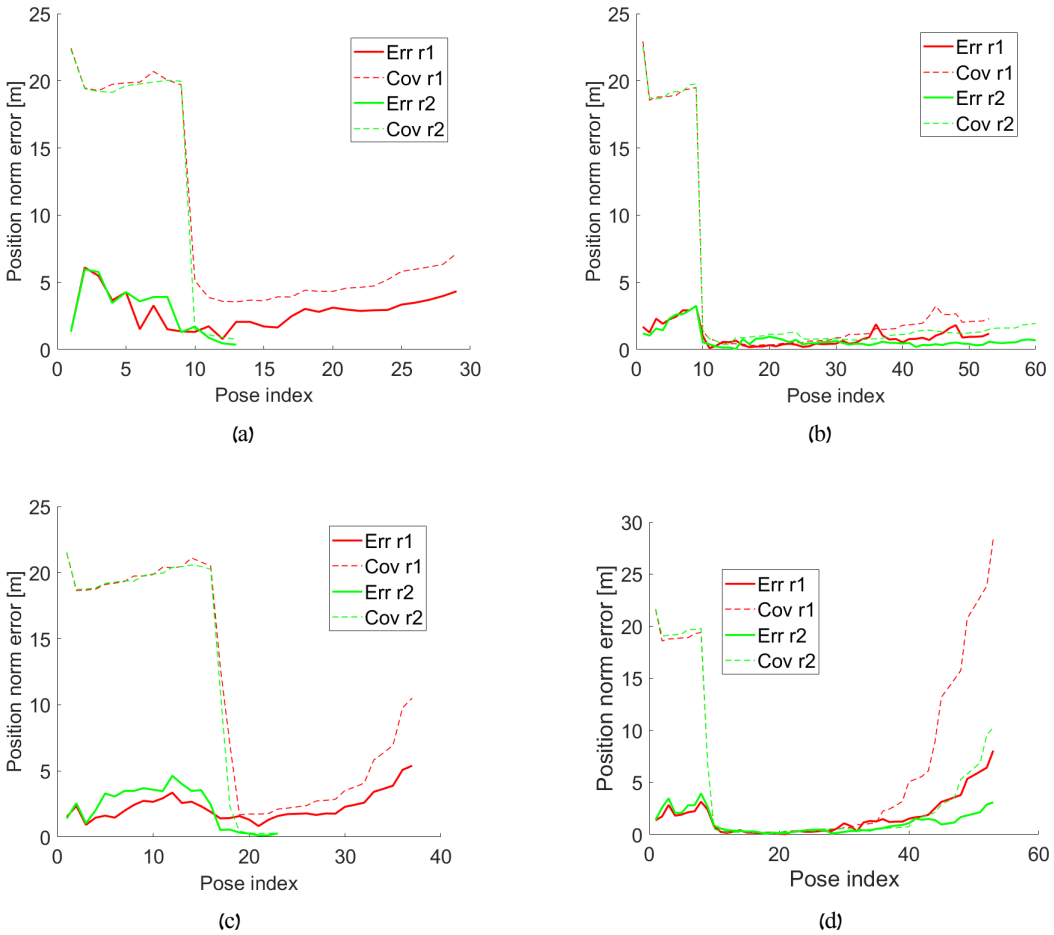


Figure 5.12: Performance comparison in terms of position estimation uncertainty and error for the two methods considering the two scenarios: (a) 'Expendable-MR' with "full environment", (b) 'BSP-MR' with "full environment", (c)'Expendable-MR' with "dark-corridor environment" and (b) 'BSP-MR' with "dark-corridor environment". Robot r_1 is the strategic robot presented with red lines and robot r_2 is the expendable robot presented with green lines. The covariance is presented with the dashed line and the error with the solid line. See text for details.

Performance comparison between the two methods for each scenario in terms of position estimation uncertainty and path lengths is presented in Figure 5.12 and summarized in Figure 5.13. One can observe that our approach, 'Expendable-MR' (Figures 5.12(a) and 5.12(c)), always yields better performance in terms of trajectory length of the strategic robot r_1, while keeping similar levels of accuracy compared to the 'BSP-MR' method. In particular, path length of robot r_1 is ~1.1km in our approach compared to ~2km for the 'BSP-MR' method. On the other hand, in terms of r_1 position uncertainty at the goal, the performance is scenario-dependent. For the

"full environment" scenario (Figures 5.12(a) and 5.12(b)), the 'BSP-MR' method yields slightly better performance (difference of ~5m between the covariances), probably due the continuous cooperation between the robots from the known region till they reach the goal. However, for the "dark-corridor environment" (Figures 5.12(c) and 5.12(d)), our approach yields significantly better performance (difference of ~20m between the covariances) for the same reason, i.e. for the 'BSP-MR' method, the robots stop the cooperation when entering the "dark-corridor" and the long path cause a bigger uncertainty drift. To summarize, in both scenarios, our approach yields enhanced performance of robot r_1, in terms of path length (short path to goal), while the expendable robot r_2 manages to reduce the uncertainty of r_1 to attain high accuracy upon reaching the goal.

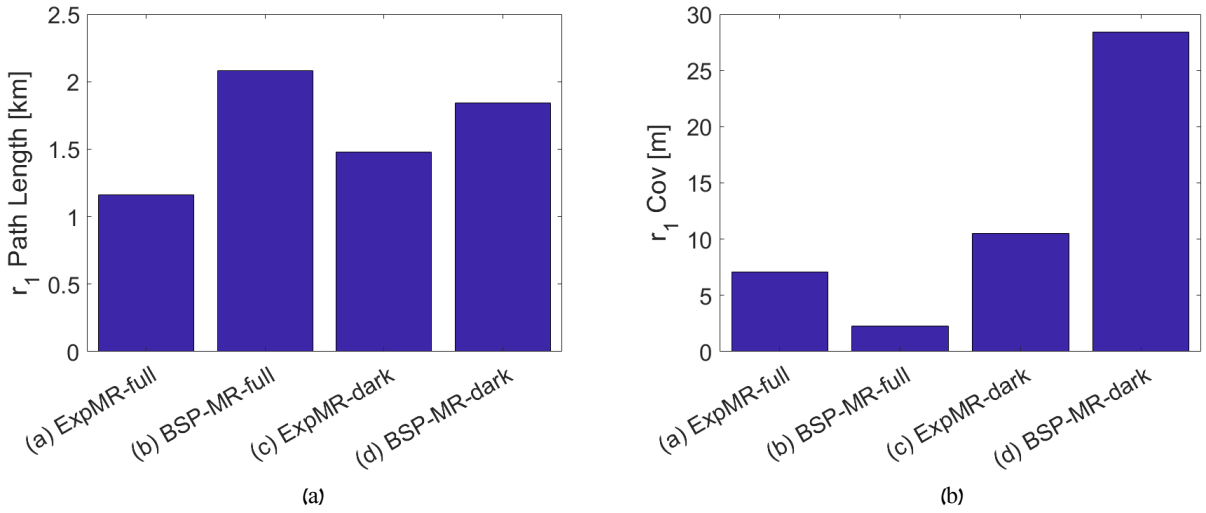


Figure 5.13: Comparison summery for the four simulation runs. Robot $r_1's$ (a) path length and (b) position covariance comparison. One can see that our approach always yield better results in term of path length. On the other hand, in term of navigation accuracy the performance is depend on the scenario, for example in a case with region without landmark our case probably will yield better results (also related to the path length).

Chapter 6

Conclusions

We addressed the problem of high-accuracy navigation for vision-aided inertial navigation systems considering both single and multi-robot frameworks and a GPS-deprived environment. Especially, we addressed two problems related to belief space planning in partially unknown environment, the first for single-robot framework and the second for collaborative multi-robot framework.

For the single-robot case, we presented an approach for active inertial sensor calibration for visual-inertial SLAM setup. This approach advances the state of the art in belief space planning by incorporating within the belief and objective function also the IMU calibration state estimation. By appropriately modifying the objective function, our approach is capable of calculating optimal actions for reducing estimation error within inference, reducing estimation error growth rate via IMU sensor calibration, or a combination of both.

The multi-robot framework builds upon the single-robot approach and introduces the concept of expendable indirect multi-robot cooperation. This approach further advances the state of the art by defining two collaborative robot subgroups, the "strategic group" and the "expendable group". The key idea is to have robots from the "expendable robots" improve performance, in terms of estimation accuracy, of robots from the "strategic group" while the latter execute time-critical missions. This approach is based on indirect cooperation concept, that enables one robot updating the other robots that can be in different regions of the environments.

We examined the performance of our approach in simulation, considering as application autonomous navigation to pre-defined goals within partial unknown environment. Simulation results demonstrate estimation performance is significantly improved as a consequence of incorporating calibration aspects within the belief or objective function and by using the expendable indirect cooperation concept.

As for future work, we plan to further investigate the expendable multi-robot cost function contributions and its connection to the general and common definitions. In addition, a concept for optimal adjustment of the weight matrices in the objective function. is a possible direction for increasing the navigation accuracy.

Chapter 7

Appendix - Indirect Cooperation Theoretical Analysis

We present a simple theoretically analyze for demonstrate the indirect update concept. We assume simple case of two robots, r_1 and r_2, starting with some initial correlation (e.g. mutual observation or prior knowledge). We will assume that robot r_1 observes an a prioiri known region and the other robot doesn't make any observation. Our hypothesis is that the covariance matrix of r_2 will be updated by the observation of the a prioiri known region by r_1 due to the initial correlation.

Let Σ represents the covariance matrix of the problem. The covariance matrix Σ or the information matrix I, are defined as

$$I \doteq A^T A = R^T R \ , \ \Sigma = I^{-1} \doteq \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \tag{7.1}$$

Where A is the Jacobian matrix and R is the square root information matrix. The matrix R can be calculated from the information matrix using Cholesky factorization.

Although we could have shown the following analysis via Kalman Filter in covariance form, we conduct the analysis in information form, specifically considering the square root information matrix R, because we use iSAM that use the information form [25, 23]. The usage of iSAM is more convenient in many aspects such as sparsity conservation, additive calculation and more.

The matrix R, after marginalization previous states as shown in the following equation, is an upper triangular shape.

$$p\left(x_i^r | x_{i-1}^r, u_{i-1}^r\right) = \int p\left(x_i^r | x_{1:i-1}^r, u_{i-1}^r\right) dx_{1:i-2}^r \tag{7.2}$$

For the simple case of our analysis its form is defined as shown in figure 7.1, where the columns and rows represent the robots and r_{12} represent the correlation between them.

The Jacobian matrix A, is the Jacobian over the observation model $z = h\left(x\right) + v$ and it is calculated by,

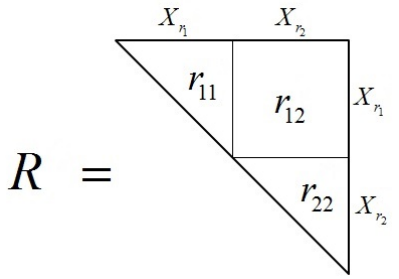


Figure 7.1: The square root information matrix R for the simple case of two robots

$$A \doteq \nabla_X \left(h \left(X \right) \right) = \left[\begin{array}{ccc} a_1 & \dots & a_n \end{array} \right] \tag{7.3}$$

where n is the number of robots. For our case, only robot r_1 observe a landmark, so in the observation time the Jacobian matrix is,

$$A = \left[\begin{array}{cc} a_1 & 0 \end{array} \right] \tag{7.4}$$

For our case we will examine the matrix R in two times: (a) The moment when the constraint between the robots was created, t_0, and (b) the moment when robot r_1 observed the oracle, t_k. To calculate R^k we need to update the matrix R^{k-1} with the Jacobian A^k, as shown in Figure 7.2.

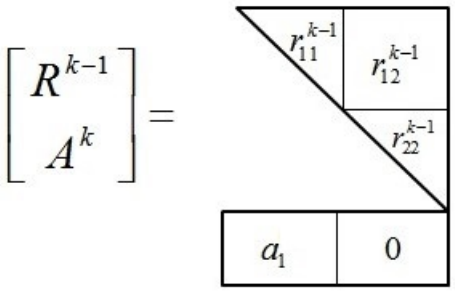


Figure 7.2: The matrix at time t_k after adding the Jacobian A^k. This matrix is updated using Given Rotation in order to calculate the a posteriori information matrix R^k

This update can be efficiently performed using Givens Rotations that are used to nullify all entries of A^k and appropriately updating the entries of R^{k-1} (see Figure 7.2), yielding the a posteriori information matrix R^k,

$$R^k = \left[\begin{array}{cc} r_{11}^k & r_{12}^k \\ 0 & r_{22}^k \end{array} \right]$$

In our case a_1 is a very accurate measurement due to the a priori known region observation. We want to show that this measurement impact also robot r_2 estimation and covariance, i.e. Σ_{22}^k will be updated although robot r_2 did not measured anything.

To calculate R^k from $\begin{bmatrix} R^{k-1} & A^k \end{bmatrix}^T$ we can use the relations in [15]

$$\left(r_{ii}^k\right)^2 = \left(r_{ii}^{k-1}\right)^2 + \left(a_i^{(i-1)}\right)^2 \tag{7.5}$$

$$a_j^{(i)} = \frac{-a_i^{(i-1)} r_{ij}^{k-1} + r_{ii}^{k-1} a_j^{(i-1)}}{r_{ii}^k} \tag{7.6}$$

Using these relations in our case we get

$$\left(r_{22}^k\right)^2 = \left(r_{22}^{k-1}\right)^2 + \frac{\left(-a_1^{(0)} r_{12}^{k-1} + r_{11}^{k-1} a_2^{(0)}\right)^2}{\left(r_{11}^{k-1}\right)^2 + \left(a_1^{(0)}\right)^2} \tag{7.7}$$

To calculate the covariance matrix of each robot at time t_k, Σ_{ii}^k, we can use the relations in [10, 22] using $R = (r_{ij})$

$$\Sigma^{ll} = \frac{1}{r_{ll}} \left(\frac{1}{r_{ll}} - \sum_{j=l+1,\, r_{lj} \neq 0}^{n} r_{lj} \Sigma^{jl} \right)$$

$$\Sigma^{il} = \frac{1}{r_{ii}} \left(- \sum_{j=i+1,\, r_{ij} \neq 0}^{l} r_{ij} \Sigma^{jl} - \sum_{j=l+1,\, r_{ij} \neq 0}^{n} r_{ij} \Sigma^{lj} \right)$$

for $l = n, \ldots, 1$ and $i = l-1, \ldots, 1$.

Generally in our case for moment t_j

$$\begin{aligned} \Sigma_{11} &= \frac{1}{(r_{11})^2} \left(1 + \frac{(r_{12})^2}{(r_{22})^2} \right) \\ \Sigma_{12} &= -\frac{r_{12}}{r_{11} (r_{22})^2} \\ \Sigma_{22} &= (r_{22})^{-2} \end{aligned} \tag{7.8}$$

Using Eq. (7.7) in Eq. (7.8) we get

$$\Sigma_{22}^k = \frac{1}{\left(r_{22}^{k-1}\right)^2 + \frac{\left(-a_1^{(0)} r_{12}^{k-1} + r_{11}^{k-1} a_2^{(0)}\right)^2}{\left(r_{11}^{k-1}\right)^2 + \left(a_1^{(0)}\right)^2}} = \frac{\left(r_{11}^k\right)^2}{\left(r_{22}^{k-1}\right)^2 \left(r_{11}^k\right)^2 + \left(-a_1^{(0)} r_{12}^{k-1}\right)^2} \tag{7.9}$$

It is shown in Eq. (7.9) that in the case without correlation between the robots, i.e. $r_{12}^{k-1} = 0$, the covariance matrix is not updated

$$\Sigma_{22}^k = \left(r_{22}^{k-1}\right)^{-2}$$

On the other hand, with a correlation between the robots and observation made by robot r_1, the covariance of robot r_2 is updated - the stronger the initial correlation, the stronger the impact.

Bibliography

[1] M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart. Path planning for motion dependent state estimation on micro aerial vehicles. In IEEE Intl. Conf. on Robotics and Automation (ICRA), pages 3926–3932, 2013.

[2] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas. Decentralized active information acquisition: Theory and application to multi-robot slam. In IEEE Intl. Conf. on Robotics and Automation (ICRA), 2015.

[3] Mitch Bryson, M. Johnson-Roberson, and Salah Sukkarieh. Airborne smoothing and mapping using vision and inertial sensors. In IEEE Intl. Conf. on Robotics and Automation (ICRA), pages 3143–3148, 2009.

[4] W. Burgard, M. Moors, C. Stachniss, and F. Schneider. Coordinated multi-robot exploration. IEEE Trans. Robotics, 2005.

[5] L. Carlone, M. Kaouk Ng, J. Du, B. Bona, and M. Indri. Rao-Blackwellized particle filters multi robot SLAM with unknown initial correspondences and limited communication. In IEEE Intl. Conf. on Robotics and Automation (ICRA), pages 243–249, 2010.

[6] F. Dellaert. Factor graphs and GTSAM: A hands-on introduction. Technical Report GT-RIM-CP&R-2012-002, Georgia Institute of Technology, 2012.

[7] F. Dellaert and M. Kaess. Square Root SAM: Simultaneous localization and mapping via square root information smoothing. Intl. J. of Robotics Research, 25(12): 1181–1203, Dec 2006.

[8] J.A. Farrell. Aided Navigation: GPS with High Rate Sensors. McGraw-Hill, 2008.

[9] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. On-manifold preintegration for real-time visual–inertial odometry. IEEE Transactions on Robotics, 2016.

[10] G.H. Golub and R.J. Plemmons. Large-scale geodetic least-squares adjustment by dissection and orthogonal decomposition. Linear Algebra and Its Applications, 34: 3–28, Dec 1980.

[11] Karol Hausman, James Preiss, Gaurav Sukhatme, and Stephan Weiss. Observability-aware trajectory optimization for self-calibration with application to uavs. arXiv preprint arXiv: 1604.07905, 2016.

[12] G. A. Hollinger and G. S. Sukhatme. Sampling-based robotic information gathering algorithms. Intl. J. of Robotics Research, pages 1271–1287, 2014.

[13] V. Indelman. Towards cooperative multi-robot belief space planning in unknown environments. In Proc. of the Intl. Symp. of Robotics Research (ISRR), September 2015.

[14] V. Indelman. Towards multi-robot active collaborative state estimation via belief space planning. In IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), September 2015.

[15] V. Indelman. No correlations involved: Decision making under uncertainty in the conservative information space - supplementary material. Technical Report ANPL-2016-01, Technion - Israel Institute of Technology, 2016.

[16] V. Indelman, L. Carlone, and F. Dellaert. Towards planning in generalized belief space. In The 16th International Symposium on Robotics Research, Singapore, December 2013.

[17] V. Indelman, L. Carlone, and F. Dellaert. Planning in the continuous domain: a generalized belief space approach for autonomous navigation in unknown environments. Intl. J. of Robotics Research, 34(7): 849–882, 2015.

[18] V. Indelman, P. Gurfil, E. Rivlin, and H. Rotstein. Graph-based distributed cooperative navigation for a general multi-robot measurement model. Intl. J. of Robotics Research, 31(9), August 2012.

[19] V. Indelman, S. Wiliams, M. Kaess, and F. Dellaert. Factor graph based incremental smoothing in inertial navigation systems. In Intl. Conf. on Information Fusion, FUSION, 2012.

[20] V. Indelman, S. Wiliams, M. Kaess, and F. Dellaert. Information fusion in navigation systems via factor graph based incremental smoothing. Robotics and Autonomous Systems, 61(8): 721–738, August 2013.

[21] Vadim Indelman. Cooperative multi-robot belief space planning for autonomous navigation in unknown environments. Autonomous Robots, pages 1–21.

[22] M. Kaess and F. Dellaert. Covariance recovery from a square root information matrix for data association. Robotics and Autonomous Systems, 57(12): 1198–1210, 2009.

[23] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In IEEE Intl. Conf. on Robotics and Automation (ICRA), Shanghai, China, May 2011.

[24] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. Intl. J. of Robotics Research, 31: 217–236, Feb 2012.

[25] M. Kaess, A. Ranganathan, and F. Dellaert. iSAM: Incremental smoothing and mapping. IEEE Trans. Robotics, 24(6): 1365–1378, Dec 2008.

[26] L.E. Kavraki, P. Svestka, J.-C. Latombe, and M.H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Trans. Robot. Automat., 12(4): 566–580, 1996.

[27] A. Kim and R. M. Eustice. Active visual slam for robotic area coverage: Theory and experiment. Intl. J. of Robotics Research, 2014.

[28] F.R. Kschischang, B.J. Frey, and H-A. Loeliger. Factor graphs and the sum-product algorithm. IEEE Trans. Inform. Theory, 47(2), February 2001.

[29] H. Kurniawati, D. Hsu, and W. S. Lee. Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces. In Robotics: Science and Systems (RSS), volume 2008, 2008.

[30] D. Levine, B. Luders, and J. P. How. Information-theoretic motion planning for constrained sensor networks. Journal of Aerospace Information Systems, 10(10): 476–496, 2013.

[31] T. Lupton and S. Sukkarieh. Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. IEEE Trans. Robotics, 28(1): 61–76, Feb 2012.

[32] Agostino Martinelli. Visual-inertial structure from motion: observability and resolvability. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4235–4242. IEEE, 2013.

[33] Jérôme Maye, Hannes Sommer, Gabriel Agamennoni, Roland Siegwart, and Paul Furgale. Online self-calibration for robotic systems. The International Journal of Robotics Research, page 0278364915596232, 2016.

[34] C. Papadimitriou and J. Tsitsiklis. The complexity of markov decision processes. Mathematics of operations research, 12(3): 441–450, 1987.

[35] J. Pineau, G. J. Gordon, and S. Thrun. Anytime point-based approximations for large pomdps. J. of Artificial Intelligence Research, 27: 335–380, 2006.

[36] T. Regev and V. Indelman. Multi-robot decentralized belief space planning in unknown environments via efficient re-evaluation of impacted paths. In IEEE Intl. Conf. on Robotics and Automation (ICRA), 2016. Submitted.

[37] Tal Regev and Vadim Indelman. Decentralized multi-robot belief space planning in unknown environments via identification and efficient re-evaluation of impacted paths. Autonomous Robots, pages 1–23, 2017.

[38] S.I. Roumeliotis and G.A. Bekey. Distributed multi-robot localization. IEEE Trans. Robot. Automat., August 2002.

[39] C. Stachniss, G. Grisetti, and W. Burgard. Information gain-based exploration using rao-blackwellized particle filters. In Robotics: Science and Systems (RSS), pages 65–72, 2005.

[40] David Titterton and John L Weston. Strapdown inertial navigation technology, volume 17. IET, 2004.

[41] R. Valencia, M. Morta, J. Andrade-Cetto, and J.M. Porta. Planning reliable paths with pose SLAM. IEEE Trans. Robotics, 2013.

[42] J. Van Den Berg, S. Patil, and R. Alterovitz. Motion planning under uncertainty using iterative local optimization in belief space. Intl. J. of Robotics Research, 31(11): 1263–1278, 2012.

[43] Dustin J Webb, Kyle L Crandall, and Jur van den Berg. Online parameter estimation via real-time replanning of continuous gaussian pomdps. In 2014 IEEE International Conference on Robotics and Automation (ICRA), pages 5998–6005. IEEE, 2014.

את איברי כיול הסנסורים. בנוסף, פונקציית המטרה מחשבת את המסלולים האופטימליים לפי הקטנת שגיאת המיקום, הקטנת קצב התפתחות השגיאות או שילוב של שניהם. שיטה זה יעילה במיוחד בסיטואציה בו קיים בסביבה איזור ללא ''נקודות עניין'' למצלמה, כדוגמת קיר לבן או מסדרון חשוך, בו הניווט הינו אינרציאלי בלבד. במקרה כזה, כניסה לאיזור זה עם רמת כיול סנסורים אינרציאליים טובה הינו יותר חשוב מאשר כניסה עם אי וודאות נמוך על מיקום הפלטפורמה.

שיתוף פעולה בין פלטפורמות הוכח כמשפר את ביצועי המערכת בהיבטים של דיוקי הניווט של הפלטפורמות או זמן מיפוי הסביבה. בעבודה זו פיתחנו גישה לשיתוף פעולה יעיל יותר ע''י הגדרת שתי תתי קבוצות של פלטפורמות - ''הקבוצה האסטרטגית'' ו''הקבוצה המסייעת''. הקבוצה האסטרטגית ממוקדת ביעדים העיקריים שלה, כדוגמת הגעה בזמן מינימלי לנקודת הסיום, ונעזרת בקבוצה המסייעת בשמירה על דיוקי המיקום הגבוהים הנדרשים ממנה. שיתוף פעולה זה מתאפשר ע''י פיתוח שיטה ל''עדכון עקיף'' בין פלטפורמות בהינתן קורלציה בין הפלטפורמות. כלומר, אנו מאפשרים שיתוף פעולה בין פלטפורמות, בהיבטים של עדכון שערוך משתני המצב, גם במקרה בו הרובוטים נמצאים בשני איזורים שונים. בפרט, אנו מגדירים פונקציות מטרה שונות לכל תת קבוצה. עבור ה''קבוצה האסטרטגית'' אנו מגדירים פונקציית מטרה, בדומה לעבודות קודמות, הכוללת את שערוך משתני המצב של קבוצה זו. אך עבור ה''קבוצה המסייעת'' אנו מגדירים פונקציית מטרה הכוללת את שערוך משתני המצב של ה''קבוצה האסטרטגית''. כך יוצא שבפועל פונקציית המטרה האחודה עבור על הפלטפורמות ''מטפלת'' בביצועים של ה''קבוצה האסטרטגית''. לדוגמה, עבור המקרה הפשוט של שתי פלטפורמות, פונקציית המטרה של הפלטפורמה האסטרטגית כוללת רק איבר של הגעה מהירה ליעד או מסלול קצר ביותר ליעד, ואילו פונקציית המטרה של הפלטפורמה המסייעת כוללת הקטנת אי הוודאות של שערוך משתני המצב (לדוגמה, מיקום או כיול) של הפלטפורמה האסטרטגית.

אנו מציגים את היתרון בשימוש בגישות שלנו ע''י סימולציה סינטטית המייצרת מדידות מצלמה ו- IMU סינטטיים. בסימולציה תוצאות הגישות שפיתחנו הושוו לגישות העדכניות ביותר בתחום והציגו שיפור משמעותי באספקטים רבים כדוגמת: אי הוודאות במיקום, זמן ההגעה לנקודת הסיום, ביצועים באזור המאפשר ניווט אינרציאלי בלבד (''מסדרון חשוך'') ועוד.

תקציר

בעבודה זו אנחנו מפתחים גישה חדשה של תכנון מרחב הסתברותי לכיול אקטיבי של סנסורים בסביבה ידועה חלקית, עבור מערכת מרובת פלטפורמות המצוידות במערכות ניווט אינרציאלית נעזרת ראייה ממוחשבת ומשתפות פעולה אחת עם השנייה. בניווט של רובוטים בסביבה לא ידועה, יש צורך לפתור שתי בעיות שערוך בבת אחת: שיערוך המיקום של הרובוט ומיפוי הסביבה. בעיות שערוך אלו תלויות אחת בשניה ולכן הפתרון הוא עבור שתיהן בו זמנית. כלומר, שערוך מיקום הרובוט מצריך ידיעה מדויקת (ככל האפשר) של מיקום העצמים במפה, ומיפוי מדויק של העצמים במפה מצריך ידיעה מדויקת של מיקום הרובוט. בעיה זו ידועה כ- Simultaneous Localization and Mapping (SLAM).

אנו משתמשים במודל גרפי הנקרא "גרף פקטורים" (Factor Graph) כדי למדל את הבעיה. במודל הצמתים מסמלים את הנעלמים האקראיים בבעיה, למשל מיקום הרובוט ומיקום העצמים בסביבה. הקשתות או הפקטורים מסמלים את האילוצים בין נעלמים אקראיים הללו. לדוגמה, מודל התנועה של הפלטפורמה מקשר בין המיקום הנוכחי למיקום הבא ובצורה דומה, מודל החיזוי של המצלמה מקשר בין המיקום של הפלטפורמה לעצמים בסביבה שמופו ממיקום זה.

בעבודה זו אנו מציגים שתי גישות. הגישה הראשונה מציגה כיול אקטיבי, תוך כדי טיסה, של סנסורים אינרציאליים עבור פלטפורמה בודדת המצוידת במערכת ניווט אינרציאלית נעזרת ראייה ממוחשבת. הגישה השניה, מרחיבה את הגישה הראשונה ומציגה שיטה לשיתוף פעולה יעיל יותר עבור מערכת מרובת פלטפורמות להשגת ביצועי ניווט גבוהים של המערכת. אחד החידושים העיקריים בעבודה זו הינו שבשתי הגישות המרחב ההסתברותי (Belief), הפילוג הסתברותי עבור הנעלמים האקראיים של הרובוט, מכיל גם את הנעלמים האקראיים של כיול הסנסורים האינרציאליים. שתי הגישות מבוססות על תכנון מסלולים אקטיבי ע"י הגדרת פונקציית מטרה על פיה מחושב המסלול האופטימלי לכל פלטפורמה.

בעבודה זה הרחבנו שיטה שפותחה לאחרונה הנקראת "איחוד מדידות מערכת מדידה אינרציאלית" (Pre-Integrated IMU) גם עבור זמנים עתידיים (לטובת תכנון המסלולים). שיטה זו באה לתת מענה למצב בו קיימות מדידות בתדר גבוה, כדוגמת IMU, כך שנדרש לעדכן את גרף בעיה בקצב הגעת המדידות. דבר זה מצריך כוח חישובי גבוה ולרוב לא מאפשר ביצועי זמן אמת. בנוסף, עבור אופק התכנון מגבלה זו עשויה להקטין את אופק התכנון האפשרי. העיקרון בשיטה זו הינו איחוד מספר מדידות למדידה אחת אחודה (בד"כ לפי תדר המדידה של הסנסור האיטי במערכת). שיטה זו מאפשרת ביצועי זמן אמת ואופק תכנון ארוך עבור מדידות IMU בתדר גבוה ללא צורך ביכולות חישוב גבוהות.

כיול אקטיבי תוך כדי טיסה של סנסורים, ובפרט סנסורים אינרציאליים הסוחפים עם הזמן, הינו יכולת משמעותית המשפיעה ישירות על קצב התפתחות השגיאות של המערכת וכן על דיוק הניווט הסופי. בעבודה זו משתני המצב כוללים, בנוסף למשתנים הסטנדרטיים (מיקום ומהירות הפלטפורמה ומיפוי הסביבה),

המחקר בוצע בהנחייתו של פרופ"מ ואדים אינדלמן מהפקולטה להנדסת אווירונאוטיקה וחלל בטכניון - מכון טכנולוגי לישראל

תודות

ברצוני להביע את תודתי הרבה לפרופ' ואדים אינדלמן על ההנחיה, העזרה וההכוונה בכל שלבי המחקר והתואר השני. תודה על הדחיפה המתמדת וההסברים המפורטים.

למשפחתי הנפלאה, אני מודה לכם על העידוד והסבלנות לאורך כל הדרך. תודה מיוחדת לאשתי מורן על התמיכה הבלתי מוגבלת.

תכנון מרחב הסתברותי לכיול אקטיבי של סנסורים עבור מערכת מרובת פלטפורמות המצוידות במערכת ניווט אינרציאלית נעזרת ראייה ממוחשבת

חיבור על מחקר

לשם מילוי חלקי של הדרישות לקבלת התואר
מגיסטר למדעים

יאיר בן-אלישע

הוגש לסנט הטכניון — מכון טכנולוגי לישראל
תשרי ה'תשע"ח חיפה ספטמבר 2017

תכנון מרחב הסתברותי לכיול אקטיבי של סנסורים עבור מערכת מרובת פלטפורמות המצוידות במערכת ניווט אינרציאלית נעזרת ראייה ממוחשבת

יאיר בן-אלישע